Institut für Software & Systems Engineering
Universitätsstraße 6a D-86135 Augsburg

A Symbolic Approach for Classification of Cognitive Workload in Drivers

Enrique Abdon Garcia Perez

Masterarbeit im Elitestudiengang Software Engineering

SOFTWARE ENGINEERING

Elite Graduate Program

Institut für Software & Systems Engineering
Universitätsstraße 6a D-86135 Augsburg

A Symbolic Approach for Classification of Cognitive Workload in Drivers

Matrikelnummer:	1169604
Beginn der Arbeit:	22. Januar 2013
Abgabe der Arbeit:	05. Juli 2013
Erstgutachter:	Prof. Dr. Bernhard Bauer
Zweitgutachter:	Prof. Dr. Elisabeth Andre
Betreuer:	Bryan Reimer, Ph.D., MIT AgeLab

SOFTWARE ENGINEERING

Elite Graduate Program

ERKLÄRUNG

Hiermit versichere ich, dass ich diese Masterarbeit selbständig verfasst habe. Ich habe dazu keine anderen als die angegebenen Quellen und Hilfsmittel verwendet.

Augsburg, Juni 2013

Enrique Abdon Garcia Perez

The family is one of nature's masterpieces.

— George Santayana

Dedicated to the loving memory of Humberto Perez Zamora.

1931 – 1990

ABSTRACT

Cognitive workload estimation in drivers has important implication for traffic safety and the optimization of in-vehicle information systems design. This thesis proposes a state detection system based on the use of physiological signals i.e. electrocardiogram and electrodermal activity because they are non-intrusive and non-invasive. Furthermore, the streaming data from the physiological signals is transformed into a symbolic representation to benefit from the time series component of the data and reduce its dimensionality. Then, a machine learning approach uses a similarity measure to discriminate the driver's cognitive workload levels. An alternative symbolic representation that could potentially allow the system to perform real time classification was also evaluated.

The classification system was evaluated using physiology data collected from 99 subjects driving in a naturalistic environment. An instrumented vehicle (AwareCar) was used to collect the data. During the driving period, the subjects performed secondary tasks "n-back" to induce different cognitive workload levels. Experimental results of discriminating the highest level of cognitive workload (2-back) from the lowest level (just driving) yield a 82% accuracy with the most simple classification approach and 77% using an approach able to classify cognitive workload in real time. The analysis conducted in this thesis shows that the structure of the physiology measures is similar across participants. This thesis lays the foundation for using symbolic representations of time series data to assess a driver's cognitive workload level in a real time monitoring system.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

1 INTRODUCTION

In-Vehicle Information Systems (IVIS) are becoming increasingly advanced and complex. Information systems now include multifunctional displays that allow the driver access to various in-vehicle functions, such as navigation, radio control, or even a phone interface. Newer vehicles provide a voice interface, enabling the driver to interact with these information systems without physical movements. Other systems offer semi-autonomous functions intended to improve the driver's safety. Some of these systems, such as Autonomous Cruise Control (ACC) are intended to reduce the workload imposed by the driving task, while others, such as in-vehicle navigation systems, increase visual and cognitive demand.

Coughlin et al. [13], Gopher and Donchin [27], O'donnell and Eggemeier [65], Yerkes and Dodson [91] suggest that the operator's driving performance is impacted by his current workload level. It has been reported that drivers perform better at an intermediate workload level, as opposed to very high or low workload levels Coughlin et al. [13]. The relationship between workload and driver performance is best summarized by the Yerkes-Dodson law (figure 1.1), which shows that performance deteriorates when workload is too high or too low, and is optimal at intermediate level. A driver that grows fatigued due to lack of sleep or due to monotonous conditions of driving may lose control of the vehicle or fail to respond to an emergency situation. At the other end of the curve, the combination of traffic conditions, in-vehicle distractions, internal stressors can overload the driver's ability to deal with the primary driving task.

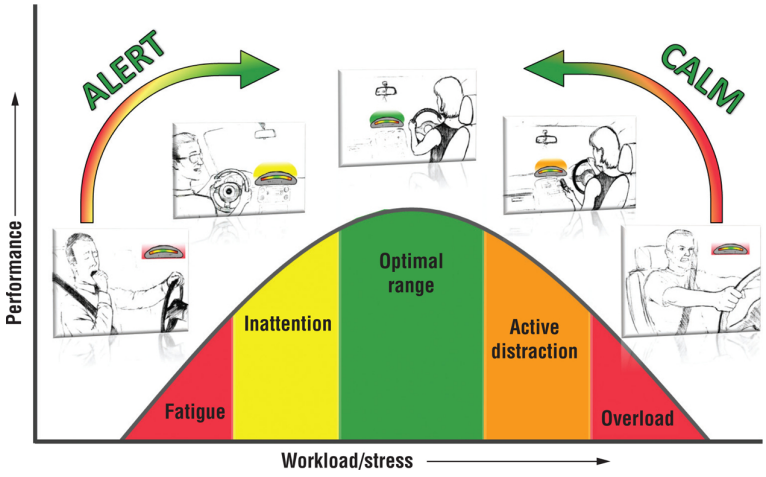

Figure 1.1: The Yerkes-Dodson law adapted to the MIT wellness concept (extracted from Coughlin et al. [14])

Driving while distracted is the cause of many accidents. Distraction can be caused by fatigue, inattention or high levels of stress, as depicted in figure 1.1. Liang and Lee [47] reported that almost 80% of crashes and 65% of near crashes involved some form of driver inattention. The drivers were most commonly distracted by fatigue, driving-related inattention, and/or performing a secondary task. The National Highway Traffic

Safety Administration (NHTSA) reported that in 2009, 5,474 people were killed on U.S. roadways, and an additional 448,000 were injured in motor vehicle crashes. Driver distraction was the cause of 16% of fatal crashes and 20% of injury crashes. Thus, it is important to provide drivers with IVIS that adapt to driver distraction.

Coughlin et al. [14] envision a framework for an in-vehicle integrated safety/wellness system able to detect the current driver state, display this information to the driver, and use in-vehicle systems to alert or calm the driver toward an optimal level of arousal. The proposed system would include vehicle performance metrics such as acceleration events, visual data, and biometrics to assess the driver's cognitive workload level and adapt the IVIS accordingly. Such a system may also be able to increase the driver's awareness and self-monitoring, allowing the driver to make behavioral adjustments. Similarly, a navigation system might be able to adapt the information displayed on the screen when the system detects that the driver is under high levels of stress. Other safety systems in the car could use driver state information to adjust their behavior. For example, an autonomous cruise control system could increase the marginal safety distance if the driver shows symptoms of fatigue or active distraction.

Modern driving is a complex activity involving an ongoing series of manual, visual and cognitive tasks; the driver has to monitor the roadway environment, the dashboard and other information systems to plan and execute a course of action to safely control the vehicle. Given the increasing cognitive demand imposed by newer IVIS, a system that is able to monitor the driver's cognitive workload would be beneficial. This thesis reviews existing approaches to cognitive workload assessment in the vehicle and proposes a data mining approach to analyze the driver's physiological data in order to identify high cognitive workload levels. The physiology data used in this thesis comes from a study carried out at the Massachusetts Institute of Technology AgeLab. The study collected data from 165 participants. Data from 66 subjects were excluded (see Chapter 5 for details), leaving a total of 99 subjects in the final dataset.

THESIS STRUCTURE This thesis presents a symbolic approach to classifying cognitive workload level in drivers. It is organized as follows.

- Chapter 1 introduces the benefits of using cognitive workload information in the driving domain.
- Chapter 2 describes the theory of cognitive workload as well as terminology and measurement techniques. Moreover, this chapter evaluates related work in the field and discusses driving state detection related literature.
- Chapter 3 provides the theoretical basis of the physiological signals (cardiovascular and electrodermal activity) used in this thesis to infer cognitive workload level in drivers.
- Chapter 4 presents the theoretical basis of symbolic approaches to detecting cognitive workload in drivers. The chapter also describes an overview of cross-validation techniques and how they are used for model selection and evaluation.
- Chapter 5 describes the apparatus and procedure followed to obtain the empirical data needed to evaluate the proposed framework for driver state detection.

- Chapter 6 describes the experimental work conducted in this thesis to evaluate the proposed approach in an empirical manner.
- Chapter 7 discusses the results obtained in the experimental work and provides ideas for future work.

2

LITERATURE REVIEW

Improving driving-related safety is one of the automobile industry's key strategic goals. Fatigue, inattention, distraction and cognitive overload are factors that not only compromise the driver's performance, but also endanger the driver's safety. Driving state detection is an emerging research field that combines machine learning principles with real-time assessments of driver workload.A variety of methods have been applied to the detection of driver workload, and satisfactory results have been reported using machine learning approaches. Thus, this chapter introduces cognitive workload and discusses different approaches for its estimation.

Section 2.1 defines the concept of cognitive workload and discusses techniques for measuring it. Section 2.2 discusses theoretical research and applied approaches for driver state detection. This chapter concludes with a general discussion in section 2.3.

2.1 COGNITIVE WORKLOAD

In the automotive domain, in-vehicle information systems (IVIS) and other mobile communication devices impose manual, visual and *cognitive* demands on the driver. *Cognitive workload* refers to the level of mental effort one is employing to accomplish a task. As modern technologies have imposed greater demands on human processing capabilities that affect task performance, cognitive workload detection has become more important.

Themain focus of this thesis is on cognitive workload detection, since it is harder to measure than visual and manual workload [98]. For example, removing one's hands from the steering wheel to enter an address in the on-board navigation system creates manual workload. Typing the address also creates visual workload. In this case, both the manual and visual workloads are measurable by quantifying the hands-off-wheel and eyes-of-road time. In contrast, mental workload is not an activity that can be directly observabed in the driver. In order to measure mental workload, alternate methods have to be employed.

This section introduces the concept of cognitive workload. First, in section 2.1.1, attentional capacity models are introduced as a starting point on the description of cognitive workload. This is followed by a definition of *workload* in section 2.1.2. Section 2.1.4 reviews existent definitions of mental workload and selects the most accurate one, from this author's perspective. Section 2.1.5 discusses several cognitive workload measurement techniques and provides a summary of their strengths and weaknesses, useful when deciding the most appropriate method to use.

2.1.1 *Attentional capacity models*

Attentional capacity models provide a useful basis on which to describe cognitive workload. There are two major perspectives on attentional capacity models. One stream

suggests that attentional capacity is limited [20, 27, 35, 81], and the other considers attentional capacity to be malleable [92].

In the field of cognitive psychology, two major categories of *limited attentional capacity models* use information theory to model the limitations on human performance when performing one or multiple tasks:

- Filter or bottleneck theories
- Resource capacity theories

Filter theories, also known as bottleneck theories, are derived from the information theory field. Thus, psychologists attempt to describe human information processing in terms of the flow of information within the neurons. Gopher and Donchin [27] state that "the human processing system was likened to a communication channel that processes messages and transmits information from a stimulus set to a response set". A crucial concept in this analogy is that the information channel has *limited* capacity to transmit information. Filter theories assume that one information channel can only process one task. Processing more than one task "concurrently" can only occur if the task arrival interval is short. In other words,concurrency implies the ability to rapidly switch between tasks, rather than processing two or more tasks simultaneously [20, 27].

Resource capacity theories, on the other hand, assume that human cognition is similar to a *processing resource* [35]. Gopher and Donchin [27] cites Kahneman [35] to describe the concept of resources as "a label applied to a single undifferentiated pool of energizing forces necessary for task performance". In contrast to bottleneck theories, tasks can be processed simultaneously while the total demanded resources are less than the total available resources. In order to measure available resources, the proponents of resource capacity theories relied on observations of physiological signals. Two subgroups of resource capacity theories exist: Kahneman [35] propose single resource theory and Wickens 84 suggest the existence of multiple resources. Consistent with their names, single resource theories assume a single source of attention, while multiple resource theories propose several sources. In terms of concurrency, both theories converge on the fact that as long as enough resources are available, more than one task can be processed simultaneously.

