Driver Cognitive Workload Detection via Eye-tracking and Physiological Modalities

by

Xin Zhao

A thesis submitted in conformity with the requirements
for the degree of Master of Applied Science
Department of The Edward S. Rogers Sr. Department of Electrical Computer Engineering
University of Toronto

Abstract

In recent years, autonomous car development has become one of the hottest topics in AI applications and the driver cognitive workload monitoring system is a critical element of the autonomous car. This study explored the feasibility of classifying driver cognitive workload levels with eye-tracking and physiological modalities individually. Around a 70% detection accuracy was obtained with both modalities for ternary classes.

Support Vector Machines (SVM) with a Gaussian Kernel function are utilized to build a monitoring system with 5-fold cross-validation. Principal component analysis (PCA) was investigated in terms of system performance. The time gaps between training and testing data are analyzed and the feasibility of using the off-line pretrained model to detect driver cognitive workload is investigated.

Acknowledgements

I would like express my deep gratitude for the advise and guidance I received from Professor Konstantinos N. Plataniotis. I could not accomplish this thesis without his valuable insights and feedbacks. Furthermore, I truly appreciate the valuable time from my colleagues in Multimedia Lab, especially Haiyan Xu, spent on advising my presentations.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

With the development of automatic driving vehicle techniques, more attention is being paid to safety issues related to self-driving vehicles. For a level 5 fully automatic vehicle as defined by SAE International, no human interaction is involved, and AI is entirely responsible for driving. However, no company can guarantee the accuracy of the AI system, and a human still needs to take over the controls when necessary. The driver needs to put their hands back on the wheel after a maximum of 10 seconds hands-free for the Tesla autopilot model. However, the driver keeping their hands on the wheel cannot guarantee that they will be able to take control when necessary. Also, the driver's attention on the driving task could be distracted by the In-Vehicle Intelligent Systems (IVIS).

A Driver State Monitoring System is required, especially for driver cognitive workload level detection. Cognitive distractions are distractions that keep your mind from staying focused during driving. If something else captures your attention or if you are having trouble concentrating on the road, it could lead to potential accidents. This may be caused by emotional stress, family or money problems, talking to someone else in the fleet vehicle, or using a phone. Compared to the other two types of distractions (manual distractions and visual distractions), cognitive distractions are less explored in the field of driver state monitoring. Cognitive distractions are much more challenging to detect compared to the other two types of distractions since they can be observed with cameras. Therefore, a driver cognitive workload level monitoring system is essential to

develop. This project will focus on detecting this type of inattention with eye-tracking and physiological measures using machine learning techniques.

1.1 Background

In this section, various modalities to model driver cognitive workload are addressed: **a)** Subjective Measures, **b)** Psycho-physiological Measures(ECG, Galvanic Skin Response, Respiration) and **c)** Performance Measures. Each of these modalities presents its own set of advantages and challenges. Among these modalities, why physiological and eye-tracking modalities were chosen to detect drivers' cognitive workload will be explained based on the discussion of the pros and cons of each modality. Detailed information on these modalities is given below.

1.1.1 Subjective Measures

Subjective measures are considered to be the easiest and least expensive way of evaluating workload. One of the most widely-known methods for performing subjective measures is the NASA Task Load Index (NASA TLX). This index contains six rating scales: Mental Demands, Physical Demands, Temporal Demands, Own Performance, Effort, and Frustration[3]. Questionnaires for self-reporting workload commonly refer to a task already performed. This means the self-reporting of workload usually covers only a single task and answering questionnaires is an intrusive procedure since it adds another task to the existing task. With these considerations, self-reporting questionnaires are designed to be done after perform- ing the secondary task. However, this brings uncertainty about the accuracy of the self-assessment since it is not simultaneously estimated. Also, subjective measures are performed after completing the secondary task, and subjective measures can not detect sudden variations. Thus, subjective measures are not suitable for a real-time driver monitoring system.

1.1.2 Performance-based Measures

With the assumption that an increased cognitive load diminishes driver performance, performance-based measures can be utilized as one indicator of the scale of cognitive workload. This assumption is backed by the Yerkes-Dodson-Law[4] which is a classical model used to relate task performance with mental arousal. The Yerkes-Dodson-Law states that task performance degrades when the arousal level is low, which corresponds to the danger of driving while fatigued or in low-vigilance situations.

Numerous articles and reports have investigated the effects of cognitive load on driving performance. Performance-based measures included indicators of driving performance such as lane deviation, speed, steering wheel angle, reaction time and time perception. Driving performance measurements are convenient to acquire through mature commercial products with high user acceptance.

However, for real-time applications, driving performance under high cognitive workload might depend on the participant's awareness of potential risks. The majority of articles indicate that cognitive load has little or no effect on longitudinal control in a simulated environment[5, 6, 7, 8]. However, [6, 9] found that there was a reduction in speed when hand-held phone tasks were performed. Also,[5] indicated there is a speed increase when the driver is instructed to drive slower than the typical speed for the road. Participants are consciously aware of increased risks introduced by lower instructed driving speed and hand-held phone use. This may indicate that when drivers are consciously aware of increased risks, they perform a type of compensation to reduce their speed to maintain an acceptable risk level. This conflicts with the findings in [5, 6]. Based on the discussion, performance-based measures might not accurately reflect driver cognitive workload level in real-time applications.

1.1.3 Physiological Measures

Physiological measures include heart rate, skin conductance level, Electrocardiogram (ECG), Electroencephalogram (EEG), respiration rate and other types of indicators. Those measures are reliable and accurate and are sensitive to the driver's inner state

changes. However, those measures might not be very robust when outside of a laboratory. Increases in heart rate and galvanic skin response were commonly observed when the mental workload increased [10, 11, 12]. With real-world measurements, the noise produced by the movement of the body profoundly affects measurements that use physiological sensors. Also, physiological modalities are highly intrusive as external hardware is worn by the user.

1.1.4 Video-based Measures

The video-based measures use facial-derived information such as blinks, gaze, head movements and facial expressions to model cognitive workload. Those measures are sensitive to driver states and are excellent for capturing drivers' behavioral responses via movement in real time. However, the computational requirements needed to process the images captured are high compared to other types of measures.

1.1.5 Summary

For a practical real-time driving monitoring system, it is unrealistic and unreliable to use subjective measures and performance-based measures. Therefore, external observations of the driver's body response is the focus of this study. With the development of this tech- nique, eye-tracking commercial sensors are utilized to compensate for the high computation time associated with image processing techniques. Also, less intrusive wearable sensors are being developed to obtain physiological measures, which is helping to monitor driver state. Thus, physiological and eye-tracking measures will be focused on in this study.

1.2 Problem Statement

This thesis aims to develop a driver cognitive monitoring system which could be employed by smart vehicles. Specifically, this study aims to measure driver cognitive workload detection with less intrusive sensors. To ensure the user's acceptance, eye-tracking and

physiological measures will be the focus in monitoring driver cognitive workload.

To achieve the final goal of this study, two research questions need to be answered:

- What kind of features carry the most predictive power for driver cognitive workload level?
- Can an off-line trained model be used to monitor driver cognitive workload?

These questions are explored in the ensuing subsections, with explanations of challenges associated with them.

1.2.1 Feature Extraction

The most difficult problem associated with developing a driver monitoring system is the lack of domain knowledge about driver cognitive workload indicators. Several studies found that drivers' heart rate, blinking rate and pupil diameter are good indicators for cognitive workload detection. Based on this prior knowledge, we utilized the same kind of features extracted from eDREAM data to detect the driver cognitive workload level. Usually, those features need to be summarized within a length of data , and averaging is the commonly-used summarization function. Other summarization functions such as standard deviation value and root mean square values are also widely used in this field. Those summarization functions were also utilized in this study. We explore how summarization parameters affect the system performance. Data mining techniques' Principle Component Analysis is explored to improve the system performance. Machine learning algorithms with Support Vector Machines are explored for building subject-independent models for monitoring driver workload levels. We evaluated how well the proposed features perform and discuss issues associated with this process. The research hypothesis for this step is that the proposed features carry predictive power.

1.2.2 Off-line Trained Model

The current studies are contributing to building a system with a pretrained model and using it to monitor driver cognitive workload. However, the variability of the measures

collected from drivers are not taken into consideration. Thus, this study aims to analyze the feasibility of using an off-line trained model for monitoring drivers' cognitive workload. Model performance built with data collected within the same time periods are compared within different time periods. With the eDREAM dataset, the maximum time difference that existed during the data collection procedure was 1.5 minutes. We evaluated how well the system performs with data collected within a time range and how well it performs with data collected outside the time range. The research hypothesis for this step is that the driver cognitive workload level may vary after the driving time increases.

1.3 Thesis Contributions

The following contributions are made towards the development of a driver cognitive monitoring system:

- We synchronized the physiological measures with a miniSIM driving simulator to extract and label the collected data.
- We proposed features that are sensitive to driver cognitive workload level for eye-tracking modalities and physiological modalities.
- We examined the effect of the feature summarization function on system performance.
- We investigated the feasibility of using an off-line trained model to detect driver cognitive workload.

1.4 Thesis Organization

The thesis is organized as follows:

- **Chapter 1** summarizes the motivation of this study and introduces various modalities used for cognitive workload detecting in the context of automatic driving. The background and research scope are identified in this chapter.

- **Chapter 2** describes prior works and state-of-the-art studies performed with the same data collection sensors. There is further discussion of features used in the literature and the accompanying results obtained.

- **Chapter 3** provides an overview of the dataset utilized in this study including the sensors apparatus and the various features available for each modality. The pre-experimental implementation such as time synchronization is performed and discussed. The increased discrimination among different levels of the cognitive workload with the summarization functions applied in this study is discussed and presented.

- **Chapter 4** provides the experiment pipeline, feature processing methodology and simulation results for both visual information and physiological modalities. Also, the relationship between performance and the time gap between testing dataset and training dataset is analyzed and discussed.

- **Chapter 5** concludes on the contributions of this thesis and provides suggestions for future research.

Chapter 2

Related Work

This chapter provides a summary of the related works that were drawn on when performing this study. The flow of the literature research is shown in Figure 2.1. Only literature relevant to this study are reviewed and several conditions need to be met in order to be considered relevant. The data collection set up, sensors utilized for data acquisition and methods used to stimulate the cognitive workload level are key factors that need to be considered and compared with this study. Two different experimental settings will be discussed first: on-road driving and driving with a simulator. Sensor noise and environmental variables need to be taken into consideration when designing the data collection protocol. The data utilized in this study were collected under lab experimental conditions, which can limit the number of variables that might affect the cognitive workload labeling accuracy. Commercial sensors for data acquisition have a built-in algorithm to remove noise that might affect the results. It is notable to compare the results obtained with the same brand sensors in the literature. Then, two approaches for feature number reduction were explored in this study: the top-down or bottom-up process. The top-down process focuses on a small set of features that are hand-picked based on prior knowledge in this area. And the bottom-up process utilized a statistical learning approach to select a set of features from the feature pool.

This chapter consists of three parts:

- Section 2.1 compares the statistical findings when participants were driving on the

road and driving on a simulator

- Section 2.2 introduces statistical findings on eye-related measures and physiological measures in the related domain of increased driver cognitive load,
- Section 2.3 state of-the-art driver cognitive load estimation systems,
- Section 2.4 explores data fusion techniques applied in the literature that might be implemented in future works.

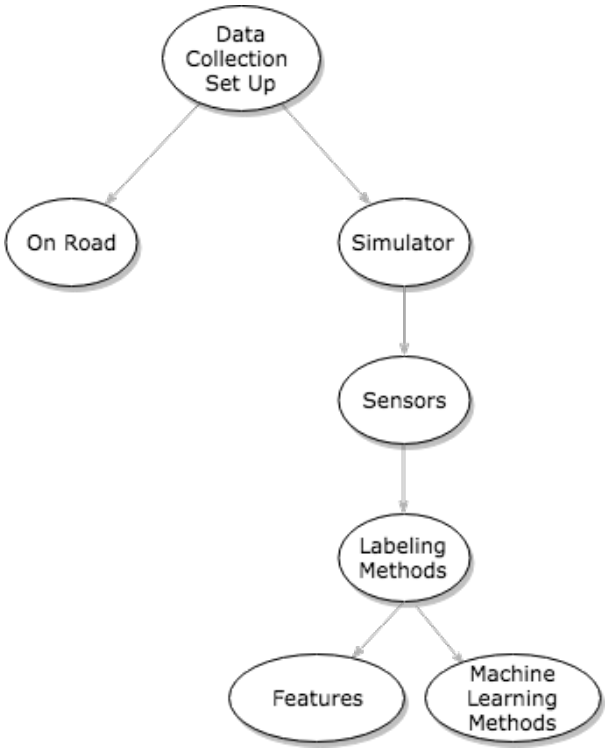


Figure 2.1: Literature review flowchart

2.1 Experimental Setup

Based on the dataset used for investigating cognitive workload when performing a secondary task while driving, researcher approaches can be categorized into two main classes: simulation datasets and real driving datasets. In this section, whether observations made during on-road driving can also be noticed with the simulator is discussed. If the same

patterns were noticed, then approaches to access drivers' cognitive workload during on-road driving can be borrowed and applied in this study.

[13, 12] described changes in heart rate and skin conductivity under an artificially controlled cognitive workload level, which was induced with n-back tasks under on-road driving conditions. The result showed that those two physiological measures can discriminate different levels of cognitive demand since they both increased significantly along with the increase in cognitive demand. This replicates the simulation work [14] by showing a remarkably consistent pattern of heart rate change in an on-road setting using the same protocol. Skin conductance data collected in the field also show a pattern similar to that seen in the simulation data. In this study, driver cognitive workload level was introduced with n-back tasks, and the same pattern was also expected in this study.

[15] designed a virtual driving environment which replicated a previous real-world experiment and they compared the data they collected in this experiment with data collected in a real-world experiment under different cognitive loading conditions. The participants were required to drive three different routes, each of which had four sections, and they were asked to finish four tasks with only primary driving tasks, four verbal tasks and four spatial tasks, separately. The participants needed to generate a word which began with one of the given four letters in the verbal tasks and imagine the vertical and horizontal rotation of the letters in the spatial tasks. The authors found that the patterns that characterized eye movement data collected in the simulator were identical to those that characterized eye movement data collected in the real world. However, in this study, cognitive workload was induced with verbal and imaging tasks, which might affect the observed pattern compared to data stimulated with the n-back task.

Narrow gaze concentration or visual tunneling are both observed in simulator-collected data and on-road assessment[16, 17, 18]. While the driver performed at a certain level on the secondary task which would introduce cognitive workload, the gaze distributions were significantly smaller[17]. Also, [15] found that the number of speedometer checks was significantly decreased in both verbal and spatial tasks. Eye-scanning behavior for studies with the simulator and on-road vehicle showed similar patterns.

Based on the literature discussed above, it has been shown that the same pattern

would be observed under both on-road and simulator conditions. However, in real driving conditions, the collected data would be noisier because a moving vehicle can present challenges such as variations in lighting, changes in background and vibration noise; also, sunglasses definitely affect the tracking of eyes. Also, performance or measures not only depend on the designed task, but other traffic also produces high cognitive load. Few researchers have dealt with a real-world driving environment due to the danger produced by the secondary task. Simulator environments were focused on in the later literature.

2.2 Feature Selection

As described above, feature selection can be based on prior knowledge and also automatically extracted by machine learning algorithms. In this section, the feature selection methods used in the literature will be discussed based on the modality used.

In the initial study, it was realized that there is no specific index to scale or quantify the cognitive workload of the driver. Based on the task itself, the effects on the driver vary as shown in different observations. Thus, only studies conducted with n-back tasks were analyzed here. Two approaches can be utilized to extract the features for classification: a manual features selection approach based on studies from the literature and machine learning techniques to extract the most essential features from the pool of features.

2.2.1 Manually Selected Features

Table 2.1 shows related studies that focus on detecting drivers' cognitive workload, and the results are obtained through a laboratory experiment. Compared to field experiments, laboratory experiments are more controllable, which means the variables that could affect the experimental results are limited and have been considered in the process of designing the experiment. All the above related studies except [27] all used FaceLab as the eye-tracker, which is the same as the eDREAM dataset. In this case, the result obtained with the eDREAM dataset is comparable with them.

Liang [23] classified distraction into four different binary states based on task condition and driving performance. Three feature combinations are used in the experiment

Table 2.1: Summary of features employed for detecting driver cognitive load with eye-tracking measurements

	Acquisition Equipment	**Feature Used**	**Secondary Task**
[19, 20]	FaceLab	Standard Deviation of (Vertical Gaze, Horizontal Gaze)	N-back
[21]	FabeLab	Standard deviation of combined (gaze angle of eyes, head orientation angle) Eyes and head tracking quality	Arithmetic loads involved verbally subtracting a prime number (for example 7) from 1,000 successively. Conversation loads involved asking the subjects to describe a route which they regularly commuted (such as the road from school to home).
[22]	FaceLab	Standard deviation of combined (gaze angle of eyes, head orientation angle) Eyes and head tracking quality Pupil diameter	Same as the one above
[23, 24, 25]	FaceLab	Fixation (duration, mean of position) Pursuit(duration, distance, direction, speed and percentage of pursuit in time) Mean of blinking frequency	Three driving tasks and auditory stock ticker
[26]	FaceLab	3-D angles of head rotation blink frequency, percentage of eye closure pitch/yaw angles for left and right eye gaze for eye movements	Auditory counting

including eye data, eye minus spatial data and eye plus driving. Eye data features were obtained from gaze vector-screen intersection coordinates. First, the raw eye data were categorized based on two characteristics: dispersion and velocity. Dispersion shows the range of the gaze vector covered in radians and velocity describes the speed and direction

of the gaze vector. Both dispersion and velocity are calculated within 6 frames with a 60 Hz sampling frequency. Based on the likelihood of dispersion and velocity for fixation, pursuit and saccade, the eye movements were categorized after exceeding the posterior probability. The summary of those three categories: duration, distance and direction are calculated and treated as eye data to train the model. Three combinations of features showed that adding spatial information on eye movements contributes to the detection of driver distraction by increasing the model sensitivity. Spatial information describes the mean of the horizontal and vertical position of fixation. In her later research [24]], one more feature was included, which is blinking frequency, and also [28] proved that blinking frequency reduced the uncertainty in the detection by 37%. Blinking frequency and spatial information are also examined in this study. The eye fixation duration is not in- cluded in this study, even though the author in [23] demonstrated that it contributes to the detection rate of drivers' cognitive workload. Studies have shown contradictory findings in the relation between workload level and fixation duration [29, 30].

Miyaji et all [22, 21] ed conduct an experiment using a mock-up type driving simulator. Subjects were instructed to pay attention to the speedometer and keep their speed around 60 km/h. Subjects first practiced familiarizing themselves with driving, then drove without cognitive loads, followed by driving with arithmetic loads, and driving with conversation loads. Arithmetic loads involved verbally subtracting a prime number (for example, 7) from 1,000 successively. Conversation loads involved asking the subjects to describe a route which they regularly commuted (such as the road from school to home). Individual features and combined features were both examined. Three kinds of features were involved in this experiment: pupil diameter, heart rate and visual information. Visual information contained the gaze and head rotation angle. The gaze and head rotation angle were converted from the pitch angle and yaw angle. With pupil diameter alone, performance can reach 85%, and detection accuracy increased to 87.7% with visual information together. Among the three features, pupil diameter was larger than visual information in the order based on the accurate result in F value. Also, with the combination of two features out of three, pupil diameter and heart rate demonstrated a higher F value than others. It can be concluded that pupil diameter contributed a lot

in detecting drivers' cognitive workload compared to visual information. In this study, pupil diameter is also taken into consideration during the experiment. Based on the above studies, pupil diameter, the mean value of the fixation position of eye movement and blinking frequency are key features that can be used as an index to detect drivers' cognitive workload. However, in those studies, the cognitive workload is stimulated with auditory tasks including stock price and conversation. Few studies have used an n-back as a secondary task and applied classification to distinguish the level of cognitive workload. In studies by Son et al [19, 20], the experimental scenario is very similar to this study. First, it was conducted on a simulator. As described above, if the driver drove with caution it would considerably affect the data collected. When the experiment was conducted in a one-road driving environment, subjects drove with caution compared to the driving simulator. Second, the acquisition equipment used was also FaceLab with a 60-Hz sampling frequency. Last and most important, the secondary task they used to stimulate the subject's cognitive workload was an n-back task. The experiment was conducted in a fixed-base driving simulator and four levels of cognitive workload were considered in this experiment from baseline to 2-back tasks. In the end, n-back tasks with n equal to 0,1, and 2 were categorized as low-medium-high workload loads. Two types of eye movement data included standard deviation of horizontal gaze and the standard deviation of vertical gaze as the input features to detect high cognitive workload demand. Raw data collected with FaceLab were filtered with 3 criteria to calculate eye movement. First, the gaze quality index of both eyes should be categorized as optimal. Then, the gaze position should be in a design range. Last, the data point should be within a set of six valid measurements. After that, those points were summarized with window size to remove noise. With eye movement features alone, the performance can reach to 83.3% when a 30-second window size is chosen. A reproduction of drivers' cognitive workload detection is conducted in this study and the results are obtained with the eDREAM dataset. The model performance comparison is discussed in a later section.

For physiological measures, the most widely used indicators for drivers' cognitive workload are heart rate (HR), electroencephalogram activity, respiration rate, skin conductance level (SCL), and others [31, 12, 14]. The literature on measuring physiolog-

ical signals in the car can be divided into simulator studies [32, 23, 14] and on-road studies[33, 23, 17]. With increased cognitive demand, heart rate and skin conductivity both increased significantly for both simulator and on-road conditions [13, 12].The authors claimed that heart rate and skin conductance level are sensitive for discriminating discrete differences in demand associated with n-back tasks but might vary with a different task. In this study, the n-back task was utilized. . Thus, heart rate and skin conductance level are expected to be good indicators of drivers' cognitive workload level.

2.2.2 Statistic Methods for Feature Selection

With a large number of features, the features contain most of the information needed to be selected. This could help to remove some features that are either redundant or irrelevant and also reduce the computation time for classifiers with a lower number of features.

A forward feature selection algorithm (Sequential Forward Selection – SFS) was used to find the best-reduced feature set of nine features [34, 32]. A Decision Tree was used in [27] and all possible inputs were considered. In [35], a high number of dimension input signals was eliminated, leaving around 20 features with regularization when doing a linear regression. Stepwise regression was used for feature number reduction in [36] with 945 features. Also, in [37],15 features remained after applying principal component analysis (PCA). In this study, PCA is also explored with eye-tracking measurements to reduce the number of features from 117 features to 15. For physiological measures, no PCA was implemented, as the number of features was already small enough.

2.3 Summary of the Perforamnces

In the study of [23], training instances are obtained with windowing and overlapping procedures. For each distraction state, a randomly selected 25% of total instances were used for training. Four different distraction definitions classified the binary states of distraction. In this experiment, driving tasks including following the LV and responding to an LV break event, keeping the subject vehicle from drifting toward the lane boundaries

and driving in the center of the lane as much as possible, and the final task is to detect the appearance of a bicyclist on the right side of the road. IVIS tasks included tracking stock prices. Based on the tasks, the two definitions are DRIVE (IVIS or base drive) and STAGE (IVIS or not). Another two definitions are based on driving performance. Also, an SVM model and logistic models were constructed for each participant. Individual models were constructed for different feature combinations, window size-overlap combinations, and distraction states per participant, thus a total of 1548 SVM models and 1548 logistic models were constructed from the training data. Performance is obtained by averaging all the models for each machine learning model. In her research, individual subject analysis is utilized instead of non-subject analysis, thus her application is focused on detecting a particular driver's cognitive workload and can not accurately detect another driver's cognitive workload level. The study showed that a large window size would increase the models' accuracy and sensitivity, which suggests that using longer periods to summarize the data made the distraction signal easier for the models to detect. Also, the increased redundancy of input data between adjacent windows improved the performance. In her later work[28], she claimed that increasing the window size would decrease the number of training instances since the total quantity of the dataset would be fixed. In this case, the system performance would be undermined because there are fewer training instances. However, since the training dataset is randomly selected from total instances, and total instances are obtained with various windowing and overlapping values, the similarity between nearby instances is high, which can cause the testing dataset to be highly similar to the training dataset. In this case, the model is basically not trained at all. Also, the definition of cognitive workload is different from [23], and only the DRIVE stage is considered. The same training model is implemented, which includes the methods used to split the training dataset and testing dataset. In her later research, [24] also explained the top-down approach and bottom-up approach in the area of detecting drivers' cognitive workload. The top-bottom approaches require knowledge of the targets. However, the bottom-up approach can overcome this limitation and use data mining methods to extract characteristics of the targets. Data mining methods include decision trees, support vector machines and Bayesian networks. Numerical models

such as SVMs have fewer computational difficulties compared to graphical models such as BNs [24]. Also, when comparing Bayesian Networks with SVM, BNs that model time-dependent relationships between the driver's behavior and cognitive state produced the most accurate and sensitive models.

In the study of [21], the dataset is equally divided into two sets for each labeling class for evaluating purposes. In this way, a 2-fold cross-validation method is implemented with combinations with two sets from each labeling class. AdaBoost is used in this study. AdaBoost is a learning algorithm which creates different weak classifiers while successively changing the weighting of the learning data. . The final decision is made based on a weighted majority decision. Stump is used as the weak classifier. The number of weak classifiers was set to 1,000. Just as with the dataset used in [22], subjects are required to complement two different secondary tasks during the driving task. Arithmetic loads involved verbally subtracting prime numbers (for example, 7) from 1,000 successively. Conversation loads involved asking the subjects to describe a route which they regularly took (such as the route from school to home). The labeling of the dataset is a binary state with respect to each secondary task. For each secondary task, AdaBoost is applied and the performance is compared with the performance obtained with SVM with a Gaussian Kernel function. The authors claimed that AdaBoost is superior for the detection of driver distraction compared with the SVM since the average accuracy for cognitive distraction detection using visual information by AdaBoost was 84.6% in the arithmetic; furthermore, the F value was 83.1%. . The author in the study [22] did a further study on the consideration of the implementation time. The amount of calculation time for SVM to detect the driver's cognitive workload requires inner production calculations in the optimization process, which takes longer. However, AdaBoost only uses a threshold process and makes the decision based on the weighted majority decision. With this consideration, AdaBoost is faster compared to SVM, which makes real-time driver cognitive workload detection possible. This was proven by the comparison between the two implementation times in this study. In Miyaji et al's research, the windowing size and overlapping ratio are not mentioned [21, 22], and it is assumed that the author did not use the windowing and overlapping approaches. However, due to the noise, nearby samples can have larger

differences which can create difficulties during the training process

In the most similar study [19, 20] , n-back is treated as a secondary task. The baseline and three levels of the n-back task were completed for each subject. The three levels of n- back are 0-back, 1-back and 2-back. All the driving with the n-back task is labeled as positive cognitive distraction, and the baseline is labeled as the negative cognitive distraction. Three window sizes are used in this study: 10 s, 20 s and 30 s. No overlapping is considered here. Instead of randomly selecting the instance from the dataset, time splitting is used. Each window is treated as a segment, and for every training segment, a testing segment follows it. In this case, the testing dataset is always disjointed from the training data, and the performance evaluated through the cross-validation scheme reflects the actual generalization capability of the derived estimator. A neural network algorithm for estimating driver cognitive workload is utilized in this study. Radial basis probabilistic neural networks (RBPNN) are used. The performance obtained with only visual information can reach 83.3% with a window size equal to 30 seconds.

[2] used the same dataset as this study. Liu utilized an eye-tracking modality with gaze rotation and blink duration and frequency. Two data grouping techniques were utilized: subject-based grouping and time-based grouping. Time-based grouping can be thought of as using previously collected data to train a predictive model and then applying the model to new data generated from the same group of users. Subject-based grouping was built with data collected from training participants and tested with unseen users. The highest prediction accuracy for Ternary classes reached 49.6% with the AdaBoost algorithm. The same data partition methods were utilized in this study.

2.4 Data Fusion

Data fusion techniques are not implemented in this study due to time constraints. It is still worth to discussing data fusion techniques applied in the context of driving conditions. Robust system performance is expected to increase with data fusion techniques, and future works can explore this further.

Few research studies have attempted to detect driver drowsiness through the fusion of different methods [38]. The authors in [38] provided a way to get an overall standard score using scores obtained with various methods since the different independent measures are on different scales and each can be transformed to a standard score. [39] proposes a method for monitoring driver safety levels using a data fusion approach based on several discrete data types: eye features, bio-signal variations, in-vehicle temperature, and vehicle speed. Those data are collected from different sensors, including video, electrocardiography, photoplethysmography, temperature, and a three-axis accelerometer, that are assigned as input variables to an inference analysis framework. A Fuzzy Bayesian framework is designed to indicate the driver's capability level. Realistic testing of the system demonstrates the practical benefits of multiple features and their fusion in providing more authentic and effective driver safety monitoring. Even a few sensors [39] have demonstrated better performance, and with more sensors and data collected with regard to physiology, performance and behavioral measures, it is assumed that the performance is going to be even more robust.

Studies that applied data fusion methods for monitoring drivers' cognitive workload are discussed below.

Data fusion techniques can be classified based on the relationship between the data sources[40].

- complementary: When the information provided by the input source represents different parts of the scene and could thus be used to obtain more complete global information. For example, in the case of visual sensor networks, information on the same target provided by two cameras with different fields of view is considered complementary.

- redundant: when two or more input sources provide information about the same target and could thus be fused to increase the confidence. For example, data coming from overlapping areas in visual sensor networks are considered redundant.

- cooperative: When the information provided is combined into new information that is typically more complex than the original information. For example, multimodal

(eye-tracking measures and video data) data fusion is considered cooperative.

2.4.1 Complementary Inputs Fusion

Some researchers only use one sensor or similar sensors with respect to the same type of measures and fusion technology is applied. In these situations, the data fusion techniques can be categorized as complementary. They have shown increased performance compared to individual signals.

[41] [40] proposed using cameras and active infrared illuminators to acquire video images of the driver. The visual features include eyelid movement, gaze movement, head movement, and facial expressions. A probabilistic model is developed to model human fatigue and to predict fatigue based on the visual features obtained. The simultaneous use of multiple features and their systematic combination yields a much more robust and accurate fatigue characterization than using a single visual feature.

[42] [40] proposed a deep learning approach for driver activity anticipation in sensory-rich robotics applications. Two cameras are used in this project with one facing the driver and another one recording the external environment. The proposed architecture first passed sensory streams independently through separate Recurrent Neural Networks (RNNs). The high-level representations from all RNNS are then concatenated and passed through a fully connected layer that fuses all representations.

2.4.2 Cooperative Inputs Fusion

To get better performance, multi-sensors with multi-type of measures mentioned in the Background section are used, and the information fusion based on the signals collected are applied.

[39] proposes a method for monitoring driver safety levels using a data fusion approach based on several discrete data types: eye features, bio-signal variations, in-vehicle temperature, and vehicle speed. Those data are collected from different sensors, including video, electrocardiography, photoplethysmography, temperature, and a three-axis accelerometer, which are assigned as input variables to an inference analysis framework. A Fuzzy

Bayesian framework is designed to indicate the driver's capability level. Realistic testing of the system demonstrates the practical benefits of multiple features and their fusion in providing more authentic and effective driver safety monitoring.

2.4.3 Summary on Data Fusion Techniques

Data fusion techniques and an architecture's performance highly depend on the data set. The architectures can be classified as Decentralized architecture, Distributed architecture and Centralized architecture[40]. Centralized architecture collects data from all sensors and then fuses them for analysis, thus it has high requirements from the processor, and it produces delays when transferring the information between the different sources and affects the results. For decentralized architecture, each sensor has processing capabilities, and there is no single data fusion. That means data fusion is performed with local information and information received from its peers. The disadvantage of this is that the computation complexity is relatively large. In a distributed architecture, the sensor information is processed independently and then sent to the fusion node, which means the information is already analyzed based on its local vie and then fused with other sensor information for a fused global view.

The available data fusion techniques can be classified into three non-exclusive categories: data association, state estimation and decision fusion[40]. Data association is usually performed before state estimation of the detected targets. The classification results highly depend on the data association phase, which refers to establishment of the set of observations by some target over time. State estimation is referred to as classification or target tracking. The state estimation is also can be treated as cognitive workload indicator. It includes methods of maximum likelihood, maximum posterior, the Kalman filter [43], particle filter, distributed Kalman filter and covariance consistency methods. Other machine learning methods are also used for data association and state estimation. SVM [23], Fuzzy classifier[44] and RNNs[42] are options associated with data fusion methods.

Chapter 3

eDREAM Eye-tracking and Physiological Modalities

3.1 eDREAM Experiment

The eDREAM dataset utilized in the experiment was collected by Liu and Dengbo[1].It was collected using various sensory and visual signals when participants experienced three different levels of cognitive load during driving. The sensory signals are categorized into four types: **a)** Vehicle-Based Measures, **b)** Physiological Measures(ECG, Galvanic Skin Response, Respiration) and **c)** EEG Measures, **d)** Video and Eye Tracking Measures. Each cognitive workload level was presented on a separate drive: the lowest level was driving with no added secondary task, which is an auditory-recall n-back task described in [2]. The median or high levels were imposed using an n-back task with a factor of 1 or 2, respectively. The primary driving task is conducted using a driving simulator called the NADS miniSIM. The primary and secondary task together induce levels of cognitive workload. Since the experiment is conducted in a simulator which can control the driving scenarios and limit the external conditions difference, the only external variable it could affect participants' cognitive workload would be the n-back factor. However, subjects' differences need to be taken into consideration to isolate responses induced by the increased cognitive workload. Details on the data collection campaign can be found in [2]. The following sections will give a summary of the dataset and then discuss details about

the Physiology and Eye-tracking modalities.

3.1.1 n-back Secondary Task

N-back is the secondary task utilized here to introduce cognitive workload. Compared to other secondary tasks such as detecting pedestrians, the n-back task is purely focused on cognitive workload without introducing visual distractions. This is most important for measures involving eye-tracking techniques. In this experiment, a modified n-back task, an auditory recall task with letter stimuli, was used to model drivers' cognitive workload level. n is the number of letters the participant needed to store in working memory and they had to recall how many n-backs existed in the ten consequent letters. Traditionally, a numerical number has been used instead of an alpha value; however, the participant needed to answer with a numerical number in the end, which could affect the participant' performance. Thus, letter stimuli were utilized; for example, giving 10 letters.

$$ABCDCDDFAF$$

n-backs exist if the current letter is the same as n steps ago. **D D** is 1-back and **D C D** is 2-back since D repeats itself with a step in between. Participants need to answer how many n-backs exist in the 10 letters.

The n in the n-back task is a load factor to increment the cognitive workload systematically. As is mentioned in Chapter 2, cognitive workload introduced with a secondary task can be measured with three methods: subject measures, performance measures, and psycho-physiological measures. In this study, only psycho-physiological measures were analyzed to detect drivers' cognitive workload level modelled with the n-back task.

In this study, three load factors were introduced (no task, 1-back and 2-back). Participants were trained in non-driving conditions first. Example tasks were given before the data collection procedure to ensure participants understood and were capable of performing this task. During the data collection procedure, the participant was informed the task detail ahead of time, thus the participant could get ready for the secondary task. Then, ten audio letters were presented, and the participant needed to say how

many n-back occurrences there were. Every ten letters, n-backs were presented as the Focus Period. 6 Focus Periods existed in one drive, but only 4 of them were analyzed, since another 2 involved another event which would affect the cognitive workload.

3.1.2 Participants

A total of 37 healthy, gender-balanced participants were involved in the data collection campaign. Several requirements needed to be satisfied for each participant to minimise the external factors which would affect the data collected.

- Hold full driver's license for at least three years
- Under 35 years old
- No Vision-Correction Glasses(contacts allowed)

Within those 37 participants, 18 of them were female and the mean age of them was around 27 years old with four years of variance. The second requirement minimized the age variation effect of the function of working memory, which is a mental resource. Also, the eye-tracker would function better with no reflection from glasses.

3.1.3 Cognitive Workload Labeling

Based on the data collection campaign in [2], two Critical Audio Sections are involved in each drive. Each drive is approximately 5-10 minutes in total, and each Critical Audio Section lasts for 2 minutes. A break of about 45 seconds is provided in between Critical Audio sections. Critical Audio Sections start with an audio introduction of the n-back task, followed by three Focus Periods. In each Focus Period, participants will hear ten letters presented at a rate of 2.25 seconds per letter and will speak the result of the n-back task after. The total time of each Focus Period is 25 seconds. Each participant completed 6 Focus Periods for each drive as shown in Figure 3.1. Among 6 Focus Periods, 2 of them involved a leading vehicle braking event. The design of the leading vehicle braking event would be beneficial when analyzing the response time of participants when under variant

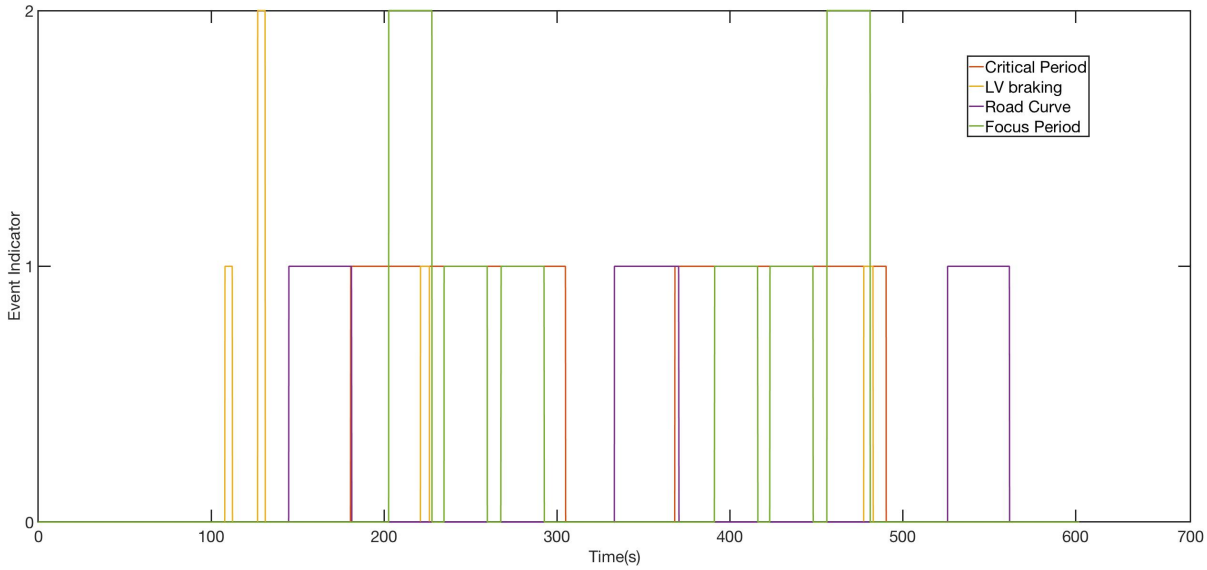


Figure 3.1: Experiment conditions recorded in meta-data files from a single drive. For Focus Period, value of 2 indicated that it accompanied with leading vehicle braking event, and 1 indicated nBack task. For Lead Vehicle Braking, 1 indicate that lead vehicle is braking and 2 indicate it is a heavy braking

levels of cognitive workload. However, this is not considered in this experiment and it might affect the driver's cognitive workload state. Thus, only 4 of them were selected and labelled based on the n-back task involved during the data collection procedure.