Young and Stanton [92] propose the theory of malleable capacity models. Contrary to limited capacity models, where it is suggested that reducing task demands improves performance, Young and Stanton [92] states that "Evidence is accumulating that simply reducing demand is not necessarily a key to improving performance". Malleable resource theory proposes that total attentional capacity is flexible and it can change in response to changes in task demands.

This thesis assumes that there are limitations on human attentional capacity, that cognitive workload has an impact on task performance, and that attention has an influence on cognitive workload.

2.1.2 *Defining workload*

Before we can define *mental workload,* the term *workload* must first be defined and understood. Mehler et al. [60] state that many definitions of workload exist, but none are accepted among all researchers, despite growing interest on the subject. Definitions

of workload must consider both the amount of demand imposed on a person and the amount of effort required to complete a task. Mehler et al. [60] cites Linton et al. [53] on these basic tenets that frame the complexity of workload:

- Workload reflects relative, rather than absolute individual states. It depends on both the external demands and the internal capabilities of the individual. This relativity exists qualitatively as well as in dimensions of quantity and time.
- Workload is not the same as the individual's performance in the face of work or tasks, nor is it synonymous with our way of measuring performance.
- Workload involves the depletion of internal resources to accomplish the work. High workload depletes these resources faster than low workload.
- Individuals differ qualitatively and quantitatively in their response to workload. There are several different kinds of task demands and corresponding internal capacities that handle these demands. Persons differ in the amount and type of capabilities they possess, and their strategies for employing them.

As this thesis is supported by research in [56, 57, 58, 59, 70, 71, 72], *workload* is considered to be the level of effort that appears to be associated with the task or conditions.

Workload is commonly used as synonym of some other terms, such as *attention*, *demand*, *distraction* or *stress.* As noted in this chapter, there is no clear and universal definition of workload. However, this thesis intends to provide a clear understanding of the term by differentiating it from related concepts.

2.1.3 *Workload and distraction*

Distraction and *workload* are often treated as the same thing from an operational perspective in research design, i.e. the same secondary task that is used to generate an elevated workload in one study is used to generate distraction on another study [60]. Several definitions of *distraction* can be found in the literature; Regan et al. [69] propose key elements for defining distraction:

- there is a diversion of attention away from driving, or safe driving;
- attention is diverted toward a competing activity, inside or outside the vehicle, which may or may not be driving-related;
- the competing activity may compel or induce the driver to divert attention toward it; and
- there is an implicit, or explicit, assumption that safe driving is adversely effected.

Distraction and workload are not the same concepts. For instance, driving in a high density traffic area can impose high levels of workload on the driver without creating distraction. Conversely, a driver can be distracted with internal thoughts without experiencing high levels of workload. Furthermore, low levels of workload can lead to boredom and eventually to distraction. While distraction is an issue relevant to safety, it is equally important to understand whether the driver is able to respond to changing events [60].

2.1.4 *Defining cognitive workload*

Similar to *workload*, the term *cognitive workload* has no universally accepted definition [93]. Xie and Salvendy [88] assert that "the simple fact is that nobody seems to know what mental workload is." However, many attempts to define cognitive workload have been made.

After an extensive literature review, Cain [9, p. 4-2] concludes that mental workload can be characterized as "a mental construct that reflects the mental strain resulting from performing a task under specific environmental and operational conditions, coupled with the capability of the operator to respond to those demands." This definition of mental workload reflects the relationship between task demand and human capabilities. Similarly Young and Stanton [93, p. 39-1] states that "The mental workload of a task represents the level of attentional resources required to meet both objective and subjective performance criteria, which may be mediated by task demands, external support and past experience." Both definitions are useful in the context of this thesis, because they reflect the fact that when a driver is presented with multiple tasks, such as driving and a cell phone call, the driver can adjust the performance goals for each task according to current priorities; the driver could reduce speed when having a conversation over the phone or the driver could drop the call when entering an unfamiliar driving area to reduce the workload level.

Wilson and Eggemeier [85] define mental workload as "a multidimensional construct, and refers to the ability of the operator to meet the information processing demands imposed by a task or system". This definition acknowledges that the workload level depends on the demand level imposed on the individual and the personal characteristics of that individual.

Xie and Salvendy [88] define mental workload as "the amount of mental work or effort that an individual or a group makes to perform task(s)". Furthermore, [88] introduce a multi-attribute framework that includes instantaneous workload, peak workload, average workload, accumulated workload and overall workload. Each attribute describes one aspect of mental workload. [88] argue that this framework avoids the confusion between average workload, accumulated workload and overall workload, thus making the measurement more accurate.

Figure 2.1 depicts the attributes used to describe cognitive workload in this framework; below is a brief description of each attribute (for a more detailed description see [88]) :

- **Instantaneous workload:** Measures the cognitive workload at any given point in time for the task duration
- **Peak workload:** Is the maximum value of instantaneous workload when performing a task
- **Accumulated workload:** Is the area below the instantaneous mental workload curve in figure 2.1. Accumulated workload measures the total amount of workload that the operator, or in our case the driver, experiences during the task.
- **Average workload:** Is the average of all instantaneous workloads. This measure is equal to the accumulated workload per unit of time.

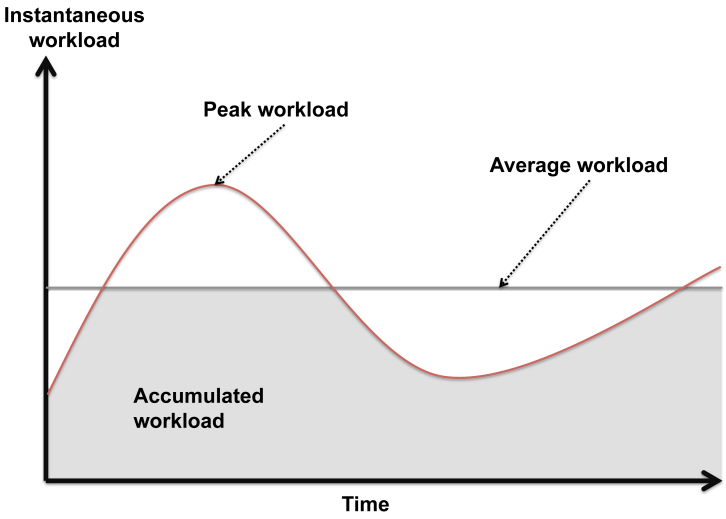

Figure 2.1: Cognitive workload attributes (adapted from Xie and Salvendy [88])

- **Overall cognitive workload:** Is described as the individual's experience of mental workload based on the whole task performance. This measure is not equal to the accumulated workload, nor to the average workload.

Xie and Salvendy [88] emphasized that by combining average and accumulated cognitive workload measures, the workload for long-term and short-term tasks can be measured accurately. This is because mental workload is related to task duration.

Mehler et al. [60] revise Xie and Salvendy [88], drawing on different sources that help to better understand the meaning of cognitive workload:

- Xie and Salvendy [88] suggest that cognitive workload is the amount of mental effort necessary to complete a task over a given period of time. Mehler et al. [60] states "While a useful short statement, this leaves out a number of considerations, some of which are taken up below."

- "Cognitive workload cannot be detected directly, but must be estimated through the measurement of some other variables that are thought to correlate highly with it, such as subjective ratings, performance, or physiological data."Mehler et al. [60]

- "Cognitive workload has dynamic attributes that lead to their distinguishing among instantaneous, peak, accumulated, average, and overall workload."Xie and Salvendy 88.

- Xie and Salvendy [88] note, as do most other careful considerations of mental workload, that individuals have limited processing capacity or resources. Engagement in mental activity does, to some extent, result in the depletion of internal resources. High workload can be expected to deplete these resources faster than low workload. The demand on resources can be (and often will be) unbalanced when performing a particular task or set of tasks. In these situations, some resources may remain under-loaded while other resources are overloaded. Mehler et al. [60] add "These concepts are developed extensively in work on Wickens [83] multiple resource theory. Balancing workload across available resources should

in theory reduce effective workload, extend spare capacity, and have particular relevance in automotive HMIs."

- Xie and Salvendy [88] state that cognitive workload is a multi-dimensional construct and is affected by many factors involving properties of the task, the individual, and their interaction. Mehler et al. [60] agree, stating, "This point has been made a number of times, but is remains centrally important in thinking about mental workload."

Furthermore, Mehler et al. [60] notes that the capacity to manage multiple tasks simultaneously declines as the age of the operator increases. Mehler et al. [60] refers to Merat et al. [61] to state that in-vehicle information systems don't take into account the needs and limitations of older drivers in their effort to increase traffic safety.

2.1.5 *Measuring cognitive workload*

Previous sections introduced the concept of cognitive workload. This section intends to describe current techniques used to assess the cognitive workload level in an operator. As previously stated, mental workload is a multi-dimensional construct, but proposed measurement techniques can only capture some of these dimensions. Thus, when evaluating cognitive workload measurement techniques, is important to understand the explicit or implicit definition assumed by the technique in question.

O'donnell and Eggemeier [65] established some criteria to guide the selection or development of an appropriate workload measurement technique:

- Sensitivity: Capability of a technique to discriminate significant variations in the workload imposed by a task or group of tasks. Figure 2.2 presents a theoretical relationship that evaluates the sensitivity of a procedure to the application.
- Diagnosticity: Capability of a technique to discriminate the amount of workload imposed on different operator capacities or resources (e.g., perceptual, executive, or motor resources),
- Intrusiveness: The tendency for a technique to cause degradations in ongoing primary task performance,
- Implementation requirements: Factors related to the ease of implementing a particular technique. Examples include instrumentation needs and operator training requirements.
- Operator acceptance: Degree of willingness on the part of operators to follow instructions and actually utilize a particular technique.

Cain [9] analyzed the aforementioned criteria and formulated the general requirements of selecting a workload measurement technique as follows:

- The method must be reliably sensitive to changes in task difficulty or resource demand and discriminate between significant variations in workload.
- The method should be diagnostic, indicating the source of workload variation and quantifying contributions by the type of resource demand.

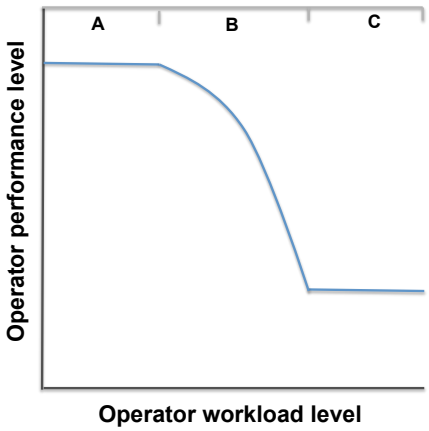

Figure 2.2: Hypothetical relationship between operator workload and performance. There are three distinct regions in this relationship. Under low to moderate levels of operator load (Region A), increases in workload are not accompanied by variations in performance. It is assumed that in this region the operator has sufficient spare processing capacity or resources to compensate for increased levels of load and can therefore maintain adequate performance. In Region B, higher levels of workload exceed the capability of the operator to compensate, and performance decrements occur. In this region a monotonic relationship exists between workload and performance. Under extremely high levels of load (Region C), very low levels of performance are assumed to result from the operator's lack of capacity to deal with the workload being imposed (extracted from O'donnell and Eggemeier [65])

- The method should not be intrusive or interfere with performance of the operator's tasks, becoming a significant source of workload itself.
- The method should be acceptable to the subjects, having face validity without being onerous.
- The method should require minimal equipment that might impair the subject's performance.