3.2 Eye-tracking Data

3.2.1 Hardware Apparatus

Eye-tracking data collection is performed via a commercial remote eye-tracker system developed by Seeing Machines called faceLAB 5.0. The sampling frequency of faceLAB is 60 Hz. Data collected with the eye-tracker were stored locally in a standalone computer that runs the eye-tracking software, and part of the data was forwarded to the driving simulator miniSIM computer and stored in the miniSIM log file. An active Near-Infrared (NIR) LED and two NIR cameras were used to acquire high-quality images that would be invariant to changes in illumination. As shown in Figure 3.2[1], the yellow-marked cameras are mounted at the center of the dashboard. With this setting, those two cameras are facing the frontal view of the driver to capture accurate results. The angle of the

cameras was adjusted to capture the frontal face of participants. The head model and screen model were always re-calibrated in regard to the sitting position of the participant to ensure the data quality. FaceLAB tracks facial feature points to create customized 3D

Figure 3.2: Camera and eye-tracker placements in the driving simulator.Image taken from [1]

head models, which were calibrated for each participant. Also, before the experiment, the gaze intersection was calibrated using a dot-chasing process. During the process, participants were asked to follow a dot that would be located at four corners and edges of the main plane. With this gaze tracking system, the estimated gaze locations would be calibrated with the actual dot locations. It should be noted that the eye-tracking system uses a world-wide reference point, which is the center of those two cameras. As shown in Figure 3.3[1], the midpoint between two cameras is defined as the origin of the world coordinate in faceLAB. Three screens and one dashboard were used during the data collection experiment, and faceLAB can distinguish which display participants

were looked at by checking the worldwide gaze coordinate. Also, another application,

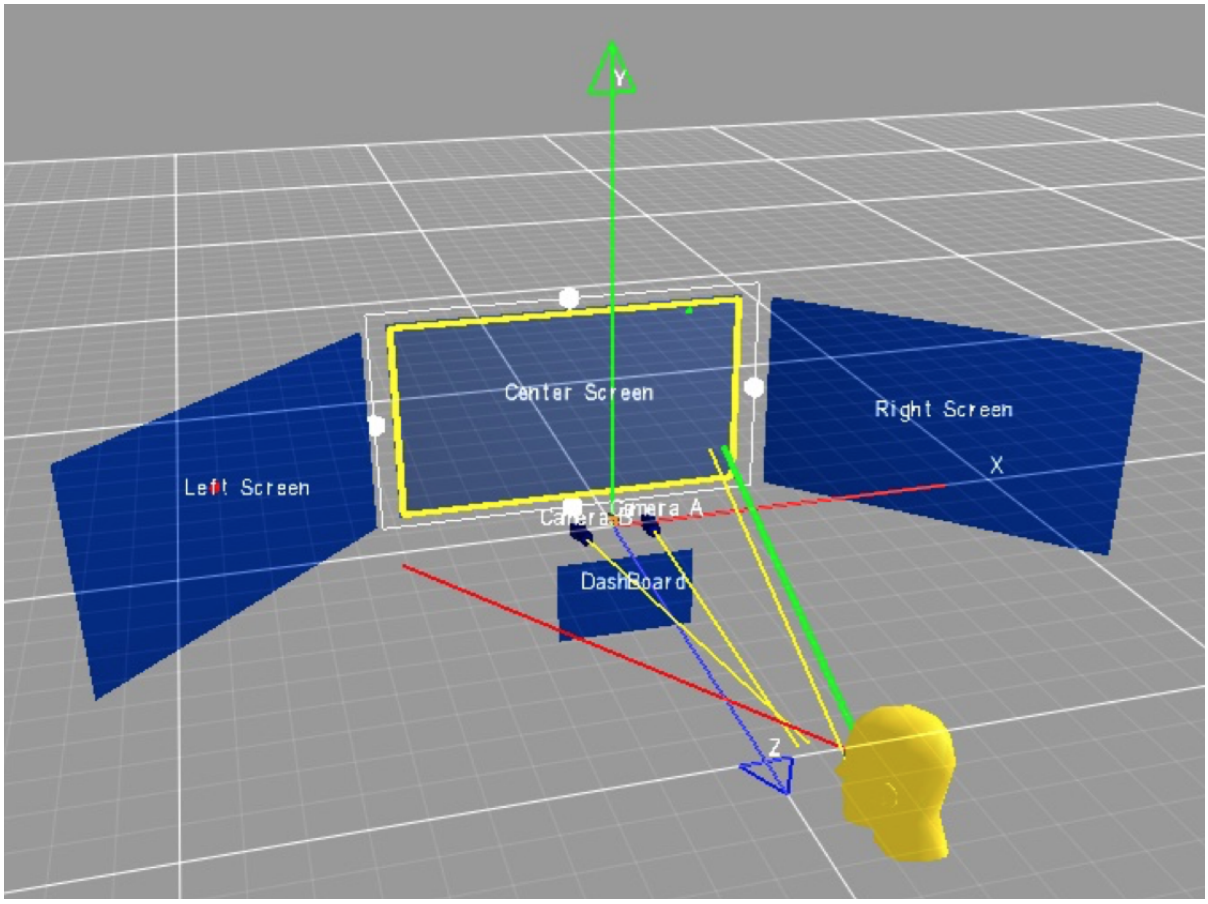


Figure 3.3: Customized 3D visual environment used in eDREAM dataset collection. The X, Y and Z axis denoted in red, green and blue arrows. The origin of the world coordinate is the midpoint between two faceLAB cameras. Image taken from [2].

EyeWorks, grabs image frames from miniSim and overlays the tracked gaze position on it, producing an intuitive visual record of where the eye-tracker estimates the person is looking within the central screen. The output obtained with the eye-tracker is higher-level features (e.g. blinks or saccade detection) and are overlaid on the images captured with the cameras to provide the overall performance. An example screenshot is provided in Figure 3.4[1].

3.2.2 Eye-tracking Data Description

The output obtained with the faceLAB eye-tracker contains five output files based on the official "Output Data Reference Guide".

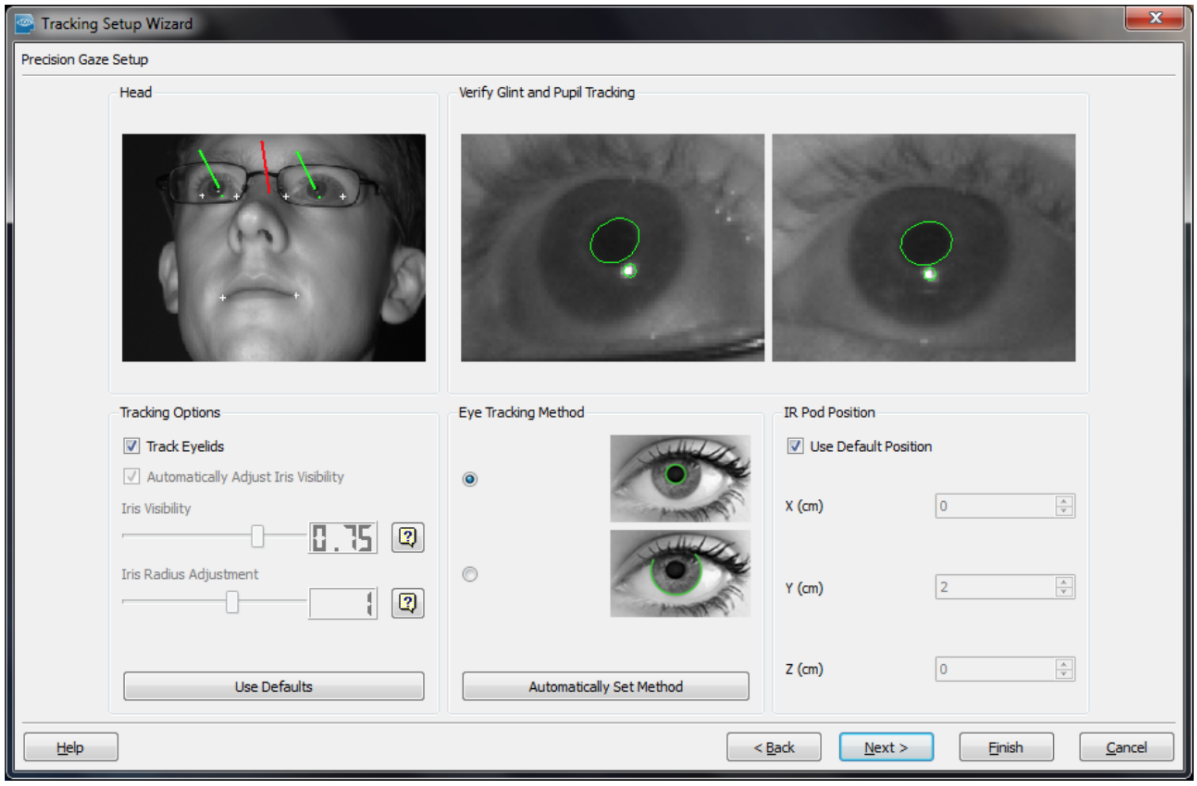


Figure 3.4: Example of the EyeWorks application which overlays tracking results on the image. Gaze direction was presented in green and head direction was presented in red. The green circle showed the pupils location and size. Image taken from [1].

- Time Output Data: Timing information related to the experiment.
- Head Output Data: Head position, rotation and other relative information
- Eye Output Data: Eye states information, blinking frequency and others
- Facial Feature Output Data: 3D coordinates with respect to the head-reference frame
- World Output Data: Gaze intersection for both eyes (plate or world coordinate)

Those output data were indexed with the frame number generated with a faceLAB eye-tracker instead of using the Time Output Data in this experiment. The output of the eye-tracker is high-level features such as blinking frequency and gaze intersections. Since the eye-tracker is commercial equipment, hidden details on how those features are calculated are not explained in the official user manual. The output variables such as blinking and

Table 3.1: Example of FaceLab variable and description

File Label	Variable	Description
Head	Frame number	Facelab frame number
	Head position(x,y,z)	Head position vector in the world coordinate system in meters
Eye	Blinking	1 means blink is occurring and 0 otherwise
	Gaze rotation(of right and left eyes, w.r.t.x and y axis)	The orientation of the eye gaze w.r.t. the world coordinate
WorldV2	Faze intersection in World Coordinates (x,y,z for right,left and both eyes)	Label of the world model item the gaze vector intersects with, which could be "Center$_{S}creen$", "Left$_{S}creen$", "Right$_{S}creen$", "DashBoard", or "Nothing"

gaze intersection are separate for the left eye, right eye and vergence. Those values do not always maintain the same pattern for most of the participants. Some participants may present a more active left eye compared to the right eye. In this experiment, only Eye Output Data and World Output Data are utilize. The number of variables output from the eye-tracker is 280 without counting the Time Output Data, which is not possible to list here. Only several measures are listed in Table 3.1

3.2.3 Time Synchronization

As is mentioned in Section 3.2.1, part of faceLAB was forwarded to the driving simulator miniSIM computer and stored in the miniSIM log file. The frame number of the eye-tracker is also stored in the miniSIM driving simulator and stored in the DAQ files. In this way, the faceLAB simulator data are synced with the miniSIM data. However, this is not very stable at the beginning of the data recording. This could be caused by the initialization process of miniSIM, as it starts the log file before receiving eye-tracking

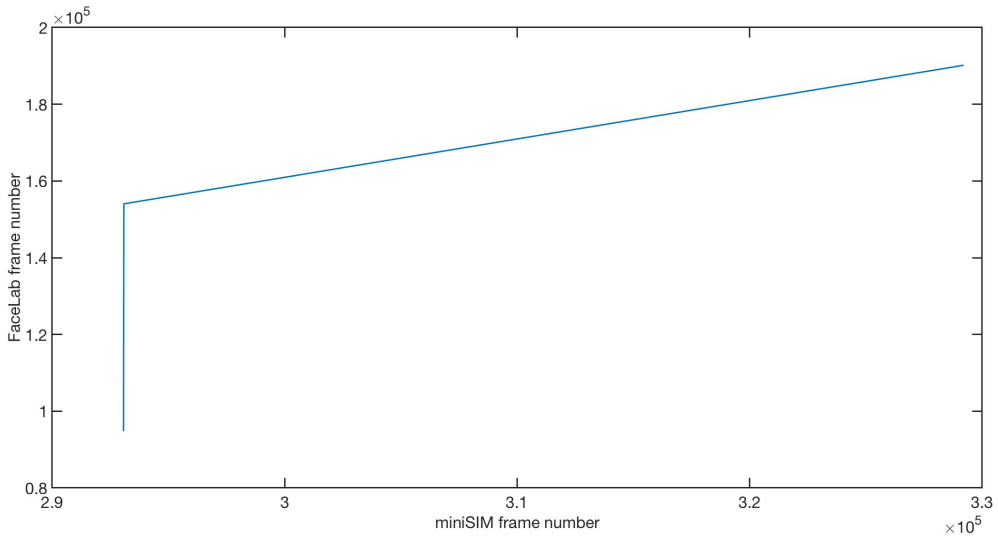


Figure 3.5: Example of the FaceLab frame number that was received by miniSIM This is the data for Participant 34's 2-back drive. Notice how the FaceLab frame number takes a jump near the beginning of the recording, it then becomes stabilized and increased steadily

information. Thus, only the data collected after the syncing was stabilized are utilized in this experiment. Figure 3.5, shows that the synchronization is not stable at the beginning of the log file.

When data collection champing was designed, the Focus Periods were allocated in each driver, which are time slots for placing n-back recordings. Also, the Focus Periods were indexed with the miniSIM frame number. With the miniSIM frame number, the corresponding eye-tracking recording can be extracted and synchronized. 3.6 shows the synchronized data for left eye rotation at the x-axis. Most important is that the sampling frequency of the eye-tracker is the same as miniSIM's, thus no further processing is needed. For physiological measures, it is more complicated, which will be discussed later.

3.2.4 Data Exploration

The Time Output Data, Head Output Data and Facial Feature Output Data were not explored and analyzed here due to the huge amount of information and no domain knowledge related to those fields. Some of the participants were excluded from this experiment

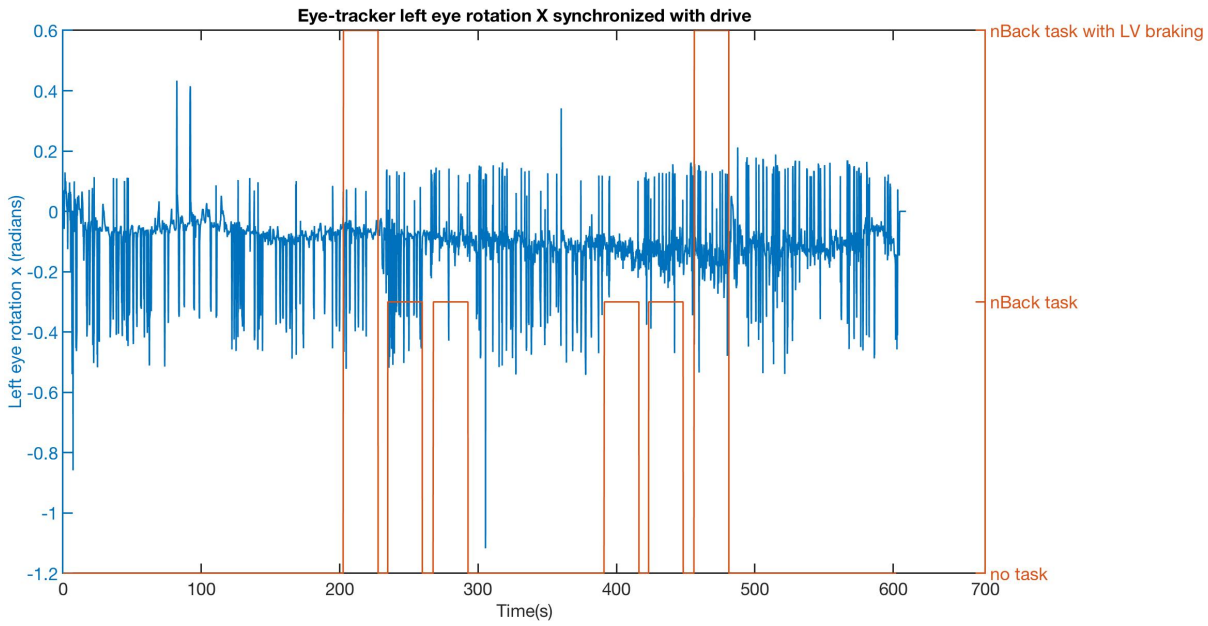
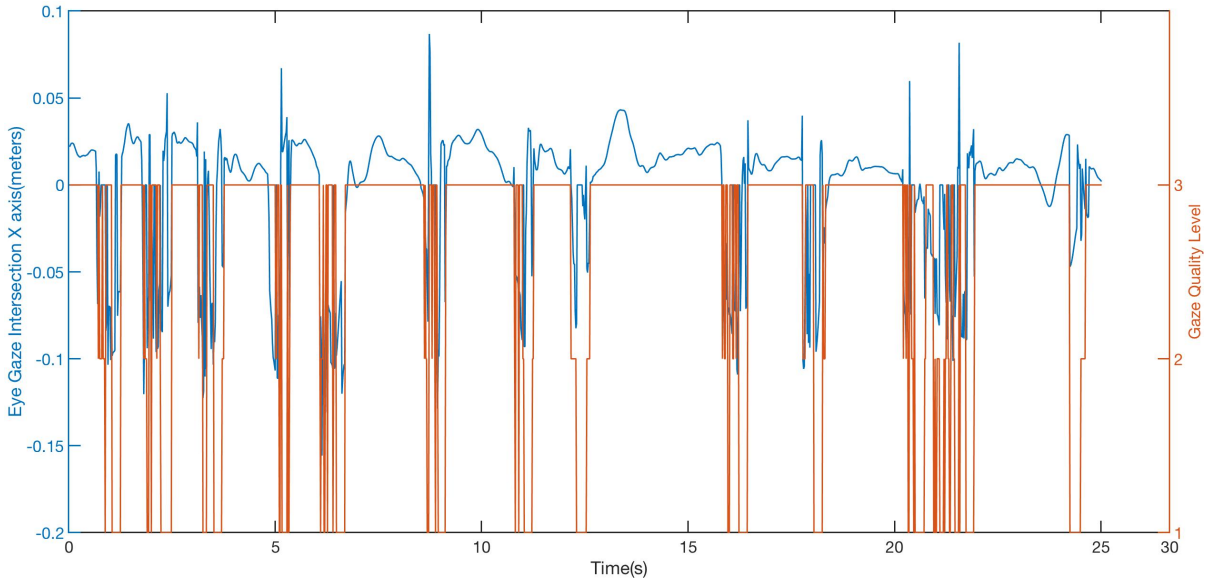


Figure 3.6: Eye tracking data synchronized with Drive Labels
The left eye rotation radians at x axis for participant 34 was presented. The red label is the Drive Labels where LV stands for Leading Vehicle

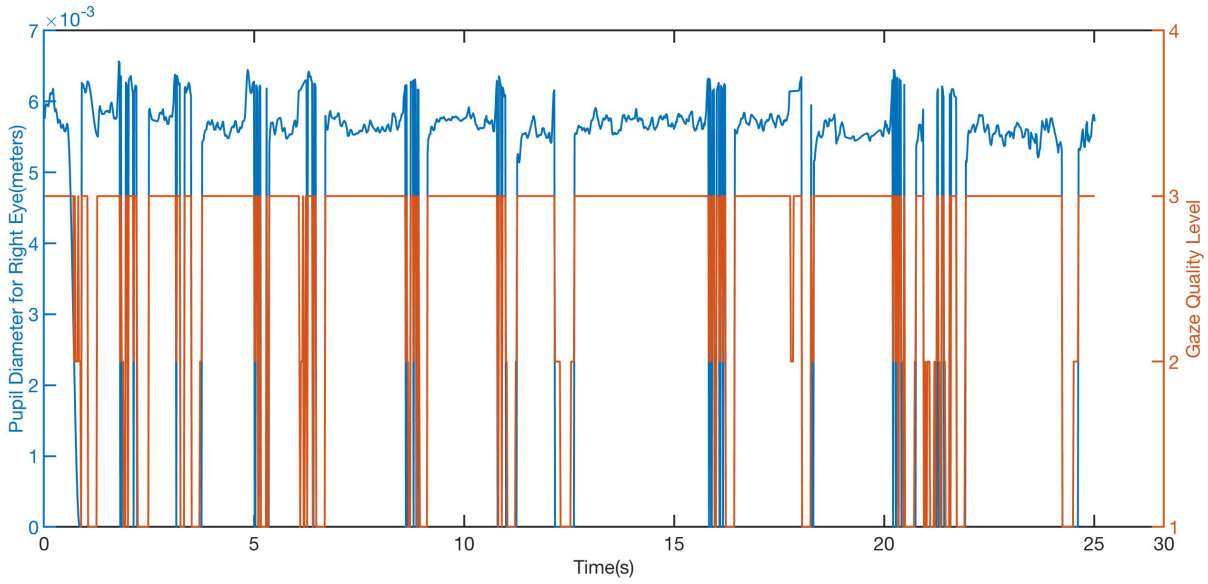
due to problems experienced during the data collection process such as falling asleep. Four datasets and four Focus Periods are extracted and visualized to understand how each measures the response to the different level of the n-back task analyzed.

Artifact Information

In this part, how eye-tracking data are presented after the time synchronization is examined. Eye-tracking results are highly related to artifacts such as eye blinks and head movements. When a participant's head moved, the eye-tracker needed to quickly identify the gaze direction based on the head-facing angle and eyeball rotation angle. When the movement is quick, the eye tracker may fail to detect the accurate gaze direction. Figure 3.7 shows the gaze intersection of the x-axis and the pupil diameter results of the right eye during the first Focus Period. The red line is the gaze quality level, which indicates the accuracy of the gaze direction. When the gaze quality level is one it means the gaze direction is the same as the head direction. It should be noted that when the gaze quality level is equal to 1, the gaze intersection value has a significant variant and could be accompanied with the movement of the head of the participant.



(a)



(b)

Figure 3.7: A set of four subfigures: (a) Gaze Intersection X-axis; (b) Pupil Diameter of Right Eye;

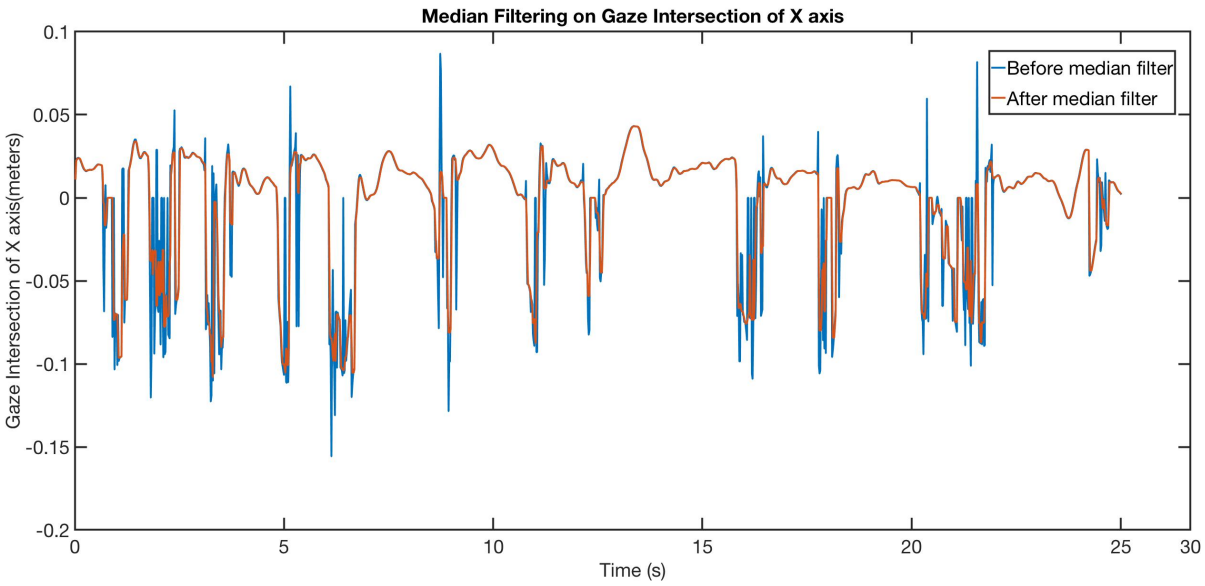


Figure 3.8: Comparison between gaze intersection of X-axis with median filter After median filter, the noise caused by head movement decreased. It also keeps the information when the gaze quality is high.

Noise removal techniques need to be take into consideration before exploring the dataset. Outliers caused by head movements and blinking existed in this dataset. Compared to the noise removal technique median filter, averaging method is more bias to those outliers. Thus, a median filter with a 0.1-second window size was selected here. 3.8 shows the results after the median filter. After applying the median filter, noise caused by head movements decreased. Most important is that the median filter did not miss any information when the gaze quality level was high.

Time Domain Static Features

The output of the eye tracker are time domain signals with a sampling frequency of 60 Hz. The value of one sample does not contain much information to determine the cognitive workload of drivers. Even though the output of eye tracker are high level features such as gaze direction and pupil diameter, more advanced features are still needed to present the proprieties of those time domain signals. Traditional static summarizing methods could be used to extract the proprieties of those time domain signals.

In this experiment, static summarizing methods such as mean value, standard devi-

ation value and root mean square value were selected. A sliding window was used while applying those functions. The sliding window size utilized here is 10 seconds with 90% overlapping. This also means that every 1 second, one summarization value was calculated and extracted. Later, in the analyzing section, the effect of the variables of sliding window size and overlapping ratio will be discussed in Chapter 4.

The static presentation of the gaze intersection of the y-axis of participant 34 at the 2-back task is shown in Figure 3.9. It can be easily observed that:

- With no static summarizing, the gaze intersection of the y-axis does not show the ability to discriminate different levels of the n-back task.
- A Clear trend shown with std and RMS static methods that the median value of those features decreased when the level of cognitive workload level increased.
- No clear trend was observed with the mean static method, but it shows the potential to discriminate different levels of task load compared to no processed gaze intersection.

This was also observed with other output measures of the eye-tracker. It can be shown that those static summarization methods can help to discriminate the different cognitive workload levels.

3.3 Physiological Data

3.3.1 Hardware Apparatus

ECG, GSR, and respiration sensors by Becker Meditec collected data at 240 Hz using D-Lab software developed by Ergoneers. Solid gel foam electrodes were used for ECG and GSR sensors. In this experiment, the ECG electrodes were placed on the participant's body to record heart electrical activity. One electrode was placed on the neck over the vertebra, one placed on the left side of the rib cage over the second lowest rib, and one placed over the uppermost part of the center-line of the rib cage. Electrical changes on the skin were detected by these electrodes.

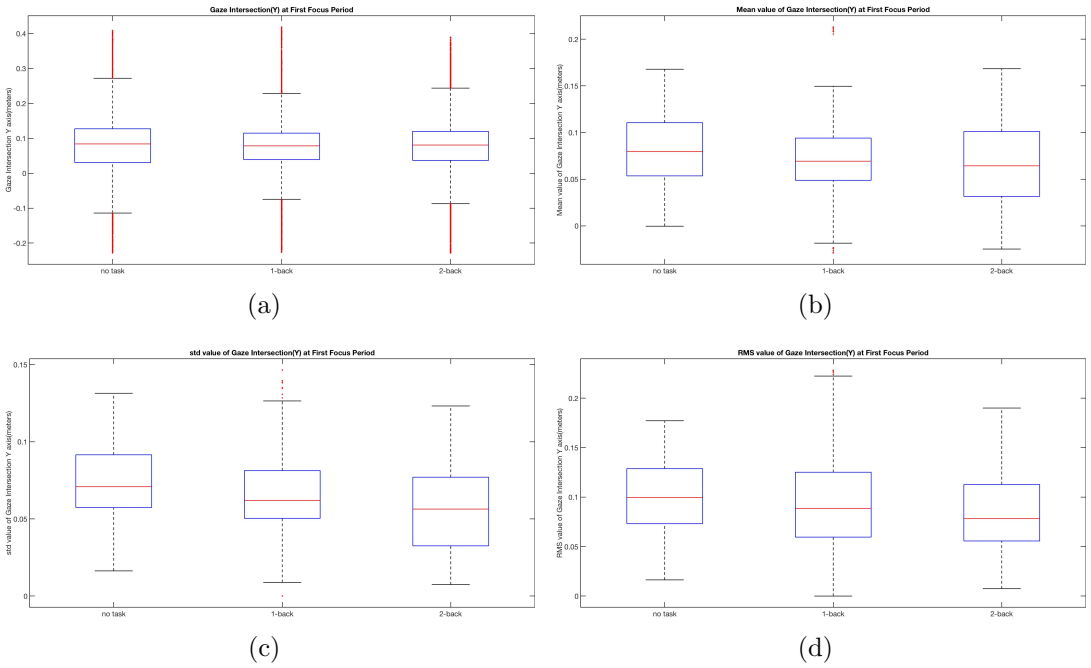


Figure 3.9: A set of four subfigures: (a) Gaze Intersection of Y axis; (b) Mean of Gaze Intersection of Y axis; (c) std of Gaze Intersection of Y axis; and, (d) RMS of Gaze Intersection of Y axis.

A respiration belt was worn around the chest or abdomen of the participant. When participants breathed, the belt detected the stretching caused by the breathing. The belt is an elastic belt which contains wires that change resistance as it stretches and relaxes. With this belt, the breathing patterns during the driving sessions can be recorded and measured by the resistance value. The natural stretch length for each participant was measured and initialed by pressing the reset button for at least 2 seconds. For some participants, the belt may be cut and adjusted for a better fit.

Sweat gland activity was monitored with a GSR Amplifier , which can measure changes in the electrical properties of the skin. The GSR sensors were attached to the bare left foot—one in the middle and another under the heel. GSR measures psychological arousal since the sweating was controlled by the sympathetic nervous system.

3.3.2 Physiological Data Description

The output obtained with D-Lab software organized all the outputs from ECG, GSR and respiration sensors and stored them in a single text file for each drive (no task, 1-back

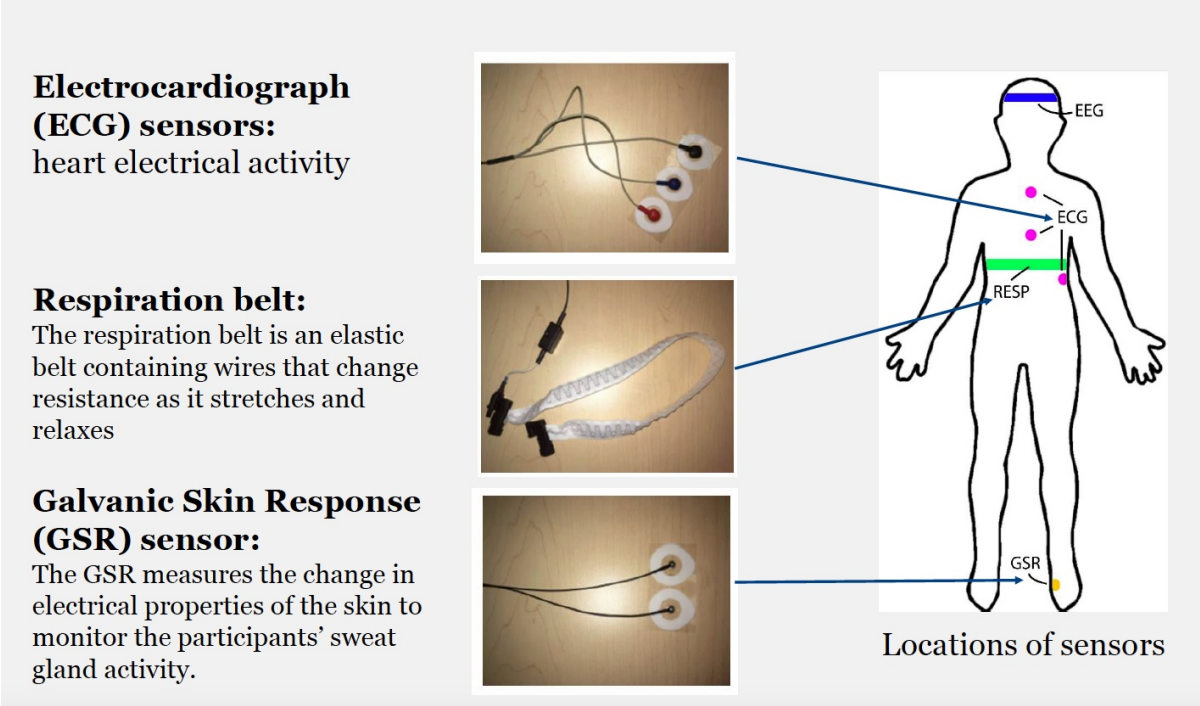


Figure 3.10: Physiological Sensors and their locations

and 2-back). The output obtained with DLab contains four output parts for each text file.

- miniSIM log information: information related to the driving scenario
- ECG sensor data: ECG measurements, heart rate and body accessories.
- GSR sensor data: changes in electrical properties of the skin to monitor the participants' sweat gland activity
- Respiration belt sensor: participants' respiration rate

The physiological measures were collected with DLAB software and synchronized with miniSim timestamps. However, due to the communication between different sensors (driving simulator, ECG and GSR sensors) and also the sampling frequency difference, the DLAB-collected data were not clean data. miniSIM data are recorded with sampling frequency 60 Hz and physiological measurements were collected with a sampling frequency of 240 Hz. The DLAB original text file is shown in Figure 3.11. It contains two parts (miniSIM log information and physiological signals), and they are synchronized with the

record time. It should be noted that miniSim log information contains repeat entries (exactly the same miniSIM frame number) and the physiological signals part has empty entries. Table 3.2 shows some variables recorded in the file.

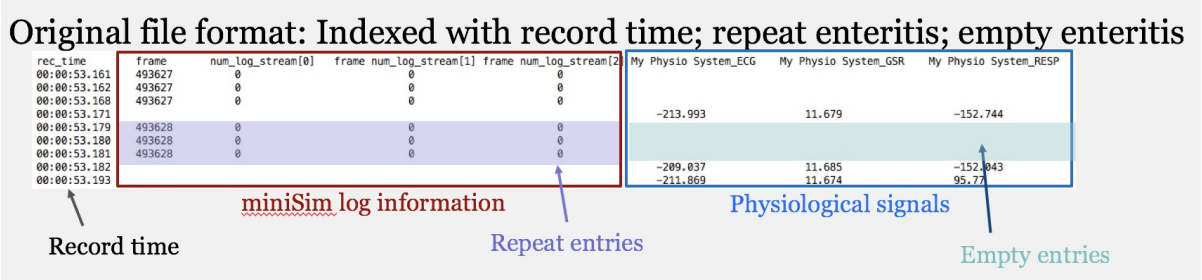


Figure 3.11: Original file format recorded with DLAB software

Table 3.2: D-Lab variables and description

Col. No.	Variable	Frequency	Description
1	MiniSim Frame Number	60Hz	Relayed from MiniSim
2	During Curve or Not	60Hz	Relayed from MiniSim, SCC_LogStream: 1: during curve; 0: during straight route
3	Audio States (n-back test)	60Hz	Relayed from MiniSim, SCC_LogStream: jumping from 0 to 1, start of the audio
4	Leading Vehicle Braking States	60Hz	Relayed from MiniSim, SCC_LogStream: jumping from 0 to 1, start of the leading vehicle brake
5	Headway	60Hz	Relayed from MiniSim, SCC_FollowInfo: unit: feet
6	ECG	240Hz	Electrical activity of the heart
7	GSR	240Hz	Electrical characteristics of the skin, unit: uSiemens
8	Respiration	240Hz	Shape change of torso while breathing
9	Heart Rate	240Hz	Number of heart beat per second

As is shown in Table 3.2, the heart rate was already extracted from ECG measures. Due to the missing knowledge on the sensors and the Dlab software used during the data collection campaign, there is no detailed information about how the heart rate was obtained from the ECG.

Those output data were indexed with the miniSIM time stamp generated with miniSIM. The first four variables are extracted from miniSIM log files to capture the event states such as the Audio States. However, this only includes the Critical Period, which

is indicated by the Audio States, and the Focus Period is not indicated in this text file. Thus, the data collected during the Focus Periods cannot be allocated directly. It also should be noted that the physiological sensors' sampling frequency is different from the sampling frequency of the miniSIM driving simulator. This needs to be taken into consideration during the data synchronization procedure. The sampling frequency of physiological is four times that of the miniSIM driving simulator. However, the miniSIM log file is processed and stored on one computer, and Dlab software was processed on another computer. Ethernet connections were used for communications between two computers. During the data transferring procedure, information packages were not delivered synchronized, thus the physiological measurements and miniSIM log information are not exactly synchronized and physiological measurements has four times of sampling frequency of miniSIM log information.

3.3.3 Time Synchronization

As is mentioned above, when synchronizing the DLab output and miniSIM log files, data transferring caused a matching problem since they were indexed with a time stamp. Also, the Dlab text file does not have the index of Focus Periods, which creates difficulties in allocating and extracting information. To synchronize the Physio- logical data, two stages are utilized in this experiment. This synchronization is based on miniSIM frame number.

For the first stage, the no pattern, not clean text file needs to be processed. Empty entries are all the empty space existed in the physiological signals part, and repeat entries are the repeat rows in the miniSIM log information part, for which it can be noticed that their frame numbers are identical.

- All the empty entries were filled with the previous non-empty entries
- Rows that are empty were removed.
- Rows with duplicated miniSIM log information were removed.

With such a process, the miniSim frame number is unique and each frame is corresponding to a set of physiological signals. The converted file format which is indexed with a miniSIM frame number is shown in Figure 3.12. Figure 3.13 shows the synchronized GSR value for Participant 34, 2-back drive. Note: P18, P36 and no task trail for P37, no physiology data were recorded in the original files.