From this set of criteria, it can be inferred that the starting point should be to define a clear objective to be satisfied by the measurement technique. This task is frequently complex, since goals are vaguely defined [65], i.e. "What is the mental workload of this system?", "Is workload too high?"

Xie and Salvendy [88] cite Lysaght et al. [54] on a taxonomy of methodologies to measure cognitive workload (depicted in figure 2.3). This taxonomy is also consistent with Wilson and Eggemeier [85]. The methodologies presented in the taxonomy will be explained in more detail in the following sections.

2.1.5.1 *Analytical Methods*

Analytical methods for cognitive workload prediction assess performance and workload according to a pre-defined set of parameters. Thus, they don't require and operator-in-the-loop to asses cognitive workload. Analytical methods include comparison, expert judgments, mathematical models, task-analysis methods and simulation models, as shown in figure 2.3. Even when a potential operator or an operator of a similar system offers expert advice in the early stages of development, this definition of analytical method applies.

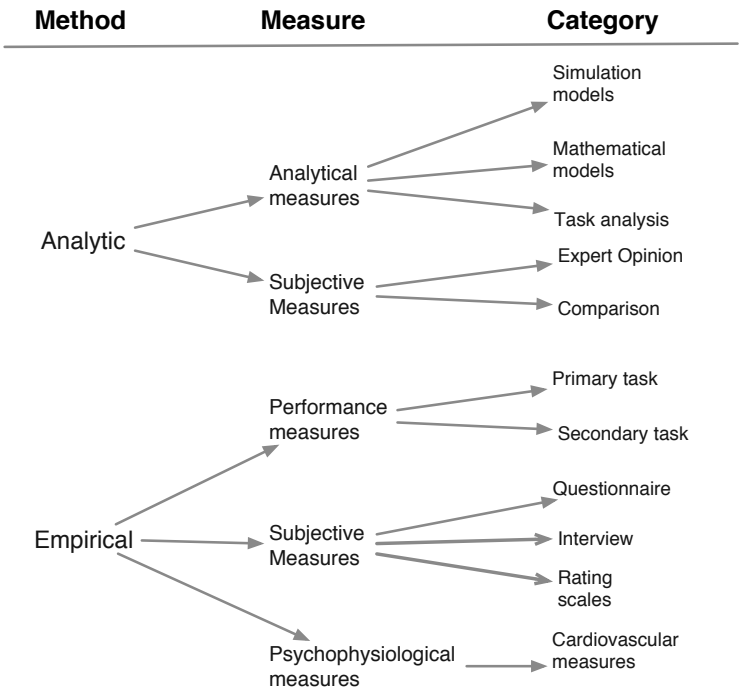

Figure 2.3: Taxonomy of workload measurement methods (adapted from Lysaght et al. [54] and Xie and Salvendy [88]).

An important advantage of analytical procedures is that they can be used in the early stages of development to predict performance and potential performance failures [54]. Two main drawbacks of analytical methods can be identified [54]. First, they cannot account extensively for individual differences. Second, they rely on assumptions which may not apply for all types of operators.

The main goal of this thesis is to provide a cognitive workload estimation for drivers. Analytical methods won´t be discussed further because of their aforementioned disadvantages.

2.1.5.2 *Empirical Methods*

Empirical methods use data from an operator and the environment to estimate workload. Three major categories within this taxonomy are found in the literature [88, 54, 85]: *Performance measures* assess an operator's workload level using some aspect of the operator's ability to perform a task. This category includes primary task and secondary task measurements. *Physiological measures* use changes in the operator's physiology to infer workload level. *Subjective measures* assess an operator's opinion about his own workload and includes questionnaires, interviews and rating scales.

Performance measures

Performance measures derive a metric of the operator's ability to perform a task, e.g. number of errors, time to complete the task, reaction time, etc. These measures are supported by the notion of limited attentional capacity theories (section 2.1.1), which state that an operator's performance changes (usually degrades) as compensation for

excessing the overall capacity/resources available. Two major types of performance measures exist: primary-task measures and secondary-task measures.

According to O'donnell and Eggemeier [65], primary-task performance is a measure of the overall effectiveness of human-machine interaction and reflects the outcome of the operator's efforts. In terms of the sensitivity criterion proposed by [65], an advantage of primary task measures is that they can adequately reflect operator performance under particular experiment conditions. Related to figure 2.2, this means that primary task measures can be used to identify different levels in Region B or to discriminate between Region B (overload) and Region A (non-overload). O'donnell and Eggemeier [65] notes that primary task measures are insensitive if the operator remains in Region A. Similarly, if the operator's spare capacity is exceeded (Region C), changes in workload won't be reflected in primary task performance, since the performance level has reached an asymptotic level. Therefore, primary performance measures are not sensitive enough to characterize a task that generates high workload and excessive workload in different subjects.

Secondary-task performance measures can be used to overcome these limitations. Secondary-task measures require the operator to perform two concurrent activities. O'donnell and Eggemeier [65] explains that the primary task workload estimate is derived from the performance of a *secondary task*. O'donnell and Eggemeier [65] identifies two types of secondary-task measures: *loading task paradigm* and *subsidiary task paradigm*.

In the loading task paradigm, the operator is asked to maintain secondary task performance even if this results in a degradation of primary task performance. The general purpose of this procedure is to shift total workload from Region A to Region B (Figure 2.2). Thus, when performance of the secondary task is held constant, changes in primary task difficulty will induce a measurable variation of performance. These types of measures are mainly used in simulation environments.

In the subsidiary task paradigm, the operator is asked to maintain primary task performance in order to measure the workload level in terms of the secondary task performance. In this case, the secondary task doesn't load the primary task, but is used to measure the amount of additional demand that can be supported by the operator while the primary task is performed at a baseline level [65]. Similar to the loading task paradigm, a shift from Region A to Region B (Figure 2.2) is expected, but is accompanied by decrements in secondary task performance that reflect the spare capacity that remains while the performing the primary task [65]. The experiments used in this thesis use *n-back delayed recall of random digits,* and fall into the subsidiary task paradigm.

Researchers often use vehicle data to assess driving performance. As the limited capacity concept states, when the demand imposed on the driver exceeds the amount of available resources, a change in primary (i.e. driving) or secondary task performance is expected. The driver can deal with the cognitive demand in at least three ways:

- Secondary task dismissal,
- reallocation of resources according to performance goals (e.g. maintain secondary task performance),
- adjustment of primary task performance goals (e.g. reducing driving speed).

Various research studies interpret changes in driving performance as an indicator of cognitive workload [66, 70, 71, 72, 21]. Nevertheless, no trivial relationship exists between driving performance metrics and cognitive workload, and sometimes results can be counterintuitive. For example, Brookhuis et al. [6] conducted a simulator study and reported reduced standard deviation of lane position (increased lane keeping performance) while the driver was executing a secondary task, in this case a phone conversation.

Östlund et al. [66, p. 75] provides a comparison of behavioral effects and performance metrics based on results of the HASTE studies (Table 2.1). The intent of this set of performance metrics is to express the behavioral effects of various IVIS on driving performance.

Table 2.1: A selected set of driving metrics and the behavioral effects they try to quantify, related causes, and general interpretation of the authors (extracted from Östlund et al. [66, p. 75])

Metric	**Behavioral effect**	**Likely secondary task-related cause**	**Interpretation**
Mean speed	Speed reduction	Visual	Safety margin compensation on regulation layer to increase available time in the tracking control loop.
	Large speed increase/reduction	Cognitive	Reduced monitoring performance.
Maximum speed	Large speed increase	Cognitive	Reduced monitoring performance
Mean lateral position	Changed position in the lane during visual load	Visual	Safety margin compensation to increase time-to-contact to the side which is perceived the more risky (e.g. the one with oncoming traffic)
Modified lateral position variation	Increased variation	Visual	Reduced lateral tracking control due to reduced visual input.
	Reduced variation	Cognitive	Enhanced lateral tracking control due to reduced visual input.
Line crossings	Increased frequency of medium-large reversals	Visual	Reduced tracking control due to reduced visual input.
Steering wheel reversal rate	Increased frequency of medium-large reversals	Visual	Reduced lateral tracking control due to reduced visual input.

	Increased frequency of medium-large reversals	Cognitive	Reduced regulating (safety margin setting) performance.
Min time headway	Increased headway	Visual	Safety margin compensation on regulation layer to increase available time in the tracking control loop.
	Reduced headway	Cognitive	Reduced regulating (safety margin setting) performance.
Min time headway	Reduced min headway	Cognitive	Reduced regulating (safety margin setting) performance.
Break reaction time	Increased BRT	Visual	Reduced forward visual attention.
	Increased BRT	Cognitive	Reduced regulating/monitoring ability (generally reduced situation aware- ness).
Brake jerks	Increased frequency	Visual	Reduced longitudinal tracking control.
	Increased frequency	Cognitive	Reduced regulating (safety margin setting) control.

Subjective Measures

Subjective measures, also known as self-report measures, can be used to estimate cognitive workload. As figure 2.3 shows, subjective techniques can be questionnaires, interviews or rating scales. Cain [9] doesn't consider questionnaires and interviews to be reliable measures of cognitive workload, since they are just verbal descriptions of the operator's experience; the author rather recommends rating scales since they can be quantified and validated empirically . Cain [9] cites Jex [34]: "In the absence of any single objective measure of the diffuse metacontroller's activity, the fundamental measure, against which all objective measures must be calibrated, is the individual's subjective workload evaluation in each task".

Cain [9] and O'donnell and Eggemeier [65] examine various self-report techniques, such as the NASA Task Load Index (NASA TLX), Rating Scale Mental Effort (RSME), and Subjective Workload Assessment Technique (SWAT), among others. The validity and sensitivity o subjective measures are based on the assumption that increased capacity in Regions A and B of Figure 2.2 will be related to feelings that the operator can report accurately [65]. Subjective measures have practical advantages such as ease of implementation and non-intrusiveness. However, several disadvantages are also known: validation with respect to the definition of workload is non-existent [27], restricting interpretation of results. These limitations are entangled with other potential influences on subjective estimates, such as factors that might increase or decrease the degree of workload perceived by the operator (e.g. confounding mental and physical workload) and/or methodological constraints (e.g. delay in reporting the workload ratings) [65]. For a detailed examination of subjective measures see [9, 27, 65].

Physiological Measures

Physiological measures provide a continuous and objective measurement of the driver's state without interfering with primary task performance. The field of psychophysiology relates physiological variables to cognitive processes and attempts to interpret the effects that psychological processes have on the body. The success of these measures raises the possibility of a continuous workload monitoring system, able to detect changes in the driver's physiology and correlate those changes to a certain cognitive state.

According to de Waard [18], researchers measure two anatomical structures as physiological indicators, the Central Nervous System (CNS) and the Peripheral Nervous System. The same author asserts that: "The CNS includes the brain, brain stem and spinal cord cells. The Peripheral Nervous System can be divided into the Somatic Nervous System and the Autonomic Nervous System (ANS). The Somatic Nervous System is concerned with the activation of voluntary muscles, the ANS controls internal organs and is autonomous in the sense that ANS innervated muscles are not under voluntary control. The ANS is further subdivided into the Parasympathetic Nervous System (PNS) and the Sympathetic Nervous System (SNS) ... Most organs are dually innervated, i.e., both by the sympathetic and the parasympathetic nervous systems. While traditionally these branches are seen as subject to reciprocal central control - as a continuum from parasympathetic to sympathetic dominance - recently, a two-dimensional autonomic space was proposed with a parasympathetic and a sympathetic axis...".