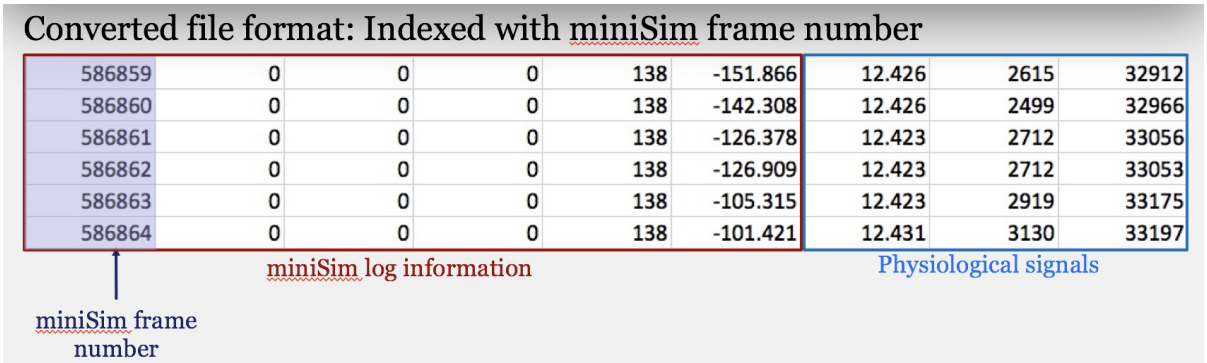


Figure 3.12: Converted file format after data cleaning

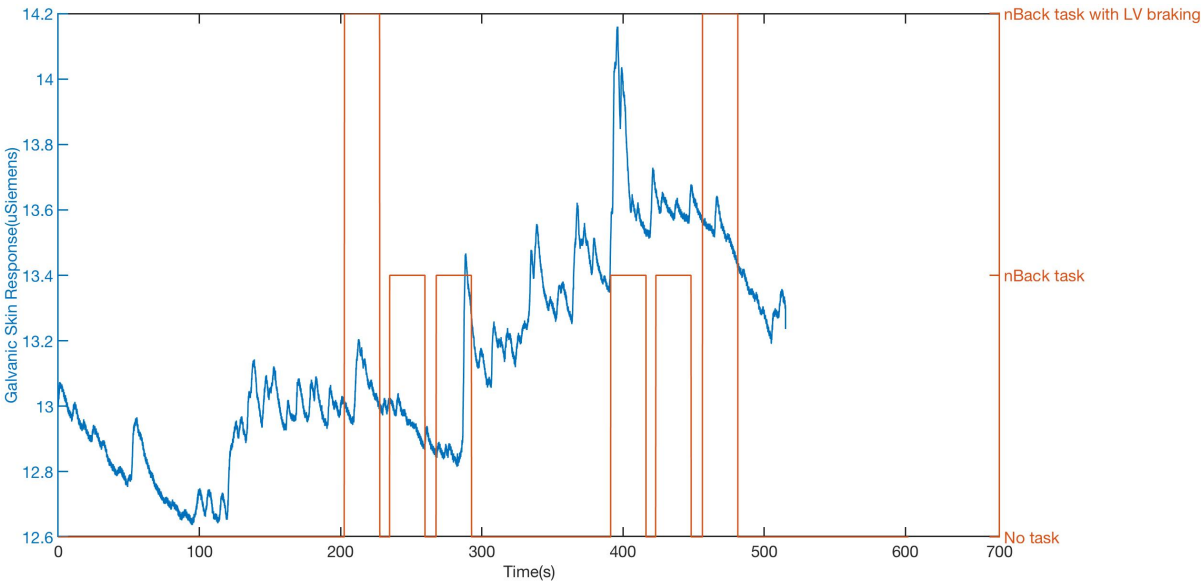


Figure 3.13: Physiological data synchronized with Drive Labels
The GSR value for participant 34 was presented. The red label is the Drive Labels where LV stands for Leading Vehicle

Chapter 4

Cognitive Workload Estimation Model

With the data collected with the eye-tracking modality and physiology modality from the eDREAM dataset, we explored eye-tracking measurements and physiology measurements as those two modalities can be utilized to build a driver cognitive load estimation system. The experiment presented in this chapter will first discuss the overview of the system and details on each modality will be given. By applying supervised classification methodologies to labelled data of individual modalities from the eDREAM dataset, drivers' cognitive workload level can be estimated. In this study, a data-driven experiment is performed with machine learning classification approaches with each modality. By using the labeled eDREAM dataset, features selection is implemented at raw sensor data to extract the most relevant features, then supervised classification methodologies that Support Vector Ma- chines (SVM) are selected for constructing the predictive classification model. The predictive classification model is categorized into three different models based on the purpose of the application and it would be easy to do compression with the previous study conducted by Liu[2]. Also, with the assumption that the driver's cognitive workload level varies with respect to the increasing of driving time for each driving routine, more analysis is conducted with the time gap that exists between training instance and testing instance.

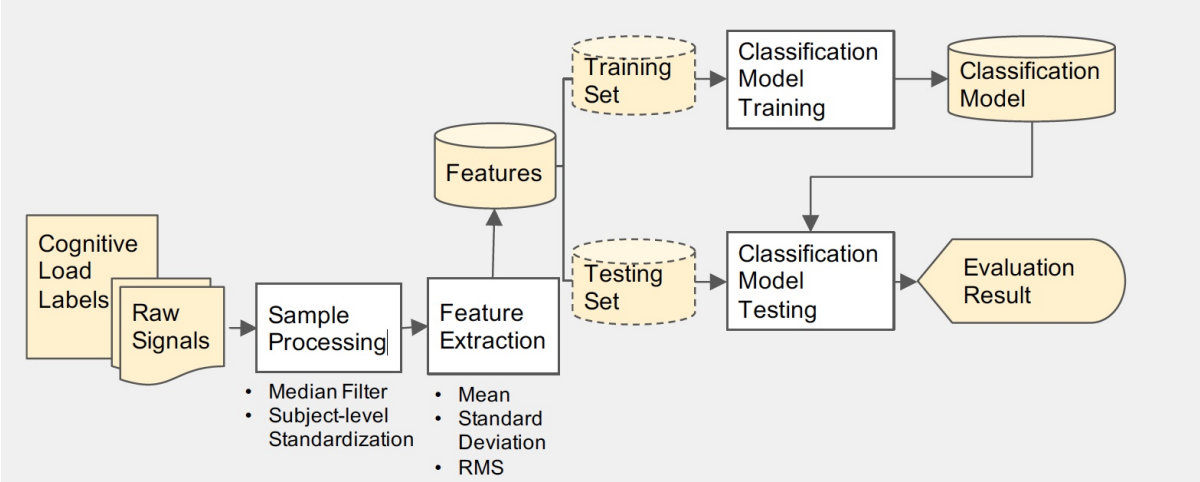


Figure 4.1: Experiment overview

4.1 Experiment Overview

The entire experiment was divided into two parts. Eye-tracking measures and physiological measures are implemented individually to build systems for detecting drivers' cognitive workload. Details about the data and how they are labelled are provided in Chapter 3. Raw data output from FaceLab sensors first go through the data selection step, and then a median filter is used to remove the noise and outliers. Finally, for each measurement, the mean value, standard deviation and root mean square are extracted and treated as features of the classification system. Due to the size of the feature numbers extracted from Face- Lab, the feature reduction method Principal Component Analysis (PCA) is utilized before sending to the classifier. In this experiment, the Support Vector Machines algorithm is selected as the classification method to use to detect drivers' cognitive workload level. The same procedure is conducted with physiological measures except for the PCA, with the consideration that the feature number is already small for physiological measures.

4.2 Data Pre-Processing

Data prepossessing is an important step in a machine learning project. The phrase "garbage in, garbage out" is particularly applicable to machine learning projects. In

this project, it is especially important since noise is caused by sensor connections, body movements and inaccurately labeled data might cause misleading results. Thus, sensor measurements go through three pre-processing stages to remove dirty data.

4.2.1 Data Selection

With the analysis of the experimental data collection notes and observation of the measurements across all 37 users, 29 out of the 37 users (for physiological modality, 28 for eye-tracking modality) are chosen for the analysis. Users dropped are excluded due to problems experienced during the data collection process. They had problems:

1. Missing eye-tracking data because of setup failure.

2. Self-reporting a lower workload though NASA-TLX as the level of cognitive task increased.

3. Exhibiting obvious fatigue symptoms during the data collection experiment, such s drooping eye-lids and drifting away from the lane.

4. Missing the logging information or missing the physiology information during the data collection experiment.

As described in Chapter 3, only data from the Focus Periods with no-task, 1-back task and 2-back task would be extracted and labeled with low, medium and high cognitive loads, respectively, in this experiment. The rest of the dataset are not considered in this experiment. Also, as described in section 3.1.3, there are three Focus Periods in each critical period and two critical periods per drive, thus there are total six Focus Periods in each driving. Two out of the six Focus Periods has involved with leading vehicle braking events, which could pose significantly affect the participants, and could override the effect of cognitive load. With excluding the two Focus Periods which involved with leading vehicle braking events per each driving, total four Focus Periods were analyzed in this study.

4.2.2 Sample Processing

Smoothing

To mitigate the effects of noise artifacts and inter-user differences, it is recommended to perform sample processing and subjective normalization. Frequency domain filtering is a standard technique used to remove the noise in signals collected from sensors. How- ever, due to the lack of domain knowledge and a complex parameters selection procedure, a median filter is utilized in this experiment based on the benefits of having fewer parameters. Smoothing always involves some form of local averaging of data. The most common technique is moving average smoothing, which replaces each element of the series by either the simple or weighted average of n surrounding elements, where n is the width of the smoothing "window". Medians can be used instead of means. The main advantage of using the median compared to moving average smoothing is that its results are less biased by outliers. Thus, if there are outliers in the data (e.g., due to measurement errors), median smoothing typically produces smoother or at least more "reliable" curves than moving average based on the same window width. By analyzing the eDREAM data, the median is chosen instead of the moving average. The median filter is a nonlinear digital filtering technique often used to remove noise from an image or signal. The median filter is widely used to remove signal noise and keep edge information. A 0.1-second window size is selected for the median filter. Figure 3.8 shows the resultant signals from applying the median filter on the eye gaze time-domain signal.

Sample Selection Criteria

With the benefit of the eye-tracker used during the data collection procedure, each frame measurement is labeled with estimation confidence and gaze-tracking quality level. Thus, after the median filter step to remove the noise and outliers, validate sample selection steps are followed with three criteria[10].

- The FaceLAB's automated gaze quality index for the left and right eyes was categorized as optimal.

- The x-axis position of gaze location was between [-1.5 1.5] m, and the y-axis position of gaze location was between [-1 1] m.
- The neighbor 6 measurements are all valid.

Subject-Standardization

To achieve subject-level standardization, the baseline no-task is taken as the baseline data for standardization statistics. For each participant, the mean and standard deviation is computed from the baseline data and the labeled n-back task data are standardized with the z-score equation shown in Equation 4.1.

$$X_{std} = \frac{X_{raw} - \mu_{base}}{\sigma_{base}} \tag{4.1}$$

Where X_{std} is the standardized labeled dataset, X_{raw} is the original labeled dataset, μ_{base} is the mean of the baseline data and σ_{base} is the standard deviation of the baseline data. After this standardization, the inter-subject differences between participants is decreased.

4.3 Fature Processing

4.3.1 Feature Summarization

In this study, the output of physiology and eye-tracking sensors contains not only the raw measures but also features derived from those measurements with sensors themselves. For example, the output of the physiology sensor DLab contains not only ECG measures but also heart rate, which is derived from ECG measurements. However, only looking at one frame of the input signals or one and not considering the temporal information would be insufficient for analyzing time series inputs. Thus, statistic functional applied over a certain time window to transform measurements into input instances were widely used for machine learning algorithms. To prevent false detection, the observation usually is summarized from a sequence of measurements. Thus, in this experiment, some summa-

rizing methods are implemented such as mean value, standard deviation and root mean square values at the output of sensors. For eye-tracking sensors, 39 measurements are the output for each frame, and with summarization functions, 117 features are obtained overall. For the physiological sensor Dlab, four measurements applied summarization functions, thus there were 12 features for physiological modalities. The summarization functions are listed in Table 4.1. Where x is the individual output of the sensor, and N is the number of samples in each window, i is the index of the sample in the window.

Table 4.1: Feature summarization functions

Mean($\overline{x}$)	$\overline{x} = \frac{\sum_i^N x_i}{N}$
Standard Deviation(std)	$std = \sqrt{\frac{\sum_{i=1}^N (x_i - \overline{x})^2}{N-1}}$
Root Mean Square(rms)	$rms = \sqrt{\frac{\sum_{i=1}^N (x_i)^2}{N}}$

A sliding window is utilized to summarize those measurements. With a large window size, more information is taken into consideration; however, this also brings noise and irrelevant information. This window size is a crucial parameter when implementing the feature summarization. Another key parameter that needs to be taken into consideration as well as with the sliding window is the overlapping ratio. With a higher overlapping ratio, the system more frequently generates estimation results. For example, a 10-second window with a 90% overlapping ratio would generate an estimation result every 1 second. Also, each focus period is 25 seconds long, and with a low overlapping ratio, the less instances the sliding window would get. In machine learning applications, the number of instances affects the performance of the classification. The more instances sent to the classifier for training, the less the classifier would be overfitting. To get a fast estimation, in reality, an overlapping ratio of 90% is utilized. Thus, in this study, a higher overlapping ratio is determined as 90%.

For this study, n-back tasks are utilized, which means that participants need to recall n steps back from the current moment. In other words, the n step window information is meaningful to determine the participant's cognitive workload. Thus, the window size of the n steps would be reasonable since it shows how the features are changed in the n step. As described in Chapter 3, n-back tasks were stimulated with ten sequence letters

and the entire focus period was 25 second long. Thus, each letter was presented in 2.5 seconds. Based on the number of n, the participant needed to recall information from n times 2.5 seconds ago. Thus, the window size is set to be a factor of 2.5 seconds.

Table 4.2: Parameter of window size

times of 2.5 seconds	**Window Size**
1	2.5 seconds
2	5.0 seconds
3	7.5 seconds
4	10 seconds

4.3.2 Feature Number Reduction

With 117 features obtained with the eye-tracking modality, two approaches are conducted to decrease the number of features sent to the classifier. One approach manually selected features which are discriminant based on the literature. Another approach uses a feature reduction technique such as principal component analysis (PCA). Principal component analysis (PCA) is a statistical procedure that uses an orthogonal transformation to convert a set of observations of possibly correlated variables into a set of values of linearly uncorrelated variables called principal components. For example, if there are n observations with p variables, then the number of principal components is $min(n-1, p)$, and the first principal component has the largest possible variance.

Manually Select Features

As described in Chapter 3, gaze location, blinking frequency and pupil diameter are widely used in the literature as indicators for the participant's cognitive workload level. Thus, in this study, those measurements are selected and shown in Table 4.3. For gaze location, there are three values (x,y,z) which correspond to the world coordinate corresponding to a reference point defined during the data collection setup procedure. Pupil diameter measures for both eyes and blinking frequency are derived by the eye-tracking sensor. Those six measures are applied with feature summarization functions listed in Table 4.1, In the end, a total of 18 features are utilized for manual selection approaches.

Table 4.3: Measurements selected for eye-tracking modality

Measurements	**Unit**
Gaze Location(x,y,z)	Meter
Pupil Diameter (L,R)	Meter
Blinking frequency	Hz

For the Physiology modality, only 4 measurements are collected and a total of 12 features. All measures are approved to be good indicators for drivers' cognitive workload level. Thus, all features are selected and sent to the classifier. Table 4.4 lists all the measures collected from Dlab sensors.

Table 4.4: Measurements selected for physiological Modality

Measurements	**Description**
ECG	Electrical activity of the heart(mV)
Heart Rate	number of heart beat per second
Respiration	Shape change of torso while breathing
GSR	Electrical characteristics of the skin(uSiemens)

Principal Component Analysis

Features reduction is widely used to extract the most significant features to discriminate the difference between classes. Principal component analysis (PCA) is a statistical procedure that uses an orthogonal transformation to convert a set of observations of possibly correlated variables into a set of values of linearly uncorrelated variables called principal components. For example, if there are n observations with p variables, then the number of principal components is $min(n-1, p)$, and the first principal component has the largest possible variance. With this technique, the components that are most discriminant are selected based on the user-defined number. However, the disadvantage of this technique is that the features space is transformed. After applying PCA to 117 features of the eye-tracking modality, only 3 features are selected with the consideration of the size of the training instances and also the computation time.

4.4 Machine Learning Algorithms

Since the objective of this study is not to explore which machine learning algorithms work best with the specific features to detect drivers' cognitive workload, only one machine learning algorithm is selected, which is Support Vector Machines with the Radial Basis Kernel function. The project aims to classify drivers' cognitive workload level, thus classification is taken into consideration. Also, the number of training instances and the number of features are also needed to be taken into consideration. In the end, SVM is chosen. SVM with a Gaussian radial basis function kernel is a popular binary classification algorithm. For multiple classes, approaches such as One-Vs-All and One-Vs-One can be used to build n-modes, where n is the number based on which approach is used. For One-Vs-All, n is always equal to the number of classes. For One-Vs-One, n is equal $\frac{c(c-1)}{2}$ where c is the number of classes. The linear SVM algorithm seeks a linear hyper-plane to separate the samples with a maximum margin. Since it can not always separate samples perfectly, a soft margin is implemented which assigns a penalization C for misclassified samples and to minimize the cost. Also, not all datasets can be separated by a linear hyper-plane, thus a non-linear kernel function is needed. The Gaussian radial basis function kernel has Function 4.2

$$K(\vec{x}, \vec{x'}) = \left(-\frac{\|\vec{x} - \vec{x'}\|^2}{2\sigma^2} \right) \tag{4.2}$$

where $\vec{x}, \vec{x'}$ are two samples represented as feature vectors. σ is the hyper-parameter for Gaussian radial basis function, and it is necessary to search over a range of possible hyper-parameter to determine the optimum setup. The grid search is implemented in this study for each possible hyper-parameters. The hyper-parameter possible range in this study is listed below:

- Penalization C: $2^{-3}, 2^{-2}....2^2, 2^3$.
- Kernel hyper-parameter σ: $2^{-3}, 2^{-2}....2^2, 2^3$.

For each hyper-parameter, there are 7 candidates, thus a total of 49 possible combinations of two hyper-parameters are gone through for the grid search.

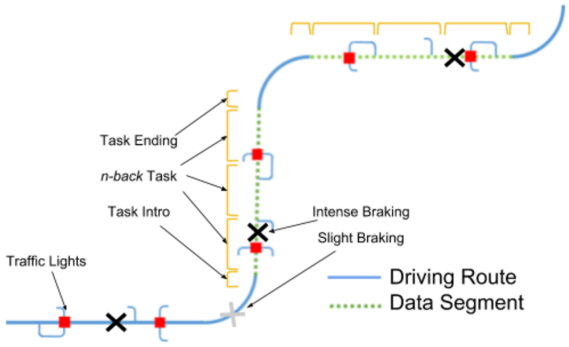


Figure 4.2: Driving Route. Figure taken from [2].

4.5 Performance Evaluation

Performance evaluation is conducted for two purposes: parameters selection and model performance evaluation. In this study, SVM is selected as the classifier which contains two hyper-parameters. To build a practical cognitive workload estimation system, the estimation system should work with unseen data. Thus, simulation experiments with machine learning approaches often require no overlapping between the training set and testing set. This is necessary to avoid testing the performance of the system with data seen before during the training procedure which led to unrealistic high performance. Thus, the dataset is split into training and testing sets which are not overlapped. The way to split the dataset is highly related to the purpose of the machine learning application. Thus, based on the purpose of the project, three methods to divide the training and testing dataset have been used in this study which are adapted from Liu's study [2].

4.5.1 Data Partitioning

As mentioned before, 2 out of 6 focus periods are excluded since those two focus periods involve leading vehicle braking events and they could significantly affect the participants, which would possibly override the effect of cognitive load. As is shown in Figure 4.2, among 6 n-back tasks in one driving trial, two of them involved a braking event. Each n-back task is referred to here as a Focus Period.

With the assumption that after the first Focus Period, other factors would affect

the participant's cognitive workload level such as losing concentration or experiencing frustration, the system will be built with two different directions: using the data collected from same time period, and using data collected from different time periods. With the performance evaluation of those two directions, the assumption made before can be examined. For each direction, same data partition methods utilized in Liu's work[2] are applied. According to Liu's work[2], two kinds of dependencies might exist in datasets studied for predicting driver cognitive load based on their grouping membership. With this consideration, she applied two different grouping methods to the dataset called time-based grouping and subject-based grouping. To compare the performance with her study, the same grouping methods are utilized. The data partitioning methods can be organized as below:

- A pre-trained model based on multi-user collected data and then tested on new users. With this model, no pre-trained model is required for each individual driver. This is corresponding to subject-based grouping methods in Liu's work[2].

- A pre-trained model based on multi-user previous data and tests on the same users' later data. With this model, a pre-trained model is required for each individual driver. However, the test data are the data collected just after the training data and there is no time gap. This corresponds to the time-based grouping methods in Liu's work[2].

Approach (1) is denoted as the Subject-based scheme, where test subjects are not used as part of the training procedure.

Approach (2) is denoted as the Time-based scheme, where subjects in the test set are used in the training procedure.

4.5.2 Cross Validation

During the training procedure, it is common to have a bias estimation system because the number of training instances is small, or the training instances have very similar proprieties. To prevent overfitting problems in machine learning simulation, a common

technique called cross-validation (CV) is applied. With this technique, the training set is divided into training subsets and validation subsets for multiple times based on how the CV is set. With the multiple splitting of the training set, the probability that the optimal performance is obtained with the bias training subset is decreased. There are several options for CV and k-fold and leave-one-out are the most common ones used in machine learning applications. In this study, k-fold is utilized with k equal to 5. So the training set is divided into five subsets, and one of them is chosen as a validation subset with a sequential order and others are used to train the model. This procedure will repeat 5 times until each subset has been utilized as a validation subset. The parameter setting which caused the highest performance will be used as the final parameters setting for training the whole training set.

4.5.3 Performance Metrics

Accuracy (ACC) is the most widely used evaluation criterion in studies on driver cognitive workload detection area. The definition of accuracy in machine learning applications is the fraction of correctly predicted test samples of all test samples. Equation 4.3 shows how accuracy is calculated:

$$ACC(\vec{\hat{y}}, \vec{y}) = \frac{1}{n_{samples}} \sum_{i=1}^{n_{samples}} [\hat{y}_i = y_i] \tag{4.3}$$

where $\vec{\hat{y}}$ is vector of the predicted class labels and $\vec{y}$ is the vector of the ground truth class labels. $n_{smaples}$ is the number of testing instances, and $\hat{y}_i$ and y_i are the i-th instances in the vector of $\vec{\hat{y}}$ and $\vec{y}$. the square barcked "[...]" is the indicator function which return 1 if $\hat{y}_i$ and y_i are equal, and 0 otherwise.

In this study, what is most important is to detect the high cognitive workload since it is dangerous when it is misclassified. Failing to detect this class may lead to traffic accidents. Also, it can be annoying when the system misclassifies low workload to high workload and sends a false alarm. In this situation, other performance metrics are needed, which are Recall and Precision. Precision is the rate of correctly predicted positive among

all the instances that are predicted as positive class. The recall is the ratio of correctly predicted positive among all the instances that are labeled as positive class. Precision and Recall are described in equations 4.4 and 4.5.

$$precision = \frac{TP}{TP + FP} \tag{4.4}$$

$$recall = \frac{TP}{TP + FN} \tag{4.5}$$

where TP is true positive, which is the number of correct predictions as positive class. FP is false positive, which is the number of wrong predictions of positive class. FN is the false negative, which is the number of wrong predictions of negative class. Trade-offs between false alarms and failing to detect events are common in many detection applications. They are often evaluated with Precision and Recall. Another measurement called F-score can be used to combine Precision and Recall by getting the harmonic mean of them for a singular numeric representation of the performance:

$$F_\beta = (1 + \beta^2)\frac{precision \cdot recall}{(\beta^2 \cdot precision) + recall} \tag{4.6}$$

where F_β is the F-Score, β is the recall weight. When the weight of precision and recall is same, the β is set to 1. However, in this application, preventing the danger is more important thus the β need to be increased to give greater importance of recall. In this simulation, both F_1 and F_2 are computed.

4.6 Manually Selected Features

4.6.1 Eye tracking measures

This section presents the evaluation results of the proposed system when applied to classifying between low, medium and high cognitive workload levels with eye-tracking measurements. The evaluation results with three data grouping methods are presented with two different feature number deduction approaches. In this section, only the manually selected 18 features out of 118 features were sent to model for classification.

The experiment with eye-tracking measures was first conducted with the manually selected features described in 3.1. Then, the feature summarization functions in Table 4.1 were applied to those features, thus a total of 12 features was sent to SVM with One-Vs-All approaches. Parameters are listed in section 4.4. The results are shown in Table 4.5.

It is not hard to observe that Time-based Grouping and no Grouping get higher results compared to subject-based Grouping. Subject-based Grouping best performance is not significantly better than guess performance and, therefore, a cross-subject scheme where models are trained and tested with separate participants is not recommended for an automated driver cognitive workload monitoring system. For no Grouping methods, training and testing instances were randomly drawn from the eDREAM dataset. A sliding window was utilized with a 90% overlapping ratio, which leads to two nearby instances having a high similarity. If all testing instances have their nearby instance at a training set it would result in a high performance. Also, this is not applicable in reality since all the data need to be collected first and then the already existing data used for testing. However, Time-based Grouping uses the previous data as training instances and predicts the current driver workload level, which could prevent accidents 1 second ahead of time.

For ternary classes, ACC for all three grouping methods achieved better than guess performance, though they are not able to correctly predict cognitive workload with high accuracy. Also, compared to Liu's [2] results, it almost has a 15% higher accuracy. This confirmed that the proposed features do carry useful power to predict drivers' cognitive workload.

Table 4.5: Performance evaluation for ternary classification with manually selected features with eye-tracking modalities

	Window Size	No Grouping			Subject-based Grouping			Time-based Grouping		
	Seconds	F1	F2	ACC	F1	F2	ACC	F1	F2	ACC
Manually Selected (12)	2.5 seconds	0.561	0.562	0.563	0.384	0.390	0.390	0.561	0.562	0.563
	5.0 seconds	0.618	0.619	0.619	0.402	0.405	0.419	0.618	0.619	0.619
	7.5 seconds	**0.639**	**0.640**	**0.640**	0.404	0.411	0.435	**0.648**	**0.649**	**0.648**
	10.0 seconds	0.630	0.631	0.630	**0.402**	**0.406**	**0.440**	0.630	0.631	0.630
Liu[2]	10.0 seconds		0.569	0.570		0.390	0.395		0.478	0.480

Note: ACC = accuracy,F1=F1-score F2 = F2-score.

With the various window sizes, performance reached the highest level with the window size equal to 7.5 seconds for No Grouping and Time-based Grouping. That is exactly three times at the time of each n-back letter stimuli time. For the 2-back task, the participant needs to remember two letters before and compare them with the current letter. Thus, three letters need to be stored in the working memory. This would be the reason why performance is highest when the window size is equal to 7.5 seconds. However, this is not observed with Subject-based Grouping. With Subject-based Grouping, the highest performance is obtained when the window size is equal to 10 seconds. Which matched with the findings in the literature [23], Liang et al. observed that when the window is size larger, better performance can be obtained.

4.6.2 Physiological measures

Three levels of a cognitive workload detection system were built with physiological manually selected features. The physiological measures include six features shown in Table 4.4, and they were also applied with the feature summarization methods listed in 4.1.A total of 18 features were obtained after. This number is considered reasonable compared to the number of instances in the training set. Thus, those features were directly sent to the SVM classifier with One-Vs-All approaches without PCA. SVM parameters used were listed in section 4.4. Table 4.6 shows the details of the simulation results.

To evaluate and compare the performance of each classification algorithm, the F1-score, F2-score and ACC were obtained. For ternary classes, the performance achieved

Table 4.6: Performance evaluation for ternary classification with manually selected features with physiological modalities

	Window Size	No Grouping			Subject-based Grouping			Time-based Grouping		
	Seconds	F1	F2	ACC	F1	F2	ACC	F1	F2	ACC
Manually Selected (12)	2.5 seconds	0.561	0.562	0.563	0.384	0.390	0.390	0.572	0.574	0.575
	5.0 seconds	0.618	0.619	0.619	0.402	0.405	0.419	0.548	0.550	0.551
	7.5 seconds	**0.639**	**0.640**	**0.640**	0.404	0.411	0.435	0.538	0.541	0.541
	10.0 seconds	0.630	0.631	0.630	**0.402**	**0.406**	**0.440**	0.577	0.581	0.579

Note: ACC = accuracy,F1=F1-score F2 = F2-score.

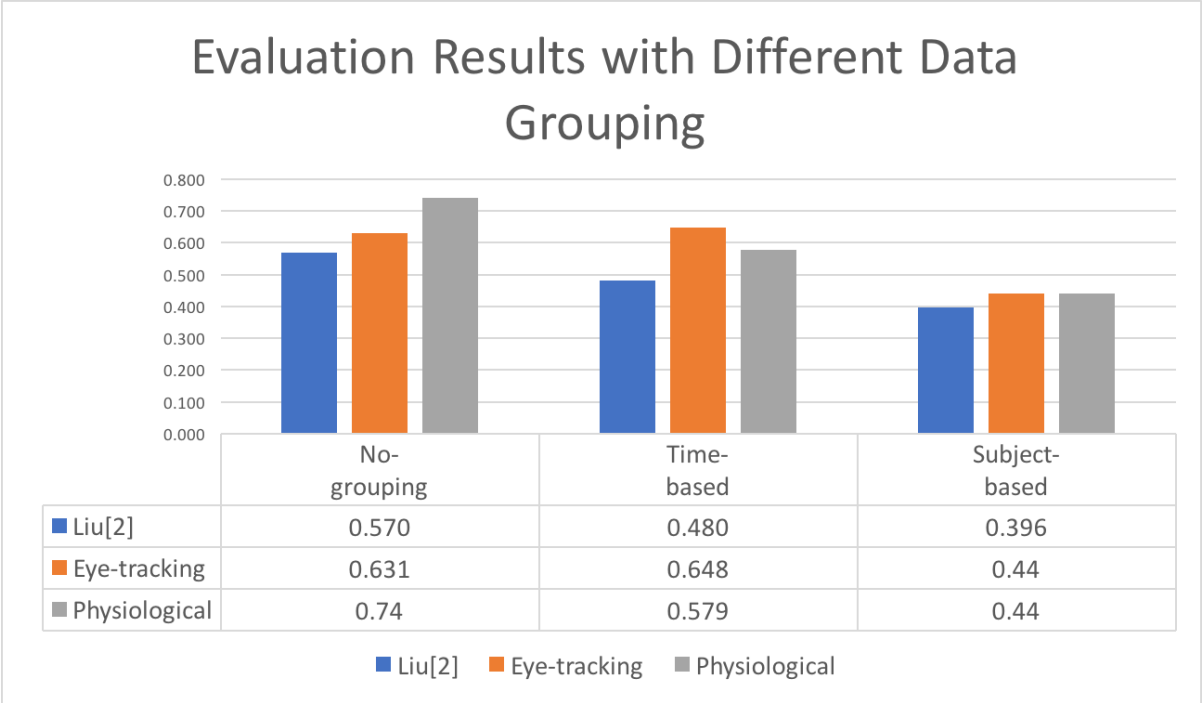


Figure 4.3: Evaluation results with different data partition approaches

better-than-guess performance. However, identifying which one of the three levels of cognitive workload level and the physiological features is inadequate.

4.6.3 Discussion

The eye-tracking measures with manually selected features reached reasonable good performance, which is 65% for the time-based grouping method. This also means that the driver cognitive workload detecting system has the ability to predict and prevent accidents caused by high cognitive workload.

Time-based Grouping and Subject-based Grouping constrains the correlated in- stances to fall together into the same set (training or test) which is more realistic compared to no grouping situations. However, subject-based training only obtained around guessing performance, which suggested that a pre-trained cognitive workload model with a subjective design would be more realistic under the content of driving conditions.

Window size is another factor analyzed in this experiment that may affect the cognitive workload detection accuracy. With the observation in Table 4.5, the highest performance is obtained with a window size equal to 7.5 seconds for both time grouping and no grouping-based models. Different observations were noted for the subject-based grouping. When the window size is maximum, the subject-based grouping reached the highest. It is worth noting that when the window size larger than 7.5 seconds, the system performance either decreased or was saturated. This matched the assumption that participants need to focus on information within 7.5 seconds in order to complete the n-back task. With data covered in more than 7.5 seconds, irrelevant information contains, thus the performance saturated or decreased. In this case, 7.5 seconds would be the best parameter for the n-back task.

Also, based on Figure 4.3, Physiological modality performance better compared to the Eye-tracking modality for no-grouping and subject-based grouping methods. However, this is not valid for time-based grouping. This might be caused by the variability of physiological measures with time being larger compared to eye-tracking-related measures.

4.7 Feature Reduction with PCA

To further improve the performance, the features reduction technique principal component analysis (PCA) was utilized and most discriminant variables were selected. This overcomes the disadvantage that manually selected features may not fully present the ability to predict driver cognitive workload level. When considering the number of instances, 15 features were set with the PCA technique. After the PCA transformation, those 15 features were sent to SVM for classification. Table 4.7 and Figure 4.4 show the results with the Gaussian kernel function. Time-based Grouping and No Grouping over-

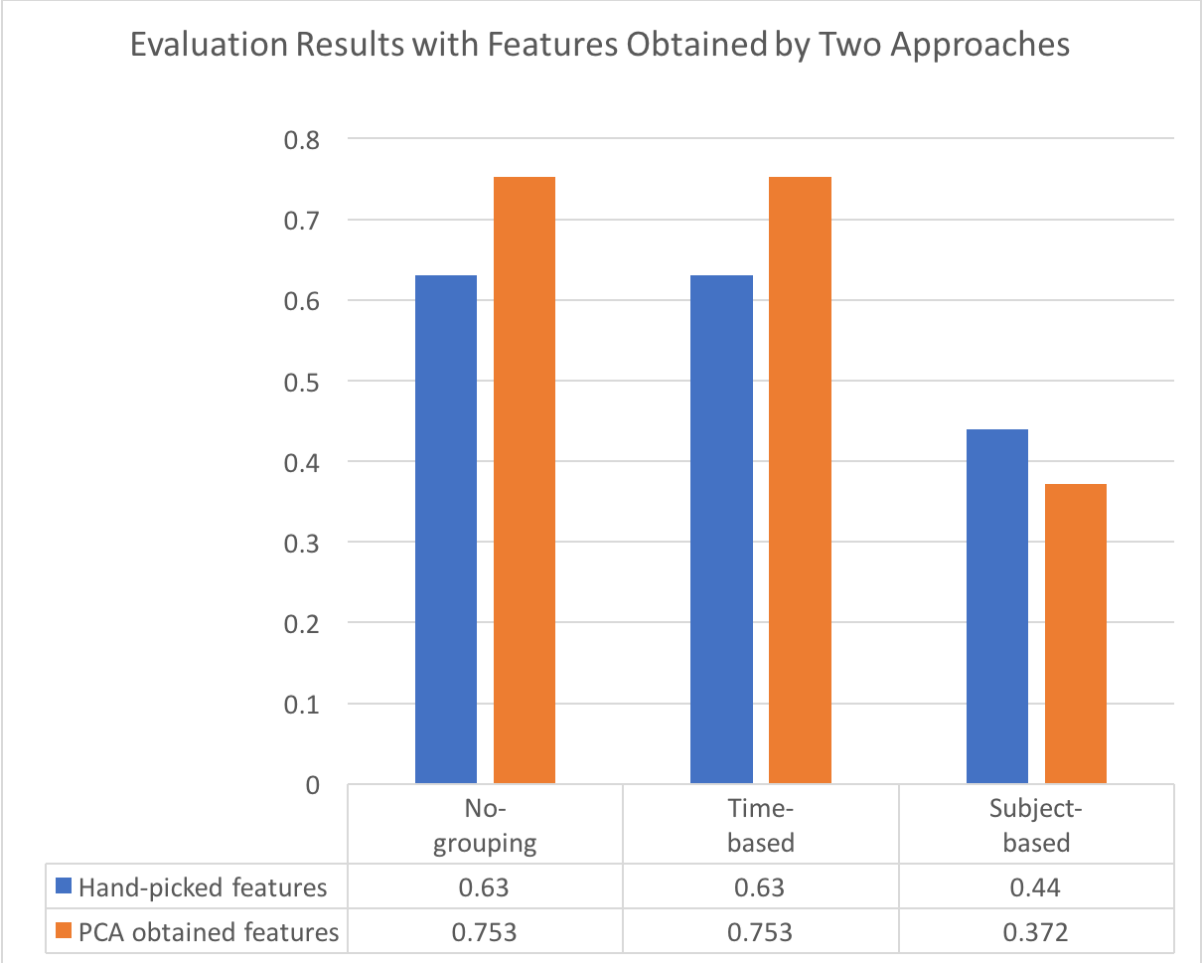


Figure 4.4: Evaluation results with features obtained by two approaches

come Subject-based Grouping. Compared to manually selected features, PCA obtained almost a 10% higher accuracy for both No Grouping and Time-based Grouping methods. This proves that PCA could boost the performance. However, for Subject-based grouping, the performance obtained with PCA is lower compared to manually selected features. This might be caused by over fitting with the PCA approach. The PCA obtained features within those training participants showed excellent predictive power, but not with test participants. This can be interpreted as hand-picked features being more general among the participants.

To compare with Liu's [2] results,three grouping methods are also used and the only difference is the features used. Since Liu only considers a window size of 10 seconds with overlapping 90%, the same setting is implemented. The performance is evaluated with

Table 4.7: Performance evaluation for ternary classification with PCA applied with eye-tracking modalities

	Window Size	No Grouping			Subject-based Grouping			Time-based Grouping		
	Seconds	F1	F2	ACC	F1	F2	ACC	F1	F2	ACC
PCA(15)	2.5 seconds	0.722	0.722	0.723	**0.365**	**0.368**	0.371	0.722	0.722	0.723
	5.0 seconds	0.730	0.731	0.731	0.348	0.355	0.373	0.730	0.731	0.731
	7.5 seconds	0.739	0.741	0.740	0.340	0.344	**0.382**	0.738	0.740	0.739
	10.0 seconds	**0.751**	**0.752**	**0.753**	0.325	0.330	0.372	**0.751**	**0.752**	**0.753**
Liu[2]	10.0 seconds		0.569	0.570		0.390	0.395		0.478	0.480

Note: ACC = accuracy,F1=F1-score F2 = F2-score.

Table 4.8: Performance comparison with Liu's result for ternary classification

	Window Size	Machine Learning Algorithm	Time-based Grouping	
	Seconds	Classifier	F2	ACC
PCA(15)	10.0 seconds	rbSVM	**0.752**	**0.753**
Liu[2]	10.0 seconds	rbSVM	0.478	0.480

Note: ACC = accuracy,F2 = F2-score.

only accuracy and F2-score. The results are listed below in Table 4.8. It is noted that with the same machine learning algorithm and parameter settings, the performance of those two approaches has a significant difference. With the simulation in this study, the performance has almost a 30% increase. The only difference behind the implementation is the features used. Since the features used in this study are obtained with PCA, which can select the most significant variance features to discriminate the three levels of driver cognitive workload.

No implementation was done with physiological measures since the number of features obtained with physiological is already small enough. No further feature reduction technique was considered.

4.8 Time Variability Analyze

It is still an open question how to precisely detect cognitive workload using physiological and behavioral signals that are subject to large variability over time [45]. In the study of [45], the day-to-day reliability of physiological measures was analyzed. Observations noticed that with a model trained with one day's measurements, same day testing showed

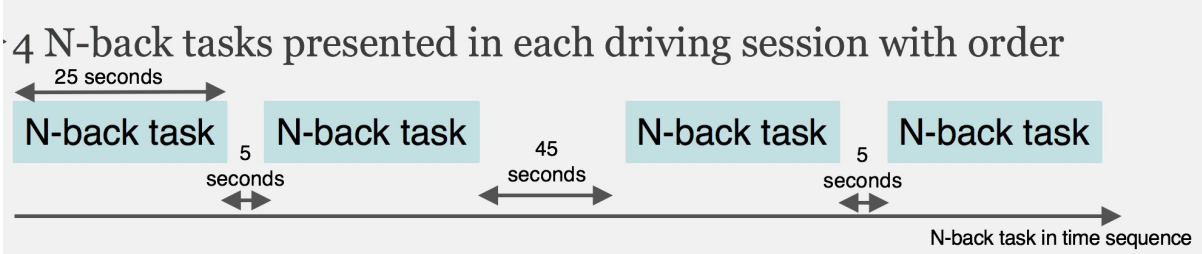


Figure 4.5: Tasks distribution in one driving session

higher performance compared to measures obtained days after. It is shown that the time gap between training and testing datasets affects the accuracy. In this experiment, instead of using the day as the time gap unit, seconds are utilized. With the same experimental design as in [45], the performance obtained with measurements in the same time period will be examined first with three evaluation methods. Then, the cognitive workload will be built with measures collected from different time periods.