Measures from the Autonomic Nervous Systems include pupil diameter, electromyography (EMG) for muscle activity, electrocardiography (ECG or EKG) for heart activity, pneumography for respiratory activity and electrodermal activity (EDA) for electrical activity of the skin. Methods of obtaining central nervous system measures include electroencephalography (EEG) for brain activity and electrooculography (EOG) or nonintrusive eye tracking [18]. Many methods record the electrical activity of a specific part of the human body. Various measures can be derived from these and related measurement methods (e.g. heart rate, blood pressure, respiratory rate or skin conductance level).

Backs and Boucsein [1] appraised the literature examining the relationship between psychological states and physiological activity. The authors distinguish between physical, cognitive and emotional demands. The survey reports the following relationships between cognitive demand and physiological measures (relationships that the authors describe with "greater confidence" are in bold).

Table 2.2: Survey results of physiological measures and their respective cognitive demand reaction (extracted from Backs and Boucsein [1]).

Physiological Measure	***Cognitive demand***
EEG alpha activity (8 – 12 Hz)	**decrease**
EEG theta activity (4-7 Hz)	**increase**
P3 amplitude	**increase**
P3 latency	increase
CNV amplitude	increase

Heart rate	increase
0.1 Hz component of heart rate	**decrease**
Respiratory sinus arrhythmia (heart rate variability)	**decrease**
Additional heart rate	increase
Respiration rate	increase
Finger pulse volume amplitude	decrease
Systolic blood pressure	increase
Diastolic blood pressure	increase
Electrodermal response amplitude	increase
Electrodermal response recovery time	increase
Spontaneous electrodermal response frequency	increase
Eye blink rate	**increase**
Saccadic eye movements	increase
Pupillary diameter	increase
Electromyogram	increase
Epinephrine (adrenaline)	**increase**
Cortisol	increase

A body of research suggests that physiological measures are sensitive to changes in cognitive workload ([4, 5, 8, 56, 60, 72, 96, 97]). While O'donnell and Eggemeier [65] describes measures of brain, eye, cardiac and muscle functions as the most commonly used techniques to assess cognitive workload, Cain [9], Young and Stanton [94] add respiration and electrodermal activity to the list.

Mulder [64], Brookhuis and de Waard [4], Healey et al. [30] report cardiovascular measures to be sensitive to mental workload changes. The heart is stimulated by both parts of the Autonomic Nervous System: the Sympathetic Nervous System (SNS), which responds to stimuli from the environment and promotes arousal, and the Parasympathetic Nervous System (PNS), which is responsible for conserve the heart activity. Heart rate is derived from cardiovascular activity; it increases during periods of higher cognitive activity, i.e. heart rate increases with an activation of the SNS, while PNS activation decreases heart rate. Heart rate-based measures such as the overall heart rate, its variability, and the resulting blood pressure have been reported to be sensitive to dynamic changes in cognitive workload [9]. Chapter 3 provides a more detailed description of the heart activity measures considered in this thesis.

Electrodermal activity (EDA) measures the electrical conductance of the (eccrine) sweat glands and their associated non-sudorific tissues [11]. These changes are activated by ANS activity. Eccrine sweat glands are controlled by the SNS. Thus, eccrine sweat glands react to environmental stimuli, producing ionic sweat. This changes the electrical resistance and conductance of the skin surface. Short events (*phasic*) elicit electrodermal responses (EDR) while longer events (*tonic*) are reflected in the electrodermal level (EDL). In general, faster EDR changs are superimposed on slower EDL changes.

Electrodermal activity has been used as a measure of physiological arousal for many years [18]. de Waard [18] asserts that numerous studies have been conducted where

EDA is related to traffic environment. Reimer et al. [71] performed an experiment that induced three different levels of cognitive workload; reported results indicated that there was a statistical increase in SCL. Nevertheless, de Waard [18] notes that EDA should be used cautiously because it measure overall SNS activation; thus movement artifacts are a possible source of noise.

Measures of eye movement such as eye blink rate, horizontal and vertical eye movement, blink duration, blink frequency, fixation duration and pupil diameter can be used to assess driver strategy [18]. Eye data can be collected unobtrusively and much of the necessary technology is in place and affordable. However, solid know-how of data pre and post- processing is required to acquire reliable information. Cain [9] notes that ocular measures can be sensitive to demand, but are also sensitive to other factors like fatigue. For instance, blink rate has been found to increase with higher mental load induced with memory tasks. At the same time, blink rate has been reported to decline with increased workload from visual stimuli.

In general, physiological measures require technical skills and solid knowledge for appropriate application. When using psychophysiological measures to evaluate mental workload, a thorough evaluation of the objective has to be performed. de Waard [18] asserts that physiological measures differ in their sensitivity to arousal or stages of information processing. A consideration that has to be kept in mind is that psychophysiological measures are not necessarily indicative of cognitive workload; activation of the autonomic nervous system influences these measures. An advantage of psychophysiological measures is that they can be measured continuously and without a conscious interaction with the operator.

Summary of empirical measures

Choosing the appropriate technique to assess cognitive workload is not a trivial task. A thorough analysis of the distinct techniques has to be performed. O'donnell and Eggemeier [65] proposes a set of characteristics to consider when analyzing such measures. The following table summarizes the capability of the major classes of measurement techniques to fulfill the proposed criteria.

Table 2.3: Summary table of the capabilities of empirical measurement techniques (Extracted from O'donnell and Eggemeier [65])

Measurement technique	Sensitivity	Diagnosticity	Intrusiveness

Primary task measures	Discriminate overload from non-overload situations. Capable of reflecting levels of capacity expenditure in overload conditions. Used to determine if operator performance will be acceptable with a particular design option, task, or operating condition.	Not considered diagnostic. Represents a global measure of workload that is sensitive to overloads anywhere within the operator's processing system.	Nonintrusive since no additional operator performance or report required.
Secondary task methods	Capable of discriminating levels of capacity expenditure in non-overload situations. Used to assess reserve capacity afforded by a primary task. Can be used to assess the relative potential for overload among design options, tasks, or operating conditions.	Capable of discriminating some differences in resource expenditure (e.g., central processing versus motor). Diagnosticity suggests complementary use with more generally sensitive measures, with the latter initially identifying overloads and secondary tasks being used subsequently to pinpoint the locus of overload.	Primary task intrusion has " represented a problem in many applications, particularly in the laboratory. Data are not extensive in operational environments. Several techniques have been designed to control intrusion. Potential for intrusion could limit use in operational environments.

Physiological techniques	Capable of discriminating levels of capacity expenditure in non-overload situations. Can be used to assess the relative potential for overload among design options, tasks, or operating conditions.	Some techniques (e.g., event-related brain potential) appear diagnostic of some resources, whereas other measures (e.g., pupil diameter) appear more generally sensitive. Choice of technique dependent on purpose of measurement (screening for any overload versus identifying locus of overload).	Intrusion does not appear to represent a major problem, although there are data to indicate that some interference can occur.
Subjective techniques	Capable of discriminating levels of capacity expenditure in non-overload situations. Can be used to assess the relative potential for overload among design options, tasks, or operating conditions.	Not considered diagnostic. Available evidence indicates that rating scales represent a global measure of load. Lack of diagnosticity suggests use as a general screening device to determine if overload exists anywhere within task performance.	Intrusion does not appear to represent a significant problem. Most applications require rating scale completion subsequent to task performance and, therefore, present no intrusion problem.

2.2 DRIVING STATE DETECTION

Coughlin et al. [13] defines driver state as: "...the overall physical and functional characteristics of the operator such as his or her level of distraction, fatigue, attentional capacity and mental workload". The authors identify seven domains that can be used as input for driver state detection, as depicted in Figure 2.4. In the proposed model, data on a driver's emotional state, physiological arousal, visual attention, driving style, and driving behavior can be combined with data on vehicle performance and environment conditions to provide an estimation of the individual driver state. This section discusses several research projects that have used some of the variables mentioned in Figure 2.4 together with different machine learning techniques to infer the driver's state.

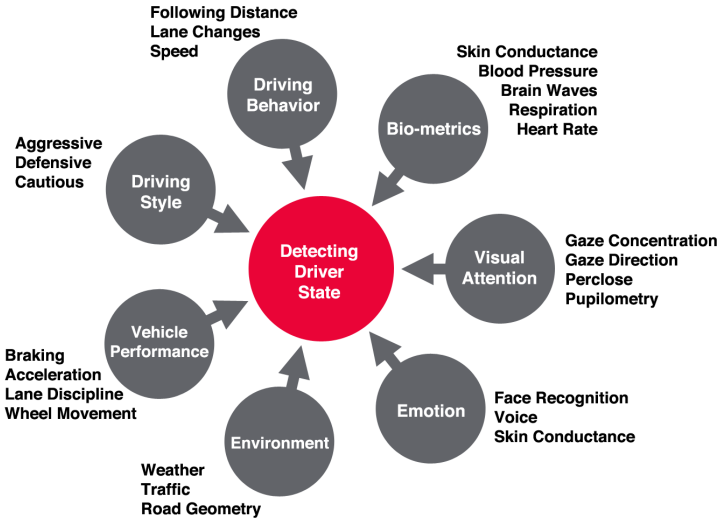

Figure 2.4: Different domains used to extract information about driving state (extracted from Coughlin et al. [14])

Healey and Picard [31] recorded electrocardiogram, electromyogram, skin conductance, and respiration data from nine drivers in a real driving environment (greater Boston area). Three driving conditions were used in the experiment: rest, highway and urban. Each driving condition was designed to cause low (rest), medium (highway), high (city) stress in the driver. Twenty-seven drives of at least 50 minutes in duration were collected; six by drivers who completed the experiment only one time and 21 by three drivers who completed the experiment seven times each. Twenty-four drives were used in the final analysis. Stress level was validated using a questionnaire. A linear discriminant function was used to classify the three different stress levels according to their measured physiological data, achieving 97.4% accuracy. The authors indicate that physiological signals such as skin conductivity and hear rate metrics are correlated with driver stress. Nevertheless, when interpreting the results one has to bear in mind that 21 of the initial 27 drives come from only three subjects.

Rigas and Goletsis [73] used physiological signals (i.e. electrocardiogram, electrodermal activity and respiration) together with driving behavior to detect stress events while driving. Data was collected from thirteen participants in a laboratory setting, but the authors note that the majority of the data come from one subject; arguing that the physiology of a stress event is similar in all humans. The authors used this one subject's data to obtain the learning parameters for a Bayesian Network in an offline setting. Then, other subjects' data were used in an online setting with the trained parameters. The highest Bayesian Network classification accuracy reported was 96% using ten fold cross-validation. From the feature selection process, the authors conclude that skin conductivity and heart rate metrics are most closely correlated to driver stress level.

de Santos Sierra et al. [17] propose a stress detection system based on two physiological signals: galvanic skin response (GSR) and heart rate (HR). The authors emphasize the non-intrusive and noninvasive characteristics of the chosen signals. Eighty female students between the ages of 19 and 32 years old took part in this laboratory study. The physiological signals were obtained under the following conditions: calm state, stimulating task, threatening task and baseline post-stress. The data were then input into a fuzzy system described by Gaussian-based antecedent functions, which attempted to classify the signals into one of the four states. Reported results yielded

over 90% accuracy levels. While this experiment is not focused on the driving domain, it is still an interesting approach for stress detection that could be potentially used in a driving environment. In de Santos Sierra et al. [16], the authors extend their work by comparing the results of the proposed fuzzy system with other machine learning approaches such as k-Nearest Neighbor, Discriminant Analysis, Support Vector Machine, and Gaussian Mixture Models. The results of the study showed that the fuzzy system outperforms the other classification approaches.