Figure 4.5 shows how four n-back tasks are distributed in one driving session in a sequential order. Each n-back task lasts for 25 seconds, and the third one is 45 seconds after the second n-back task. The same time period model is built and evaluated with data collected in individual n-back task periods. Different time period models are built with data collected in one n-back task period and tested with other n-back task periods.

4.8.1 Same time period

This section presents the evaluation results of the proposed system when applied to classifying between low, medium and high cognitive workload levels with measurements from the first n-back task period. It is assumed that the driver may lose concentration or feel tired during the experimental collection procedure, which may affect the driver's cognitive workload level. However, the first n-back task period of each drive would be more presentable for the driver's cognitive workload with the designed driving scenario. This section will first present the simulation results with the first n-back task period previously mentioned in section 4.5.1. The evaluation results with three data grouping methods are presented with two different feature number deduction approaches. One is using PCA to make feature reduction, and in the end, only 15 features out of 118 are

Table 4.9: Performance evaluation for ternary classification with first n-back task period

	Window Size	No Grouping			Subject-based Grouping			Time-based Grouping		
	Seconds	F1	F2	ACC	F1	F2	ACC	F1	F2	ACC
PCA(15)	2.5 seconds	0.988	0.988	0.988	0.325	0.325	0.366	0.816	0.816	0.815
	5.0 seconds	0.984	0.985	0.984	0.313	0.315	0.377	0.827	0.827	0.826
	7.5 seconds	**0.991**	**0.991**	**0.992**	0.288	0.287	0.351	0.876	0.876	0.875
	10.0 seconds	0.987	0.987	0.987	**0.402**	**0.410**	**0.424**	**0.949**	**0.949**	**0.949**
Manually Selected (18)	2.5 seconds	0.966	0.966	0.966	**0.375**	**0.376**	**0.399**	0.720	0.720	0.721
	5.0 seconds	0.973	0.973	0.973	0.352	0.345	0.396	0.758	0.758	0.758
	7.5 seconds	**0.981**	**0.981**	**0.982**	0.356	0.351	0.407	0.786	0.786	0.786
	10.0 seconds	0.949	0.949	0.951	0.327	0.315	0.389	**0.869**	**0.870**	**0.870**
Guess		0.341	0.341	0.341	0.337	0.337	0.337	0.334	0.334	0.334

Note: ACC = accuracy,F1=F1-score F2 = F2-score.

utilized and sent to the classifier. Another one is using manually selected 18 features out of 118 features for classification.

Eye tracking measures

In this experiment, an individual estimation system is built with each set of n-back task period data. To evaluate and compare the performance of each classification algorithm, Table 4.9 reports the test ACC, F1-score and F2-score obtained with three evaluation procedures described in Section 4.5.1. The evaluation not only considers the data splitting methods but also the parameters used for feature summarization listed in 4.2, 4.10,4.11, and 4.12. Two different approaches with the feature number reduction applied.

For ternary classes, ACC would be close to $\frac{1}{3}$ by random guess. All the classifiers not only achieved better-than-guess performance but also reached almost 90% accuracy for Time-based Grouping and No Grouping methods. However, the Subject-based Grouping can only achieve around guess performance. For the No Grouping method, the performance overcomes others which might be caused by the dependencies that exist in the dataset used for training. Since the training instances and testing instances are randomly chosen, there is a high chance that the majority of the testing instances are neighbors of training instances. For Time-based Grouping, the testing instances always belong to the last 7.5 seconds of the 25-second Focus Period. However, since a 90% overlapping ratio is used at the feature summarization step, which caused the similarity between some in-

Table 4.10: Performance evaluation for ternary classification with second n-back task period

	Window Size	No Grouping			Subject-based Grouping			Time-based Grouping		
	Seconds	F1	F2	ACC	F1	F2	ACC	F1	F2	ACC
PCA(15)	2.5 seconds	0.980	0.981	0.980	0.330	0.333	0.341	0.806	0.806	0.805
	5.0 seconds	**0.970**	**0.970**	**0.970**	0.336	0.341	0.354	0.841	0.842	0.842
	7.5 seconds	0.965	0.965	0.965	0.377	0.380	0.379	0.774	0.774	0.775
	10.0 seconds	0.952	0.952	0.953	**0.386**	**0.393**	**0.397**	**0.858**	**0.859**	**0.858**
Manually Selected (18)	2.5 seconds	0.953	0.953	0.953	0.365	0.370	0.390	0.685	0.685	0.685
	5.0 seconds	0.963	0.964	0.964	0.425	0.430	0.461	0.753	0.754	0.753
	7.5 seconds	**0.964**	**0.964**	**0.965**	0.398	0.396	0.426	0.801	0.801	0.801
	10.0 seconds	0.992	0.992	0.992	**0.474**	**0.488**	**0.494**	**0.894**	**0.894**	**0.893**
Guess		0.341	0.341	0.341	0.337	0.337	0.337	0.334	0.334	0.334

Note: ACC = accuracy,F1=F1-score F2 = F2-score.

Table 4.11: Performance evaluation for ternary classification with third n-back task period

	Window Size	No Grouping			Subject-based Grouping			Time-based Grouping		
	Seconds	F1	F2	ACC	F1	F2	ACC	F1	F2	ACC
PCA(15)	2.5 seconds	**0.989**	**0.989**	**0.989**	0.322	0.323	0.325	0.796	0.798	0.796
	5.0 seconds	0.977	0.976	0.976	0.368	0.367	0.372	0.916	0.916	0.916
	7.5 seconds	0.986	0.986	0.986	**0.381**	**0.384**	**0.395**	**0.972**	**0.972**	**0.972**
	10.0 seconds	0.937	0.937	0.937	0.345	0.350	0.369	0.916	0.918	0.916
Manually Selected (18)	2.5 seconds	0.967	0.967	0.967	0.361	0.375	0.395	0.694	0.696	0.695
	5.0 seconds	**0.979**	**0.979**	**0.979**	0.410	0.410	0.414	0.822	0.824	0.822
	7.5 seconds	0.978	0.978	0.978	**0.407**	**0.405**	**0.414**	**0.953**	**0.954**	**0.953**
	10.0 seconds	0.976	0.976	0.976	0.395	0.406	0.428	0.865	0.869	0.865
Guess		0.341	0.341	0.341	0 .337	0.337	0.337	0.334	0.334	0.334

Note: ACC = accuracy,F1=F1-score F2 = F2-score.

stances of testing and training data, even though the testing instances are not the same as the training instances, but the similarity is still existed and is caused by simulation design.

Physiological measures

This section presents the evaluation results of the proposed system when applied to classifying between low, medium and high cognitive workload levels with physiological measurements. This section will first present the simulation results with the first n-back task period mentioned in section 4.5.1. The evaluation results with three data grouping

Table 4.12: Performance evaluation for ternary classification with fourth n-back task period

	Window Size	No Grouping			Subject-based Grouping			Time-based Grouping		
	Seconds	F1	F2	ACC	F1	F2	ACC	F1	F2	ACC
PCA(15)	2.5 seconds	0.986	0.986	0.986	0.321	0.328	0.330	0.823	0.824	0.823
	5.0 seconds	0.991	0.991	0.991	0.358	0.367	0.367	**0.965**	**0.966**	**0.965**
	7.5 seconds	0.992	0.992	0.992	0.394	0.399	0.399	0.962	0.962	0.962
	10.0 seconds	**1.000**	**1.000**	**1.000**	**0.401**	**0.404**	**0.405**	0.927	0.928	0.928
Manually Selected (18)	2.5 seconds	0.966	0.966	0.966	0.331	0.350	0.373	0.694	0.694	0.694
	5.0 seconds	0.983	0.983	0.983	0.352	0.379	0.393	0.844	0.845	0.844
	7.5 seconds	**0.983**	**0.983**	**0.983**	.386	0.424	0.426	**0.950**	**0.950**	**0.950**
	10.0 seconds	0.979	0.979	0.979	**0.403**	**0.451**	**0.439**	0.889	0.890	0.888
Guess		0.341	0.341	0.341	0.337	0.337	0.337	0.334	0.334	0.334

Note: ACC = accuracy,F1=F1-score F2 = F2-score.

methods are presented.

To evaluate and compare the performance of each classification algorithm, Table 4.13 reports the test ACC, F1-score and F2-score obtained with three evaluation procedures described in Section 4.5.1. For physiological measures, no PCA is applied since the number of features is 12 before applying PCA, which is not a large value compared to the size of the training dataset. A similar observation is noticed for physiological measurements in that individual systems can reach 90% accuracy. accuracy.

Summary

Figure 4.6 is the system performance built and evaluated with data collected within same time period. Almost 90% accuracy was reached for both no grouping and time-based grouping methods and around 30% accuracy was reached for subject-based grouping. This makes clear that within the time range, the features are stable within a given time range and include information to correctly discriminate driver cognitive load level with a very high level of precision.

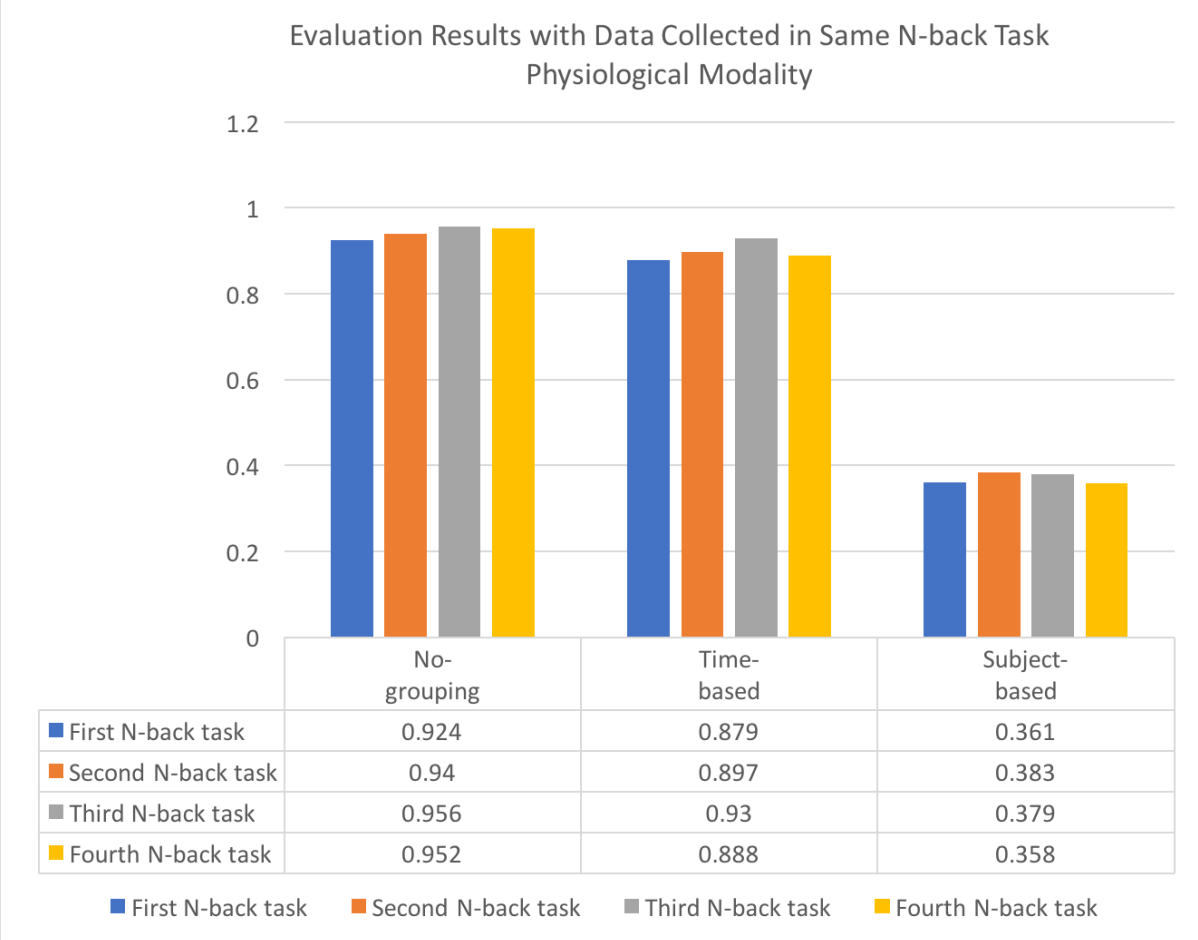
4.8.2 Different time periods

Also mentioned in Section 4.5.1, it is valuable if the system is evaluated with measures which have a time gap between testing instances and training instances. Thus, the

Table 4.13: Performance evaluation for ternary classification with individual n-back task period

	Window Size	No Grouping			Subject-based Grouping			Time-based Grouping		
	Seconds	F1	F2	ACC	F1	F2	ACC	F1	F2	ACC
First Focus Period	2.5 seconds	0.919	0.920	0.919	**0.300**	**0.315**	**0.340**	0.692	0.698	0.692
	5.0 seconds	**0.946**	**0.946**	**0.946**	0.259	0.286	0.338	0.698	0.704	0.697
	7.5 seconds	0.946	0.946	0.946	0.255	0.280	0.343	0.714	0.731	0.715
	10.0 seconds	0.923	0.924	0.924	0.297	0.324	0.361	**0.879**	**0.881**	**0.879**
Second Focus Period	2.5 seconds	0.933	0.934	0.933	0.332	0.343	0.353	0.680	0.684	0.679
	5.0 seconds	0.914	0.914	0.914	**0.404**	**0.409**	**0.407**	0.683	0.684	0.684
	7.5 seconds	0.918	0.919	0.918	0.387	0.401	0.399	0.704	0.707	0.704
	10.0 seconds	**0.940**	**0.941**	**0.940**	0.368	0.377	0.383	**0.897**	**0.897**	**0.897**
Third Focus Period	2.5 seconds	0.948	0.948	0.948	0.297	0.313	0.347	0.690	0.692	0.691
	5.0 seconds	0.948	0.948	0.948	**0.359**	**0.359**	**0.362**	0.724	0.726	0.723
	7.5 seconds	0.947	0.947	0.948	0.338	0.359	0.370	0.754	0.761	0.756
	10.0 seconds	**0.956**	**0.957**	**0.956**	0.333	0.359	0.379	**0.930**	**0.930**	**0.930**
Forth Focus Period	2.5 seconds	0.926	0.926	0.926	0.330	0.330	0.331	0.680	0.682	0.683
	5.0 seconds	0.941	0.941	0.941	**0.391**	**0.394**	**0.393**	0.735	0.738	0.739
	7.5 seconds	0.947	0.948	0.948	0.325	0.346	0.373	0.724	0.727	0.726
	10.0 seconds	**0.952**	**0.954**	**0.952**	0.306	0.333	0.358	**0.889**	**0.892**	**0.888**
Guess		0.341	0.341	0.341	0.337	0.337	0.337	0.334	0.334	0.334

Note: ACC = accuracy,F1=F1-score F2 = F2-score.



	No-grouping	Time-based	Subject-based
First N-back task	0.924	0.879	0.361
Second N-back task	0.94	0.897	0.383
Third N-back task	0.956	0.93	0.379
Fourth N-back task	0.952	0.888	0.358

Figure 4.6: System performance with model build with data in same time period

dependence caused by the simulation design problem can be eliminated. Analysis was conducted with the training model with the first n-back task period and tested with the other n-back task period in each driving trail. A 1.5-minute maximum time gap existed between the first n-back task and fourth n-back task. The experiment was also conducted on both eye-tracking measures and physiological measures.

During this simulation, since the focus is on the effect of the time variability of features and not the feature selection technique, only PCA with 15 features is implemented. Table 4.14 shows the results with test data collected from different time periods. With the 1.5 -minute time gap, the performance obtained with different time periods analysis is lower compared to the evaluation metrics obtained with only using the first n-back task period. The window size has no significant effect on the accuracy, which is around 50% for all window sizes. This could be caused by a loss of concentration or the participant's pressure increasing with time during the data collection procedure. After a 1.5-minute time gap, the cognitive workload level of the participants changed a lot, even though the n-back task level remains the same. It can be assumed that with the time gap increase, more variation would exist in the participant's cognitive workload. It is clear that the prediction performance got lower when the system was built with the first n-back task period and tested with the following n-back task period data. That could be caused by the variance of the cognitive workload level since the participant's driving time kept increasing. Factors may include the participant getting tired and losing concentration or the participant getting frustrated and feeling more pressure. However, this also may involve the quality of test data. To check whether the lower performance is caused by the quality of the test data, another experiment is conducted. However, when the estimation system is built with the first n-back task period and tested with other n-back task periods, the performance gets much lower. This proves that the data quality of each n-back task is enough to distinguish the different levels of the cognitive workload of the participants. With the consideration shown before, this could be because the participants' cognitive workload varied when the focus time increased. This indicates that using a trained model to monitor the driver's cognitive workload level might not be enough.

Time-based Grouping can reach around 90% accuracy, which shows that such an

Table 4.14: Performance evaluation for ternary classification with different n-back task period for eye-tracking modality

	Window Size	2nd Focus Period			3rd Focus Period			4th Focus Period		
	Seconds	F1	F2	ACC	F1	F2	ACC	F1	F2	ACC
PCA(15)	2.5 seconds	0.651	0.652	0.652	**0.518**	**0.518**	**0.520**	0.493	0.493	0.499
	5.0 seconds	**0.655**	**0.656**	**0.655**	0.488	0.487	0.490	0.528	0.530	0.534
	7.5 seconds	0.640	0.641	0.643	0.486	0.485	0.491	0.518	0.521	0.533
	10.0 seconds	0.536	0.540	0.556	0.493	0.492	0.500	**0.536**	**0.540**	**0.556**
Guess		0.341	0.341	0.341	0.337	0.337	0.337	0.334	0 .334	0.334

Note: ACC = accuracy,F1=F1-score F2 = F2-score.

application has the ability to detect the driver's cognitive workload level in advance and can prevent accidents. However, this application needs to use the data just before the accident as in the training samples, which is not realistic.

A similar observation is noted for physiological measurements as individual systems can reach 90% accuracy but this is lower with the later model, which consists of simulations conducted with eye-tracking measurements. The results are listed in Table 4.15.

Summary

Two different models were built to analyze the time variability of eye-tracking and physiological features. Figure 4.7 shows the results of the physiological modality with two different models. When the model was built and evaluated with the same time period data, an almost 90% accuracy was obtained, which means the features are stable within a given time range and include information to correctly discriminate driver cognitive load level with a very high level of precision. However, when models were built with different time periods, the performance decreased considerably even with a time difference of minutes. This proves the time variability of the eye-tracking and physiological features would affect the classifier accuracy.

To summarize the experiments conducted for eye-tracking measures and physiological measures with regard to time gap analysis:

- With individual n-back task periods, driver cognitive workload can reach around 90% classification accuracy for the Time-based Grouping. This proves that the

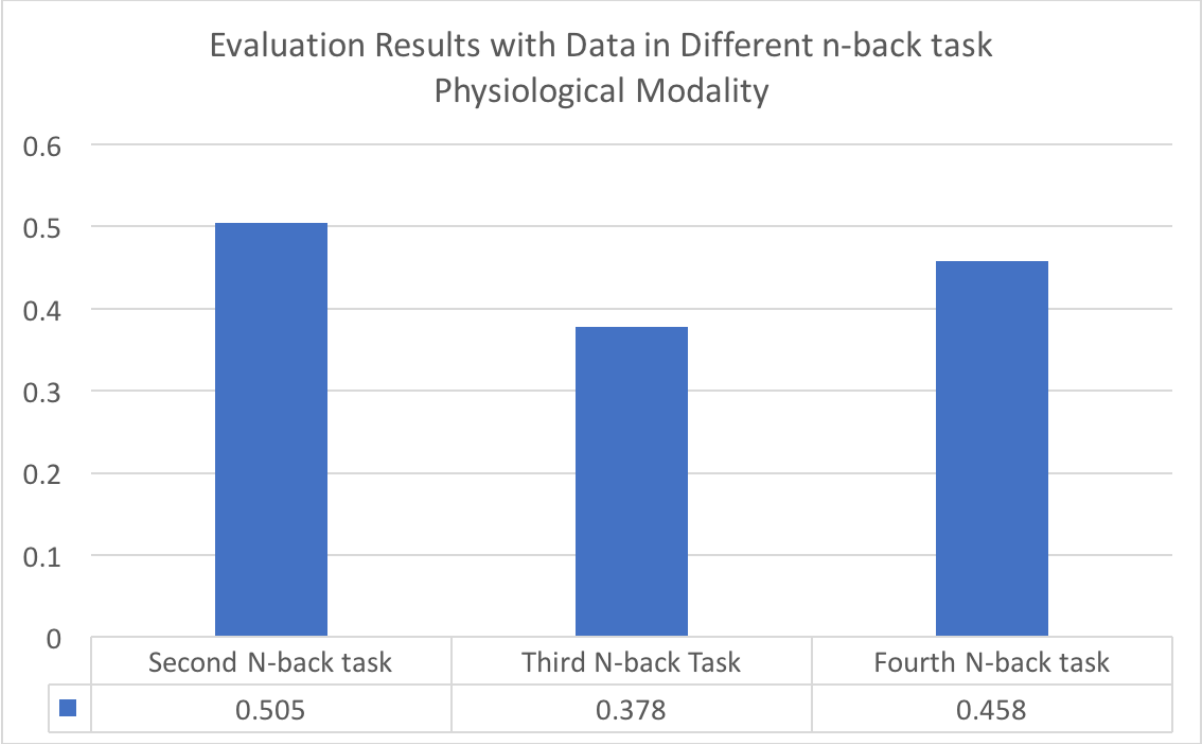


Figure 4.7: System performance with model build with data in different time periods

Table 4.15: Performance evaluation for ternary classification with different n-back task period for physiological modality

	Window Size	2nd Focus Period			3rd Focus Period			4th Focus Period		
	Seconds	F1	F2	ACC	F1	F2	ACC	F1	F2	ACC
Task-based Grouping	2.5 seconds	**0.525**	**0.528**	**0.529**	0.409	0.413	0.416	0.422	0.427	0.430
	5.0 seconds	0.515	0.517	0.517	**0.428**	**0.434**	**0.436**	0.436	0.438	0.440
	7.5 seconds	0.500	0.505	0.503	0.410	0.418	0.425	**0.448**	0.455	0.455
	10.0 seconds	0.498	0.511	0.505	0.352	0.364	0.378	0.446	**0.457**	**0.458**
Guess		0.341	0.341	0.341	0.337	0.337	0.337	0.334	0.334	0.334

Note: ACC = accuracy,F1=F1-score F2 = F2-score.

features are stable within a given time range and include information to correctly discriminate driver cognitive load level with a very high level of precision.

- With different time period methods, the detection accuracy is only around 50% when the system is built with the first n-back task period and tested with the later n-back task period.

- With the implementation listed in above two points, the reason for the lower performance could be the variance of cognitive workload level of the driver with respect to the focus time. The driver may lose concentration or be frustrated, which can vary the cognitive workload level.

4.9 Chapter Summary

In this chapter, the experiment and results associated with built models to estimate driver cognitive workload are presented. First, the experiment methodology is outlined, followed by the explanation of methods to generate training and testing sets with regards to the requirements of various applications. For Feature Processing, Data selection and smoothing are applied before the feature number reduction. Following this, the machine learning algorithm is described, and the performance evaluation metrics are established. Finally, the Subject-based Grouping and Time-based Grouping are evaluated, and simulation results are presented to discuss how those models affect system performance. Also, the performance differences between PCA selected features and manually selected features based on the literature review are presented and discussed. The assumption that the time variability of measures would affect driver cognitive workload level are examined by comparing the performance of models with different time periods and individual models built with the same time period. The performance obtained with PCA selected features are compared with Liu's [2] work and show significant improvement.

Chapter 5

Conclusion

In this thesis, a practical driver cognitive workload detection system was built separately with eye-tracking and physiological modalities. The motivation of this study and background were introduced in Chapter 1, where the reasoning for adopting visual inputs and physiological modalities to detect drivers' cognitive workload was also explained. Then, Chapter 2 provided a review of prior works with physiological measures and eye-tracking measures. Blinking rate, pupil diameter, heart rate and other measures were found to be reliable indicators of drivers' cognitive workload levels. Thus, those features were selected as the input of the monitoring system. Also, feature reduction techniques were shown to be an alternative way to extract meaningful information from a large number of dimension input signals. Chapter 3 provided the overview of the eDREAM dataset consists of physiological measures and eye-tracking measures. The apparatus of sensors used during the data collection was presented, and sensors' output signals were visualized. Pre-experimental processes including artificial information analyzing and time synchronization were also presented. Feature summarization functions have shown the ability to increase the discrimination among different levels of cognitive workload. Chapter 4 provided the experiment pipeline, including the feature processing methodology and simulation results for both eye-tracking and physiological modalities. Systems trained with PCA obtained features indicating high detection accuracy of driver cognitive workload compared to the system with manually selected features. No significant pattern was observed with system performance with regard to window size for both modalities.

Time gaps exist between training instances, and the testing instances have a significant effect on the system performance. Within the same time periods, the performance can reach 90%, but only around 50% accuracy is obtained if testing instances and training instances were not collected during the same time period.

5.1 Summary of Contributions

This thesis explored the feasibility of predicting driver cognitive load based on eye- tracking data and physiological data of eDREAM dataset individually. Signal preprocessing and feature summarization were performed to transfer the raw data into features with more information. For the eye-tracking modality, blinking frequency, pupil diameter and gaze locations were given the most attention, and the feature summarization functions such as mean and RMS values were calculated with a certain window size to extract more information from those raw measures. With the same data splitting techniques conducted in prior work [2], this thesis obtained enhanced performance of around 70% accuracy with ternary classes. The same window size, overlapping ratio and SVM machine learning parameters were utilized in this study. Instead of gaze information, pupil diameter is included in this study, which provided better performance. Withe the comparison, it shows that features extracted in this study carry more predictive information with eDREAM dataset, which could be adapted and applied to other dataset. Also, the feature summarization functions shown they can increase the ability to detect driver workload level, and to our knowledge, this is not studied in other studies. Although similar background knowledge(feature selection)were studied in other studies, the proposed system is design to extract more information with feature processing step. Lastly, this is the first study to explore physiological modalities of eDREAM dataset, and it shown the feasibility of physiological modalities to detect driver cognitive workload. Further analysis can be made based on features utilize in this study since they have shown good predictive power.

Another contribution is that this thesis determined the practical implications of improving modeling performance via dimensional reduction techniques with PCA. For the

eye- tracking modality, a comparison was made in performance between systems trained with manually selected features and with PCA transformed features. It is shown that with the PCA technique, the detection accuracy can increase the accuracy by almost 10% with ternary classes. PCA has the ability to extract the features that carry most of meaning information, and it could be used for detecting driver cognitive workload system. To be noted that, PCA extracted features is less general to all drivers compared manually selected features. Thus, PCA is not suitable when detection system is built and test with different participants. With user specific workload detection system, PCA would performance better compare to manually picking features based on domain knowledge. Also, the effect of using differing sliding window sizes was evaluated. Four different window sizes from 2.5 seconds to 10 seconds with 2.5-second intervals were used in this study. Best performance reached when window size is equal to 7.5 seconds, which is exactly the time required for 3 stimuli letters. 2-back task required participant to recall 2 letters before and compare to current letter. Thus maximum 7.5 seconds' information required to processed with participants working memory during the n-back task. This matched with our assumption that window size depend on which secondary task was used to introduce cognitive workload level. It is worth noted best window size match our assumption in this study with n-back task, further analyzed required to prove this assumption with other secondary task.

How the time variability of physiological and eye-tracking measurements affect the performance of driver cognitive workload detection system were investigated. Within data collected within the same time period, the time-grouping data splitting model can reach 90% accuracy. However, this accuracy decreased to 60% when training and testing data were collected from different time periods. It shown that features are stable within a give time range and include information to correctly discriminate driver cognitive load level with a very high level of precision. In the study of [45], [45], the day-to-day reliability of physiological measures was analyzed. Within the same measure, the performance stays around 90%. However, only binary classes were analyzed in their study, which is less complicated compared to ternary classes. In this study, the minute-to-minute reliability of both eye-tracking and physiological measures has been analyzed. It presented that

using pretrained models to monitor drivers' cognitive workload level might not be enough. Practical adaptive driver cognitive workload level detection needs to be explored.

5.2 Future Works

In terms of the feature number of reduction approaches, we are interested in studying whether using a feature selection technique would impact the system performance for detecting driver cognitive workload. In the current study, PCA was conducted and it transformed the feature space. We would like to explore whether the feature selection technique can also enhance the performance without transforming the feature space. We are also interested in ranking the features that carry most information to discriminate levels of driver cognitive workload using feature selection techniques. In future work, we will try to explore feature selection techniques instead of feature reduction techniques.

As for the enhancement of the detection accuracy, we are interested in expanding it by applying the decision fusion technique on classification results of the eye-tracking model and physiological model. It is expected that higher system performance can be obtained with the decision fusion technique. In this study, the physiological modality and eye-tracking modality were analyzed separately. Also, in this study, secondary task conditions were considered as driver cognitive workload labeling metrics. We are interested in using self-evaluated rating to label participants' cognitive workload level since it might reveal what happened to each participant even though the same secondary task was presented.

Bibliography

[1] C. C. Liu, D. He, B. Donmez, and K. N. Plataniotis, "edream data collection report," Tech. Rep., 2016.

[2] C. C. Liu, "Towards practical driver cognitive load detection based on visual attention information," Ph.D. dissertation, 2017.

[3] S. G. Hart and L. E. Staveland, "Development of nasa-tlx (task load index): Results of empirical and theoretical research," *Advances in psychology*, vol. 52, pp. 139–183, 1988.

[4] P. L. Broadhurst, "Emotionality and the yerkes-dodson law." *Journal of experimental psychology*, vol. 54, no. 5, p. 345, 1957.

[5] B. Lewis-Evans, D. De Waard, and K. A. Brookhuis, "Speed maintenance under cognitive load–implications for theories of driver behaviour," *Accident Analysis & Prevention*, vol. 43, no. 4, pp. 1497–1507, 2011.

[6] C. J. Patten, A. Kircher, J. Östlund, and L. Nilsson, "Using mobile telephones: cognitive workload and attention resource allocation," *Accident analysis & prevention*, vol. 36, no. 3, pp. 341–350, 2004.

[7] J. He, J. S. McCarley, and A. F. Kramer, "Lane keeping under cognitive load: Performance changes and mechanisms," *Human factors*, vol. 56, no. 2, pp. 414–426, 2014.

[8] J. E. Törnros and A. K. Bolling, "Mobile phone use—effects of handheld and hands-free phones on driving performance," *Accident Analysis & Prevention*, vol. 37, no. 5, pp. 902–909, 2005.

[9] B. Mehler, B. Reimer, and J. Dusek, "Mit agelab delayed digit recall task (n-back)," *Cambridge, MA: Massachusetts Institute of Technology*, 2011.

[10] J. Son, B. Mehler, T. Lee, Y. Park, J. Coughlin, and B. Reimer, "Impact of cognitive workload on physiological arousal and performance in younger and older drivers," in *Proceedings of the Sixth International Driving Symposium on Human Factors in Driver Assessment, Training, and Vehicle Design, Lake Tahoe, CA*, 2011, pp. 87–94.

[11] P. G. Jorna, "Spectral analysis of heart rate and psychological state: A review of its validity as a workload index," *Biological psychology*, vol. 34, no. 2, pp. 237–257, 1992.

[12] B. Mehler, B. Reimer, and J. F. Coughlin, "Sensitivity of physiological measures for detecting systematic variations in cognitive demand from a working memory task an on-road study across three age groups," *Human Factors: The Journal of the Human Factors and Ergonomics Society*, vol. 54, no. 3, pp. 396–412, 2012.

[13] B. Reimer, B. Mehler, J. F. Coughlin, K. M. Godfrey, and C. Tan, "An on-road assessment of the impact of cognitive workload on physiological arousal in young adult drivers," in *Proceedings of the 1st international conference on automotive user interfaces and interactive vehicular applications*. ACM, 2009, pp. 115–118.

[14] B. Mehler, B. Reimer, J. Coughlin, and J. Dusek, "Impact of incremental increases in cognitive workload on physiological arousal and performance in young adult drivers," *Transportation Research Record: Journal of the Transportation Research Board*, no. 2138, pp. 6–12, 2009.

[15] K. R. Hammel, D. L. Fisher, and A. K. Pradhan, "Verbal and spatial loading effects on eye movements in driving simulators: A comparison to real world driving," in

Proceedings of the Human Factors and Ergonomics Society Annual Meeting, vol. 46, no. 26. SAGE Publications Sage CA: Los Angeles, CA, 2002, pp. 2174–2178.

[16] T. W. Victor, J. L. Harbluk, and J. A. Engström, "Sensitivity of eye-movement measures to in-vehicle task difficulty," *Transportation Research Part F: Traffic Psychology and Behaviour*, vol. 8, no. 2, pp. 167–190, 2005.

[17] B. Reimer, "Impact of cognitive task complexity on drivers' visual tunneling," *Transportation Research Record: Journal of the Transportation Research Board*, no. 2138, pp. 13–19, 2009.

[18] B. Reimer, B. Mehler, Y. Wang, and J. F. Coughlin, "A field study on the impact of variations in short-term memory demands on drivers' visual attention and driving performance across three age groups," *Human Factors*, vol. 54, no. 3, pp. 454–468, 2012.

[19] J. Son, M. Park, and H. Oh, "Detecting cognitive workload using driving performance and eye movement in a driving simulator," 09 2012.

[20] J. Son, H. Oh, and M. Park, "Identification of driver cognitive workload using support vector machines with driving performance, physiology and eye movement in a driving simulator," *International Journal of Precision Engineering and Manufacturing*, vol. 14, no. 8, pp. 1321–1327, 2013.

[21] M. Miyaji, M. Danno, H. Kawanaka, and K. Oguri, "Driver's cognitive distraction detection using adaboost on pattern recognition basis," in *Vehicular Electronics and Safety, 2008. ICVES 2008. IEEE International Conference on.* IEEE, 2008, pp. 51–56.

[22] M. Miyaji, H. Kawanaka, and K. Oguri, "Effect of pattern recognition features on detection for driver's cognitive distraction," in *Intelligent Transportation Systems (ITSC), 2010 13th International IEEE Conference on.* IEEE, 2010, pp. 605–610.

[23] Y. Liang, M. L. Reyes, and J. D. Lee, "Real-time detection of driver cognitive distraction using support vector machines," *IEEE transactions on intelligent transportation systems*, vol. 8, no. 2, pp. 340–350, 2007.

[24] Y. Liang and J. D. Lee, "Comparing support vector machines (svms) and bayesian networks (bns) in detecting driver cognitive distraction using eye movements," *Passive Eye Monitoring: Algorithms, Applications and Experiments; Springer: Berlin/Heidelberg, Germany*, pp. 285–300, 2008.

[25] ——, "Using a layered algorithm to detect driver cognitive distraction," in *Proceedings of the seventh International Driving Symposium on Human Factors in Driver assessment, Training, and Vehicle Design, New York, NY, USA*, 2013, pp. 17–20.

[26] Y. Yang, H. Sun, T. Liu, G.-B. Huang, and O. Sourina, "Driver workload detection in on-road driving environment using machine learning," in *Proceedings of ELM-2014 Volume 2*. Springer, 2015, pp. 389–398.

[27] Y. Zhang, Y. Owechko, and J. Zhang, "Driver cognitive workload estimation: A data-driven perspective," in *Intelligent Transportation Systems, 2004. Proceedings. The 7th International IEEE Conference on*. IEEE, 2004, pp. 642–647.

[28] Y. Liang, J. Lee, and M. Reyes, "Nonintrusive detection of driver cognitive distraction in real time using bayesian networks," *Transportation Research Record: Journal of the Transportation Research Board*, no. 2018, pp. 1–8, 2007.

[29] M. A. Recarte and L. M. Nunes, "Effects of verbal and spatial-imagery tasks on eye fixations while driving." *Journal of experimental psychology: Applied*, vol. 6, no. 1, p. 31, 2000.

[30] M. Rahimi, R. Briggs, and D. Thom, "A field evaluation of driver eye and head movement strategies toward environmental targets and distractors," *Applied Ergonomics*, vol. 21, no. 4, pp. 267–274, 1990.

[31] K. A. Brookhuis and D. de Waard, "Monitoring drivers' mental workload in driving simulators using physiological measures," *Accident Analysis & Prevention*, vol. 42, no. 3, pp. 898–903, 2010.

[32] W. Hajek, I. Gaponova, K. Fleischer, and J. Krems, "Workload-adaptive cruise control–a new generation of advanced driver assistance systems," *Transportation research part F: traffic psychology and behaviour*, vol. 20, pp. 108–120, 2013.

[33] J. A. Healey and R. W. Picard, "Detecting stress during real-world driving tasks using physiological sensors," *IEEE Transactions on intelligent transportation systems*, vol. 6, no. 2, pp. 156–166, 2005.

[34] J. P. Healy and R. SmartCar, "Detecting driver stress," *Proceedings of ICPR'00, Barcelona, Spain*, 2000.

[35] N. Li and C. Busso, "Predicting perceived visual and cognitive distractions of drivers with multimodal features," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 1, pp. 51–65, 2015.

[36] R. Mahmoud, T. Shanableh, I. P. Bodala, N. V. Thakor, and H. Al-Nashash, "Novel classification system for classifying cognitive workload levels under vague visual stimulation," *IEEE Sensors Journal*, vol. 17, no. 21, pp. 7019–7028, 2017.

[37] T. F. Quatieri, J. R. Williamson, C. J. Smalt, J. Perricone, B. J. Helfer, M. A. Nolan, M. Eddy, and J. Moran, "Using eeg to discriminate cognitive workload and performance based on neural activation and connectivity," MIT Lincoln Laboratory Lexington United States, Tech. Rep., 2016.

[38] "[PDF]Measuring cognitive distraction in the automobile - AAA foundation .."

[39] B.-G. Lee and W.-Y. Chung, "A smartphone-based driver safety monitoring system using data fusion," *Sensors*, vol. 12, no. 12, pp. 17 536–17 552, 2012.

[40] F. Castanedo, "A review of data fusion techniques," *The Scientific World Journal*, vol. 2013, 2013.

[41] Q. Ji, Z. Zhu, and P. Lan, "Real-time nonintrusive monitoring and prediction of driver fatigue," *Vehicular Technology, IEEE Transactions on*, vol. 53, no. 4, pp. 1052–1068, 2004.

[42] A. Jain, A. Singh, H. S. Koppula, S. Soh, and A. Saxena, "Recurrent neural networks for driver activity anticipation via sensory-fusion architecture," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2016, pp. 3118–3125.

[43] A. Koenig, T. Rehg, and R. Rasshofer, "Statistical sensor fusion of ecg data using automotive-grade sensors," *Advances in Radio Science: ARS*, vol. 13, p. 197, 2015.

[44] L. M. Bergasa, J. Nuevo, M. A. Sotelo, R. Barea, and M. E. Lopez, "Real-time system for monitoring driver vigilance," *Intelligent Transportation Systems, IEEE Transactions on*, vol. 7, no. 1, pp. 63–77, 2006.

[45] G. F. Wilson, C. Russell, J. Monnin, J. Estepp, and J. Christensen, "How does day-to-day variability in psychophysiological data affect classifier accuracy?" in *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, vol. 54, no. 3. SAGE Publications Sage CA: Los Angeles, CA, 2010, pp. 264–268.