Singh et al. [77] use features from galvanic skin response, photoplethysmogram (PPG) and heart rate metrics to estimate driver's stress levels. This experiment consisted of 9 drivers, who drove for 20 minutes in a controlled environment. For the data analysis, the authors propose a novel shape-based feature weight allocation approach supported by the so-called Trigg's Tracking Variable (TTV), which can be used to track alarm trends in real-time. Results report that classification accuracy varied in the range of 71% to 80%.

Fong et al. [26] uses only eye metrics to predict an individual cognitive workload state. This study recorded data from 9 university students who performed the Automated Operation Span (OSPAN) Task, previously used to measure and individual's working memory capacity. The aim if this study is to compare three different machine learning techniques: Artificial Neural Networks (ANN), logistic regression and classification tree. The results showed that ANNs and classification trees performed significantly better than logistic regression for the used dataset. The authors also argue that classification trees are more transparent and easier to interpret than ANNs.

Heger and Putze [32] proposes a daily life continuous cognitive workload monitoring system by means of ECG data. The experiments were twofold. In the first setting, seven subjects participated in a laboratory experiment. Each subject completed three sessions of three workload levels (low, medium, high) on different days. In the second setting, two subjects' daily activities were monitored for three days, each one for a 3.5 to 5 hour period. The subjects labeled their activities by taking pictures using a mobile phone. To analyze the collected data, the authors used a k-Nearest-Neighbor classification algorithm. The results of the laboratory experiment yielded 82.14% accuracy, while the daily life experiment achieve 72.03% accuracy. Both accuracy percentages were estimated using ten-fold cross validation.

Kawakita and Itoh [37] designed a laboratory experimental setting to detect three levels of cognitive workload (low, medium, high) in drivers. They used electrocardiogram and visual information from 18 individuals who drove for a period of exactly 35 minutes. Cognitive workload was imposed using secondary tasks, and the NASA-TLX subjective score was used to estimate the cognitive workload level from the subject's perspective. The authors used multiple linear regression to analyze the data, achieving 93, 85 and 87 R values for high, middle and low mental workload, respectively. It is important to note that each workload level was classified using a different set of features.

Miyaji et al. [62] recorded visual data, head movement data and average heart rate RRI from eight subjects in a driving simulator. Subjects drove in a simulated rural environment trying to maintain a speed of 60 km/h. Cognitive load was imposed through conversation and arithmetic operations. Two machine learning classification algorithms were compared: AdaBoost and SVM. Results indicate that AdaBoost performs better when detecting high levels of cognitive workload compared to SVMs. Using

head movement, gaze, pupil-diameter and heart rate features, AdaBoost achieved 93% and 92.2% accuracy for conversation and arithmetic induced cognitive workload, respectively. SVM on the other hand, achieved 91.7% and 92% accuracy for the same tasks.

Liang et al. [48] conducted a simulator experiment involving ten participants. Temporal eye movement data, spatial eye movement data and driving performance data were collected while the participants interacted with available IVIS. The experiment included six 15-minute drives: two baseline drives and four drives where the participants interacted with an auditory stock ticker, tracked price changes and reported the overall trend of changes at the end of the interaction. Participants were instructed to follow a lead car, respond to its intermittent braking and maintain vehicle position as close to the lane center as possible. Additionally, participants were required to report appearances of bicyclists in the drive scene. Static and dynamic Bayesian Networks were compared for cognitive workload detection. Results indicate that dynamic Bayesian Networks produce more sensitive models than static Bayesian Networks. The average accuracy reported was 80.1%. The authors also note that blink frequency and fixation measures were particularly indicative of distraction.

Zec [95] and **Tan** [80] recorded visual, physiological and driving performance data from 99 participants. Subjects drove in a naturalistic on-road highway environment. Three levels of cognitive workload were induced using "n-back" tasks. **Zec** [95] compares the performance of four feature-base classification approaches: Naïve Bayes, Logistic Regression, k-Nearest-Neighbor and Neural Networks with multilayer perceptron. Ten-fold cross-validation was used to estimate the out-of-sample performance of each classifier. The feature vector for each subject is built using data from the three domains (visual, physiology and driving performance) and a sliding window to extract meaningful statistical values from a series of observations. Moreover, the former author presented a performance analysis of all four classifiers in a pseudo real time experiment. On the other hand, **Tan** [80] employs Support Vector Machines (SVMs) with a Radial Basis Function (RBF) kernel to distinguish between high and low cognitive workload. The author evaluates the classifier using Receiver Operating Characteristic (ROC) graph analysis. Both studies conclude that machine learning approaches are feasible to discriminate elevated cognitive workload levels in a driving environment.

The **Human machine interface And the Safety of Traffic in Europe (HASTE)** study aimed to develop methodologies to assess the effects of in-vehicle information systems on the driving task. The study involved participant data from three different settings: Forty-eight subjects in a fixed-base simulator experiment, Forty-eight subjects in a moving base simulator experiment, and twenty-four subjects in a field experiment [21]. The results of a univariate ANOVA suggested that the effects of visual and cognitive distraction differ significantly: on the one hand visual tasks induce a decreased lateral vehicle control; on the other hand, cognitive tasks caused decreased longitudinal vehicle control. HASTE researchers identified the following measures as sufficient to assess the cognitive workload level of a driver: subjective ratings, mean speed, high frequency steering, minimum headway, percent road center and peripheral detection task reaction time.

The **Adaptive Integrated Driver-vehicle Interface (AIDE)** project investigated how different in-vehicle information systems, driver support systems and nomadic devices (e.g. cell phone) could be integrated into the driving environment using

adaptive technology. The goal was to design a generic adaptive driver-vehicle interface which allows a large number of functions while maximizing the benefits of each individual function and being safe and easy to use. The study collected data from 12 professional truck drivers (e.g. eye movement, head movement and vehicle lane position), driving for 45 minutes in various environments (e.g. motor-way, city and "intermediate-complexity" environment). Participants had to perform visual and cognitive secondary tasks with different levels of difficulty while driving [44]. Support Vector Machine (SVM) was used in an effort to classify different levels of cognitive workload. The authors reported a 65-80% classification accuracy.

The **SAfety VEhicle using adaptive Interface Technology (SAVE-IT)** project developed and evaluated an interface to help minimize the risk of distraction and enhance crash warning system effectiveness [78]. The project consisted of two phases. Phase I involved detection of cognitive distraction, visual distraction, driving task demand, and driving performance. The researchers decided to exclude cognitive distraction, arguing that: "Phase 1 research was unable to develop a set of acceptable countermeasures for cognitive distraction". They stated, "the requirements for supporting the detection of cognitive distraction were beyond the current state-of-the-art for automotive grade hardware..." and that "there were no obvious countermeasures for responding to cognitive distraction in a manner that drivers are likely to accept" [78].

The **Highly Automated VEhicles for Intelligent Transport (HAVEit)** project aimed to provide the driver with an intelligent system that is able to warn the driver in safety critical situations and partially automates some vehicle systems, such as the Adaptive Cruise Control [68]. [33] describes the HAVEit system architecture as consisting of four interrelated components: (1) the driver interface components, (2) a perception layer, (3) a command layer and (4) an execution layer. The driver interface components are responsible for driver monitoring, as well as the interface between the driver and the vehicle. The perception layer consists of environment and vehicle sensors. The command layer uses driver state and real-time environment and vehicle sensor data to select the automation level (i.e. driver only, semi-automated, highly automated or fully automated) and generates a "safe motion control vector" based on the data and automation level. The execution layer receives a control vector from the command layer to control steering, brakes, the engine and the gearbox. The HAVEit system's main focus is to detect drowsiness and/or distraction instead of cognitive workload. Rauch et al. [68] proposes a conceptual method for online assessment of driver's state (sleepy, drowsy, awake) using direct and indirect measures.

2.3 DISCUSSION

The terms *workload* and *cognitive workload* have no universal definition. The aforementioned literature proposed a series of definitions that are useful for particular purposes. Cognitive workload is acknowledged as a multidimensional and complex construct. Despite the lack of a clear definition, most researchers recognize the importance of the study of cognitive workload. An informal definition of cognitive workload is: cognitive workload describes the effect that some external demands have on the operator's ability to fulfill a specific goal.

Various different approaches exist to measure cognitive workload. As workload is a multidimensional construct, selecting the appropriate method to measure cognitive workload depends on the objective to be satisfied. Measurement techniques can be classified into two major categories: analytical and empirical. Empirical methods can be divided into three subcategories: performance, subjective and physiological measures. The choice of the appropriate measure can be evaluated according to the characteristics of the measure technique, including sensitivity, diagnosticity, intrusiveness, implementation requirements and operator acceptance.

Workload state detection is a proliferating topic not only in the driving domain, but also in interface evaluation and aeronautics. This has led to a variety of laboratory experiments as well as field studies. Field studies are designed to resemble real-world environments that would be difficult to replicate with a simulator. One must bear in mind, however, that field studies introduce uncertainty about the real state of the operator, since the environment can rapidly change. The researches also have less control when collecting data on the field, since equipment failures are harder to circumvent, compared to controlled environments such as a simulator.

Various workload detection techniques and algorithms have been proposed (see table 2.4 for a summary of the different algorithms compared within the literature). Extracting and selecting the features with high sensitivity to changes in cognitive workload state is equally important as using the right algorithm. Experiment design, secondary or subsidiary tasks used to increase/decrease cognitive workload, as well as the features used, all have an impact on the performance achieved by the algorithm. Thus, a direct comparison of the different algorithms is not feasible.

While most researchers agree that an individual's condition changes over time due to illness or age, and that individual's physiological signals react differently to changes in workload, some studies are not appropriately designed to be analyzed by machine learning techniques. Therefore, it is important to make clear to the reader whether the model is built using individual characteristics, or if it is built on data from all the subjects in the study's sample.

Factors like the user acceptance, technical feasibility, and cost-efficiency need to be considered when discussing the pragmatism of cognitive workload measurement systems. For instance, electroencephalography is still a highly intrusive and expensive measurement technique, while electrocardiography systems are becoming cheaper and increasingly wearable.

There is no perfect method to assess the driver's mental state, nor is there a perfect algorithm that is able to classify different workload levels with 100% accuracy. This chapter aims to provide a methodical literature review that emphasize key points that need to be taken into consideration when designing a study. Furthermore, an overview of recent experiments and results is presented.

Physiological Signals	Environment	Population	Algorithms	References
ECG, EMG, EDA, Respiration	On-Road	6 subjects	Linear discriminant function	Healey and Picard [31]
ECG, EDA, Respiration	Driving simulator	13 subjects	Bayesian Networks	Rigas and Goletsis [73]
ECG, EDA	Driving simulator	80 female students	Fuzzy system, k-Nearest Neighbor, Discriminant Analysis, Support Vector Machine, Gaussian Mixture Models	de Santos Sierra et al. [17] de Santos Sierra et al. [16]
ECG, EDA, PPG	Driving simulator	9 subjects	Shape-based feature weight allocation	Singh et al. [77]
Eye metrics	Laboratory (not driving domain)	9 university students	Artificial neural networks	Fong et al. [26]
ECG	Laboratory, Daily life	7 subjects	k-Nearest-Neighbor	Heger and Putze [32]
ECG, Eye metrics	Driving simulator	18 subjects	Multiple linear regression	Kawakita and Itoh [37]
ECG, Eye metrics	Driving simulator	8 subjects	AdaBoost, SVM	Miyaji et al. [62]
Eye metrics, vehicle data	Driving simulator	10 subjects	Bayesian Networks	Liang et al. [48]

Table 2.4: Literature review summary on physiological signals, environment, participants and the classification algorithms used.

3

DATA DESCRIPTION

This chapter introduces two physiological signals: heart rate and skin conductance level. Both signals relate to reactions of the autonomic nervous system, specifically the sympathetic nervous system. Thus, a direct relationship between changes in these signals and mental workload level exists. Furthermore, measuring heart rate and skin conductance level can be accomplished with non-invasive methods, which makes them feasible to use in an in-vehicle workload detection system. Research that supports this statement is provided for each signal. Moreover, a detailed characterization and theoretical background of each physiological signal is presented.

3.1 ELECTRODERMAL ACTIVITY

Electrodermal Activity (EDA) or Skin conductance (SC) measures the electrical conductance of skin. It is measured by placing two electrodes on the surface of the skin and passing a small amount of current through them [15]. Skin conductance is also known as *galvanic skin response (GSR)*. This thesis refers to skin conductance as the exosomatic recording of electrodermal activity when *direct current (DC)* is applied directly to the skin and the voltage is kept constant 3. Boucsein [3] describes two standard units to measure skin conductance:

- Skin conductance level (SCL): *Tonic* skin resistance is the resistance level of relatively long term duration and independent of the presence of a stimulus [3]. Figure 3.1 depicts the SCL signal of one subject.

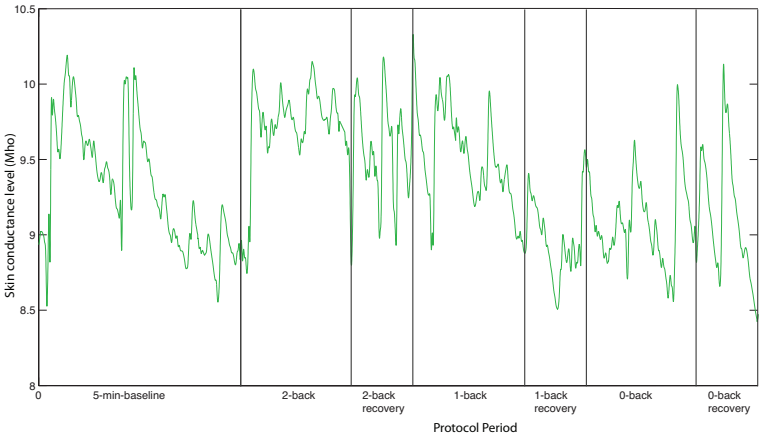

Figure 3.1: One subject's SCL recorded during each protocol period

- Skin conductance response (SCR): *Phasic* measurement in skin conductance is associated with short-term events that usually occur in the presence of an external stimulus like smell, sound or when a person engages in a thinking state. These phasic responses are usually superimposed on the tonic baseline level [2, 3]. Figure 3.2 shows the SCR measure for the same subject and same tasks as figure 3.1.

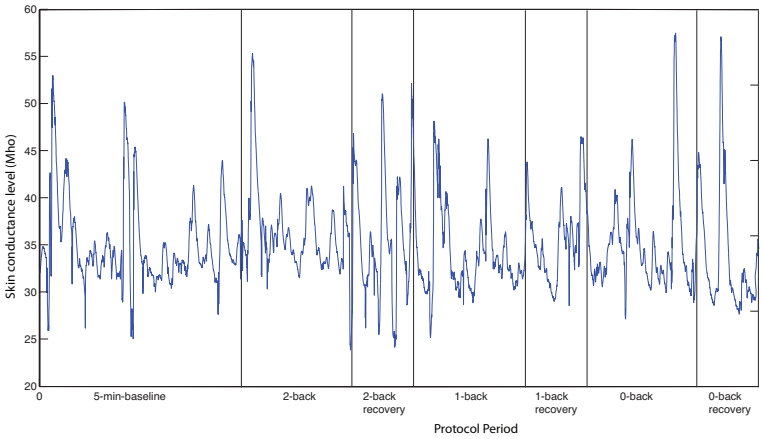

Figure 3.2: One subject's SCR recorded during each protocol period

In the study conducted using the AwareCar, skin conductance level was measured utilizing a constant current configuration and non-polarizing, low impedance gold plated electrodes that allowed electrodermal recording without the use of conductive gel. Sensors were placed on the underside of the outer phalange (#2 in figure 3.3) of the middle fingers of the non-dominant hand [71]. Figure 3.3 depicts other common sensor placements for measuring skin conductance.

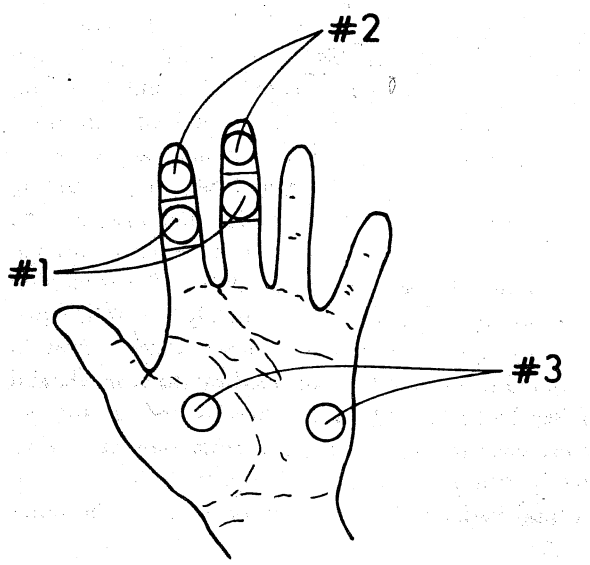

Figure 3.3: Three electrode placement for recording electrodermal activity (extracted from Dawson et al. [15]). Placement #1 Involves volar surfaces on medial phalanges, placement #2 involves volar surfaces of distal phalanges, and placement #3 involves thenar and hypothenar eminences of palms.

Electrodermal activity has been used by researchers in numerous fields to study a variety of phenomena, such as skin disease detection, detecting damage in the peripheral and central nervous systems, lie detection, etc [2]. Boucsein 2 also reports SC being used in the field of engineering psychology for marketing and product evaluation, traffic automation and human-computer interaction. SC recording is used because eccrine sweat glands are under the control of the sympathetic nervous system[15]. The

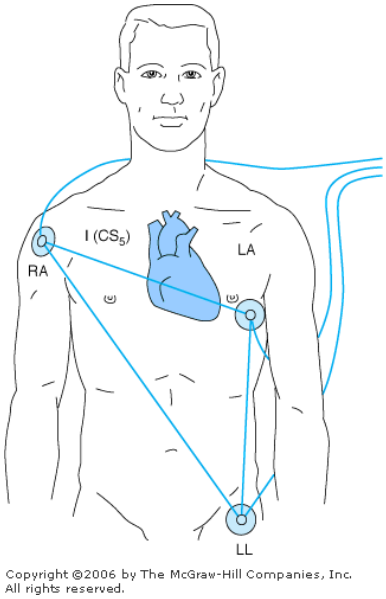

Figure 3.4: Lead II configuration used for ECG recording (extracted from Morgan et al. [63])

sympathetic nervous system is part of the autonomous neural system (ANS), which reacts involuntary to external stimuli [3].

This thesis uses skin conductance to measure autonomic nervous system activity. A body of research exists suggesting that the variations in skin conductance and mental workload are related [2, 7, 30, 31]. Similar research shows that skin conductance level increases with higher cognitive workload [22, 28, 59]. Although skin conductance level is reactive to changes in cognitive demand, Mehler et al. [59] shows the advantages of using *Heart Rate* as an additional measure to estimate the mental workload.

3.2 ELECTROCARDIOGRAM (ECG)

The Electrocardiogram (also known as ECG or EKG) signal measures the electrical activity of the heart over a period of time. It measures heart activity by detecting voltages on the surface of the skin. Heart rate is defined as the number of heartbeats in one unit of time, usually beats per minute (bpm) In this study, a modified lead II configuration was used for ECG recording, as shown in figure 3.4.

A great deal of research shows that heart rate can be indicative of the presence of emotions [46]. Heart rate has also been used in applied automotive environments Mehler et al. [56, 57], Reimer [70]. According to Brookhuis and de Waard [5], heart rate increases when driving demand increases, as when entering a traffic circle. In a similar way, heart rate decreases when driving demand decreases. According to Mulder [64], the main reason for using heart rate as a measure of mental workload is because it is easy to use and the results are equally good and sometimes better than other performance measures, such as Electroencephalography (EEG).

3.2.1 *QRS detection*

A typical EKG waveform of a heartbeat consists of six subwaves labeled P, Q, R, S, T and U (Figure 3.5). Each wave represents a specific stage in the underlying physiological

process of a cardiac cycle. Heart rate is derived from the R-R interval, i.e. the time interval between two R waves.

The P-wave indicates activation of the atrium, the QRS complex indicates the onset of ventricular activation, the T-wave indicates ventricular recovery from activation, and the U-wave represents repolarization of the papillary muscles. The QRS complex is usually the most prominent pattern in the EKG waveform of a heartbeat and therefore makes an ideal entry point for EKG analysis. Due to its characteristic shape, it serves as the basis for automatic heartbeat detection[64]. The temporal location of the peak in the R wave is usually considered to be a reference point for the temporal location of the heartbeat [64].

When analyzing an ECG signal from healthy subjects, a simple threshold algorithm can be used for R peak detection. However, ECG recordings contain noise (e.g. interference with muscle activity) and anomalies (e.g. as a result of heart conditions, electronic interference, equipment failure). Automatic QRS detection has been an ongoing field of research for several years. Köhler et al. [42] analyze a broad range of algorithms for automated QRS detection. The authors report that using benchmarking databases such as MIT-BIH arrhythmia to test the algorithms, the best approach achieves up to 99.5 % detection rate of QRS complexes. Systems used to automatically diagnose patients using ECG data need perfect detection of QRS complexes. Other applications, such as mental workload estimation, can take into account imperfections of the underlying QRS detection system and adjust the estimate accordingly.

The heart's physiology places an upper bound on the length of two consecutive R peaks. Thus, if the QRS algorithm fails to detect one peak, the system can artificially insert a heartbeat [42]. This way, the maximum error in calculating the instantaneous heart rate is constrained. Nevertheless, the robustness of each workload detection system to missing R peaks has to be analyzed.

Figure 3.5 illustrates some clinical features of the ECG waveform and Table 3.1 illustrates normal values for some of these standard clinical features in healthy adult males, together with their upper and lower limits of normality [12].

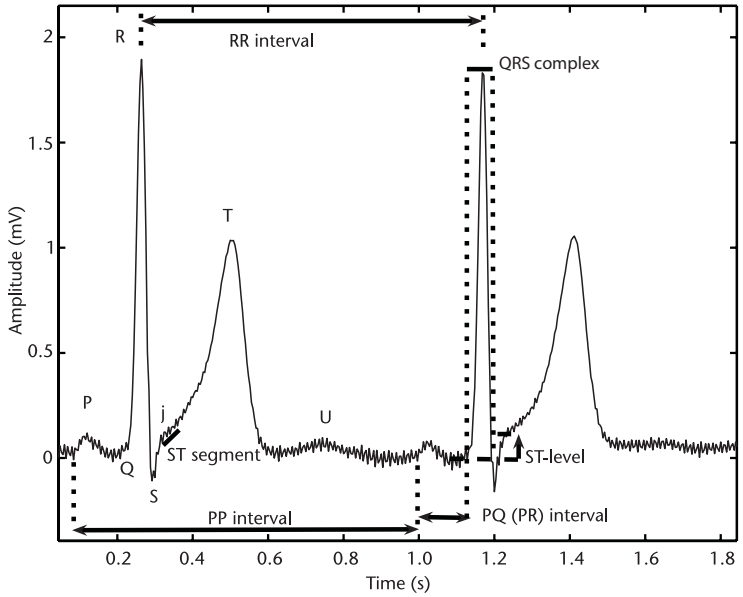

Figure 3.5: Standard fiducial points in the ECG (P, Q, R, S, T, and U) together with some clinical features (adapted from Clifford et al. [12])

Note that the figures in table 3.1 are for constant heart rate. The author also notes that heart rate is calculated as the number of P-QRS-T complexes per minute, but can be calculated over shorter segments of 15 or 30 seconds.

Feature	Normal Value	Normal Limit
PQ/PR interval	160 ms	±40 ms
QRS width	100 ms	±20 ms
QT interval	400 ms	±40 ms
QRS height	1.5 mV	±0.5 mV

Table 3.1: Some Lead II ECG Features and Their Normal Values at a Heart Rate of 60 bpm for a Healthy Male Adult (adapted from Clifford et al. [12])

3.2.2 *Heart rate*

To calculate the heart rate, we need to determine the interbeat interval (IBI), which can be derived from the time difference between two successive R-peaks in the ECG recordings. We can also calculate the heart rate per minute using a single beat interval. In this case, the heart rate is called instantaneous heart rate, and it is calculated as 60 divided by the beat-to-beat interval (RR). If we consider a continuous recording of RR intervals, each one of them will be of a different length and any changes in the ECG morphology are seen on a beat-to-beat basis. Figure (5.5) shows a typical profile of the heart rate data recorded during a highway driving experiment for a test subject.

4

DRIVER'S STATE DETECTION USING A SYMBOLIC APPROACH

Chapter 3 described the two physiological signals (i.e. ECG, EDA) considered in this thesis as measures to discriminate cognitive workload states. Both signals are continuous (*sequential*) measures of a subject's physiology. Using *sequential* or *time series* data to draw conclusion about trends in the population using statistical models, one has to consider three major challenges: dimensionality of time series data can be very high, feature extraction and selection is not trivial and the lack of explicit features makes the results difficult to interpret. Section 2.2 discussed different physiological signals and algorithms used in the recent years to identify the cognitive state of a driver. Feature based classification algorithms (as described later in this chapter) have been used to deal with the sequential data. This thesis uses a distance based classification approach, which discretizes the input signal and applies a distance measure to relate similar signals. To the best of our knowledge, no previous attempt to use a symbolic classification approach for cognitive workload has been made.

4.1 BACKGROUND

Time series data correspond to a sequence of numeric values measured repeatedly over time. The values are measured at equal time intervals (e.g. every second, hour or day) Han et al. [29, p. 587]. Classifying time series data, implies to look for sequences that slightly differ from the given time series. Several approaches to classify time series data have been propose. This thesis lays out the problem of time series classification, and a approach that transform the problem of similarity search to matching sequences in symbolic data. An approach that potentially could accomplish this task in constant time is also presented. Finally, a technique to select the best performing machine learning model is described.

4.1.1 *Time series classification*

Time series classification is applied to a broad range of real-world problems [89]. Deshpande and Karypis [19] use time-series data in genomic research to learn functions of a new protein by classifying a protein sequences into existing categories.Wei and Keogh [82] apply time series classification to: distinguish healthy patients from the ones having a heart disease; differentiate the word "the" from other words using word images; spot a person aiming with a gun in video data.

Physiological signals metrics including ECG, EDA and EEG are *time series* measurements. This thesis acknowledges the importance of using algorithms that capture the temporal relationships among the data. Thus, this section addresses the challenges faced when analyzing sequential data, as well as the widely used classification techniques. The rest of this section defines key terminology that is used throughout next chapters.

Definition 1 A simple *time series* $T = \langle (t_1, 0.1)(t_2, 0.2) \cdots (t_n, m) \rangle$ is an ordered set of m numeric-valued variables.

A sequence is an ordered list of events. An event can be represented as a numerical value, symbolic value or a complex data type.This thesis analyzes physiological measurements, thus the time series is defined as an ordered sequence of numeric values. The dataset used in this work, contains ECG and skin conductance data measurements.

Definition 2 A *d-dimensional time series*

$$T^d = \langle (T_1, \langle 0.1, 0.2, 0.4 \rangle), (T_2, \langle 0.8, 0.9, 0.4 \rangle), \ldots, (T_d, \langle 1.0, 1.1, 0.9 \rangle) \rangle$$

also known as *multivariate time series* is a set of time series associated with one event and having a time correlation [55]. Consider d time series variables $\{T_{1t}\}, \ldots, \{T_{dt}\}$. A multivariate time series is a time series vector $\{T_t\}$ where the i^{th} row of $\{T_t\}$ is $\{y_{1t}\}$. That is, for any time t, $\{T_t\} = (y_{1t}, \ldots, y_{dt})$.

Multivariate time series analysis is used, when one wants to model and explain the interactions among a group of time series variables. In some application domains, time series can be measured with a variety of attributes. Each attribute (*d in definition 2*) is measured on the same discrete interval. The measurements are not required to be independent from each other [55]. This thesis uses heart rate and skin conductance level to relate cognitive workload level. Thus, the time series have $d = 2$ and they are independent from each other. There are some applications where $d = 50$ but the dimensionality of the data is much less [55].

Definition 3 A *labeled d-dimensional time series* is a tuple $E = \left\{T^d, l\right\}$ where T^d is a d-dimensional time series and $l \in L$ where L is a discrete set of labels (not required to be binary).

For simplicity, this thesis uses a binary set of labels L, but in general this set can contain several different values. In this thesis, the periods where the participant is performing a 2-back secondary task are labeled as *high* cognitive workload, whereas baseline periods (subject is driving and no secondary demand is imposed) are labeled as *low* cognitive workload. It is important to note, *low* demand periods might have some higher demand attributed to environmental stimuli. The selected approach for classification, does not restrict the possible number of labels but the cardinality of L should be finite.

The dataset used in this thesis, $D = \langle E_1, E_2, \ldots, E_n \rangle$, consists of a set of labeled multi-dimensional time series. Each time series has a fixed time duration given by the duration of the n-back task, and all of the labeled time series have the same dimensionality, i.e. all given attributes for an event have to be present in all labeled time series.

Definition 4 The *distance* $Dist(A, B)$ is a function that takes two time series (or time series subsequences) as inputs A and B and returns a positive value R, which is the distance from A to B. A representative example is the *euclidean distance*, defined as:

$$Dist(A,B) = \sqrt{\sum_{i=1}^{n} (A[i]) - B[i])^2}$$

The distance function is useful to determine the similarity between time series. Some algorithms such as k-Nearest-Neighbor use a distance measure to group similar instances.

In this thesis time series classification, each sequence has only one class label. Depending on the experimental setting, the whole sequence is available to the classifier before classification (*offline*), or instances arrive as they are recorded (*online*) and the classifier has to deal with the fact that data arrives continuously. This thesis considers both cases, first an approach for *offline* classification is reviewed (Symbolic Aggregate Approximation SAX) followed by an extension of this approach, which deals with continuous measurements (Time Series Bitmaps TSBs).

Different methods can be used to classify time series data. Xing et al. [89], Deshpande and Karypis [19] propose three large categories: feature based, distance based and model based classification.

FEATURE BASED CLASSIFICATION Traditional classification techniques such as decision trees, support vector machines and neural networks have also the ability to classify time series datasets, when sequences are transformed into a feature vector suitable for the algorithm. Feature selection plays an important role for this methods. Different transformation method can be applied based on the nature of the sequence. For instance, for symbolic sequences the simplest way is to treat each element as a feature, e.g. sequence $AGGA$ is transformed a s a vector $\langle A, G, G, A\rangle$. Nevertheless, such a transformation cannot capture the sequence nature of the data. Xing et al. [89] examines different transformations meant to keep order of the elements such as k-grams, which are short sub-sequences of k consecutive symbols. The feature vector for this transformation will be either the presence or the absence of the k-gram or the frequency of the k-grams.

Applying the same transformation for numeric sequences requires discretization of the data which may cause information loss. Xing et al. [89] cites Ye and Keogh [90] on time series shapelets: a feature selection approach that can be applied directly to the numeric data. Time series shapelets extracts time series subsequences that can maximally represent a class and separate the data according to a distance measure threshold. Laguna et al. [45] on the other hand, proposes a dynamic sliding window approach to generate features from numeric data. A *sliding window* is a user-defined subsequence of size w slid across a time series T of length m. Each sliding window can have an overlapping factor with respect to contiguous windows. For instance, having a time series of $length = 120$, and a sliding window of length $w = 60$ and a overlapping factor of 50%, three windows are created in a range of 1-60, 30-90 and 60-120 respectively. Out of each window, one or more features are extracted. Features can be as simple as the average value for all the points in the window w, or a more complex calculation such as a trend value. Lin et al. [51] present an interesting perspective on using sliding window for clustering and classification.

DISTANCE BASED CLASSIFICATION Distance bases classification methods use a distance function to measure the similarity between time series. The distance function allows the use of classification techniques such as K-Nearest-Neighbor, Dynamic Time Warping (DTW) and Support Vector Machines with local alignment kernel [89]. KNN is an instance base learner that does not compute a learning model in the training phase, instead all the computational expensive calculations are deferred to the classification moment. A general classification process using KNN is as follows: Given a labeled time series T^d, a positive integer k, and a new sequence s to be classified, the KNN classifier uses a distance measure to group the k nearest neighbors of s in T^d and returns the dominating class label as the label of s. Different distance function have been proposed but the euclidean distance (see *definition 4*) is widely adopted. Keogh and Kasetty [38] reported that using 1-Nearest-Neighbor to the Cylinder-Bell-Funnel and Control-Chart dataset, euclidean distance performs surprisingly good compared to other similarity functions. A major drawback of euclidean distance is its sensitivity to distortions in the time dimension *[89]*, i.e. time series that have a similar structure but differ in the captured time, might be misclassified if the similarity is measured using euclidean distance.

Dynamic Time Warping (DTW) on the other hand, is a technique meant to overcome the major drawback of euclidean distance and does not require two time series of the same length. The main idea of DTW is to align two sequences and obtain their similarity measure (see more in *[89]*).

Support Vector Machines (SVMs) is another statistical learning technique that has been successfully used for time series data classification *[89]*. The idea behind SVM is to transform the time series data into a feature space and construct an hyperplane that maximizes the margin to separate to classes. SVMs use the underlying kernel methods, which serves to map the sequence to a high dimension feature space. The kernel function can be also viewed as the similarity between two sequences [89]. Defining the kernel function is not a trivial work. Xing et al. [89] provides an useful review of kernel functions used in the time series context.

MODEL BASED CLASSIFICATION Feature based and distance based classification methods utilize *discriminative* algorithms such as SVM or Linear Regression to *discriminate* between different classes. On the other hand, model based classification methods also knows as *generative* models, are a way of linking statistical models to classification problems. They rely on the assumption that each class time series is generated by a model M. The basic idea is to first model the joint distribution $P(X, Y)$ which is described as a set of parameters. During the training phase, the parameters of M are learned. In the classification process, the posterior probability of the new sequence is calculated and assigned to the class with the highest likelihood. A wide range of modeling technique are known: Naiive Bayes, Gaussian Mixture Models and Hidden Markov Models.

Naiive Bayes is the simplest generative model [89]. The fundamental assumption is that the input features are independent from each other. Similar to other generative methods, in the learning step the algorithm estimates each class prior probabilities (the probability of a measurement being of a certain class before collecting the new data) using the relative frequency in the training data. In the classification phase, the posterior probability $P(X|c_i)$ for each feature value $x_j$is calculated. Finally, $P(X|c_i)P(c_i)$

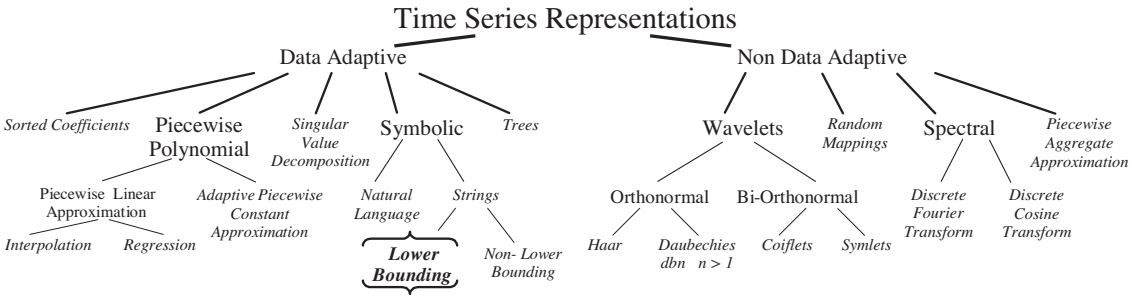

Figure 4.1: A hierarchy of various time series representations based on the literature review in Lin et al. [52]. The leafs represent the actual representation, while the nodes categorize the approaches.

is computed for each class c_i and X is classified according to the maximum posteriori hypothesis. The simplicity of Naiive Bayes is one of its strongest advantages. However, the independence assumption on the features is hard to keep in many practical datasets [89]. Hidden Markov Models can model the dependence among the features in the dataset.

4.1.2 *Symbolic Aggregate Approximation (SAX)*

Time series classification has an increasing interest among the data mining community. Due to its natural temporal ordering, time series mining methods are distinct from other common data mining problems, in which no order in the observation is present. One way to deal with time series data, is transforming the representation of the data. Several time series representations have been proposed. Lin et al. [50] provides a hierarchy of various time series representations (see Figure 4.1). The choice of representation affects efficiency and complexity of time series analysis. As depicted in Figure 4.1, the leaf from the symbolic-string-lower bounding path is bolded; such a representation allows the use of algorithms from the text retrieval community, to analyze time series data. The lower bounding property is fundamental because it allows to run data mining algorithms in the symbolic representation, and produce identical results as the algorithms that use the original data.

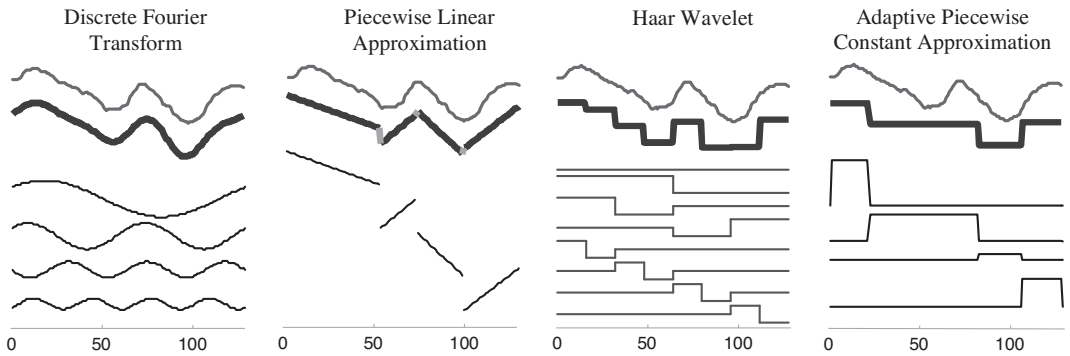

Figure 4.2: Four of the most common representation of time series data. The original time series on the top, followed by the respective representation in bold. At the bottom, the decomposition of the representation as a set of linear functions (extracted from Lin et al. [50])

Several time series representations have been proposed, including Discrete Fourier Transform (DFT) [23], the Discrete Wavelet Transform (DWT) [10], Piecewise Linear, and Piecewise Constant models (PAA) [39], Adaptive Piecewise Constant models (APCA) [39], and Singular Value Decomposition (SVD) [39]. Figure 4.2 illustrates four of the most common representations. Each representation has different strengths and weaknesses, but in terms of their power for time series identification, there is not much difference [50]. A property shared among all the above representations is the fact that they are real valued. Real valued representation are limited in algorithms and data structures, for instance defining the probability of observing a set of wavelet coefficients is zero for any real value [50]. Furthermore, non of the above representations lower bounds a distance measure defined in the original data. Lin et al. [50, 52] propose a lower bounding and dimensionality reduction approach called Symbolic Aggregate Approximation (SAX), which represents time series data as a sequence of symbols.

SAX has been proven to be useful several data mining areas including classification, clustering, indexing and anomaly detection [50, 52, 41, 76]. Because the dataset analyzed in the data mining community tend to be very large, the main reasons for the increasing popularity include numerosity reduction, dimensionality reduction and lower bounding distance. Faloutsos et al. [23] propose a simple and generic algorithm to mine large data sets. First an approximation of the original data that fits into main memory has to be created. Then, the task at hand is solved using the data approximation. The final step is to confirm the solution accessing the original data (as few times as possible). SAX fits into this general approach by approximating the original sequence using Piecewise Aggregate Approximation (PAA) which is then discretize as string. A distance measure for SAX strings is also provided and the lower bounding feature ensures that the results are consistent with algorithms applied to the original data.

SAX transforms a time series of length n into a string of length w, ($w < n$, normally $w \ll n$). The resulting string is formed using an alphabet of arbitrary length a, where $a > 2$ [50]. The pseudocode for transforming a time series to a SAX representation is outlined in Table 4.1. A simple implementation of such a discretization algorithm has three parameters: the original time series *(N)*, number of symbols in each window *(n)*, alphabet size *(a)* and window length *(w)*. In a more sophisticated implementation, this algorithm should be able to handle a numerosity reduction option *(nr)*. In the simplistic version of the algorithm, $nr = 0$ (include all words). Lin et al. [50] propose four different numerosity reduction levels: level 0 include all words, level 1 records only words that are different from the previous word, level 2 records words which min_dist is greater that zero and level 4 records only words if the subsequence is not monotonic (see [50] for a detailed description).

The input parameters to this algorithm are the original time series `N`, number of symbols in a window `n`, alphabet size `a` and window length `w`. The algorithm begins by creating two arrays that will hold the data being processed. The `norm_data` array (line 2) of the same size as `N`, holds the data after being z-normalized. The `paa_coeff` array (line 3) of size `n` holds the resulting PAA dimensionality reduction coefficients. The `window` array (line 4) of size `w` holds the extracted sliding window for each iteration. The variable `num_words` (line 5) stores the total number of words that are generated by sliding a window of size `w` through the normalized data. The two dimensional ar-

```
1   Function SAXdistretization(N,n,a,w)
2   norm_data[N.size]
3   paa_coeff[n]
4   window[w]
5   num_words = (N.size() - w) + 1
6   sax_words[n times num_words]
7   norm_data = normalize(N)
8   i = 1
9   while i + w < N.size()
10      window = norm_data[i : i + w]
11      paa_coeff = PAA(window,n)
12      sax_words[i][:] = discretize(paa_coeff)
13      i++
```

Table 4.1: Algorithm pseudocode to transform a univariate time series to a SAX representation.

ray `sax_words` (line 6) contains the words resulting from each window discretization process.

The first step is to normalize the data to have a mean of zero and a standard deviation of one (line 7); the result is stored in the `norm_data` array. The while loop iterates over the normalized data. In each iteration, a window containing `w` observations is extracted (line 10). To calculate the PAA coefficients (line 11), the `window` array is divided into `n` equal sized "frames". The mean value of the data within a frame is calculated resulting in a PAA coefficient for each frame [50]. The resulting vector is stored in the `paa_coeff` array.

The discretization process applied to the PAA coefficients produces symbols with equiprobability. This is achieved by defining *breakpoints*, that are a list of sorted numbers $B = \beta_1, \ldots, \beta_{a-1}$ such that the area under a $N(0,1)$ Gaussian curve from β_i to $\beta_{i+1} = 1/a$. These breakpoints can be determined using a statistical table (see Table 4.2). Once the breakpoints have been defined, all the PAA coefficients in the paa_coeff array having a value below the smallest breakpoint are mapped to symbol "a", all coefficients having a value between the smallest and the second smallest breakpoint ($second\_smallest > paa\_coeff \geqq smallest$) are mapped to symbol "b", etc [50].

$\beta \setminus a$	3	4	5	6
β_1	-0.43	-0.67	-0.84	-0.97
β_2	0.43	0	-0.25	-0.43
β_3		0.67	0.25	0
β_4			0.84	0.43
β_5				0.97

Table 4.2: A table that contains the breakpoints that divide the Gaussian distribution in n *(from 3 to 6)* equiprobable regions (extracted from Lin et al. [50]).

Figure 4.3 depicts the result of running the algorithm using following parameters: *N=128; n=8; a=3; w=128.* Since the word length and the original time series are

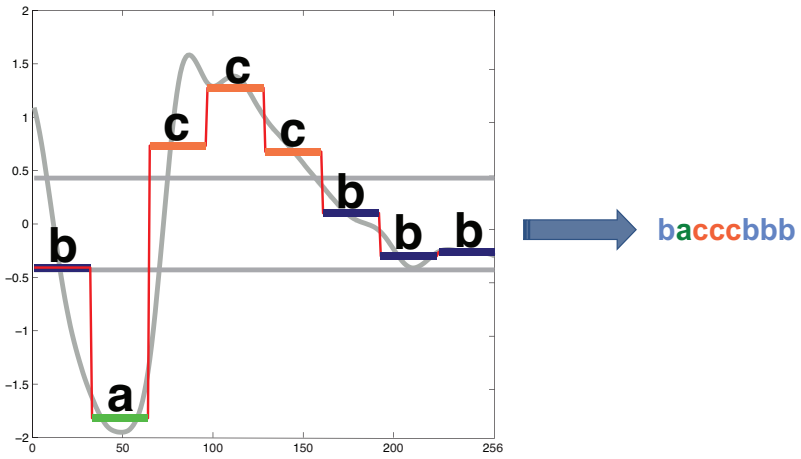

Figure 4.3: A SCL time series of length 256. It is intended to graphically illustrate the SAXdiscretization algorithm using parameters *N=256; n=8; a=3; w=256.* In this example the time series is mapped to the word bacccbbb.

the same length, only one *sax word* is obtained: ***bacccbbb***. In this example, the PAA representation averages 16 observation to obtain 8 segments which are then discretize as one sax word. The breakpoints are exemplified as two gray horizontal lines.

Lin et al. [50] define a distance measure based on the Euclidean distance. Lin et al. [52] provide mathematical proof that the proposed distance (MINDIST) measure lower-bounds the Euclidean distance in two steps: First proofing that the PAA distance lower-bounds the Euclidean distance. Next, the authors proof that the MINDIST lower-bounds the PAA distance. Finally, by transitivity is showed that the MINDIST lower-bounds the Euclidean distance.

The MINDIST function returns the minimum distance between the original time series of two sax words. MINDIST is defined for two symbolic representations (P', Q'), original time series length N and n symbols for each word:

$$MINDIST(Q', C') \equiv \sqrt{\frac{N}{n}} \sqrt{\sum_{i=1}^{n} (dist(q'_i, c'_i))^2}$$

The function uses the sub-function $dist()$ which can be implemented using a lookup table such as Table 4.3.

	a	b	c	d
a	0	0	0.67	1.34
b	0	0	0	0.67
c	0.67	0	0	0
d	1.34	0.67	0	0

Table 4.3: This is a lookup table to calculate the distance between two symbols. The alphabet size used for this table is 4, i.e. $a = 4$. The distance between two symbols is stored in the corresponding (row, column). For instance, *dist(a,d) = 1.34.*

For instance, the MINDIST for two sax words: *ababcaca* and *cdccdaad* having *N=256,n=8* and *a=4* is:

$$\sqrt{\frac{256}{8}}\sqrt{\sum_{i=1}^{8}(dist(q',c'))^2} = \sqrt{32}\sqrt{(0.67+0.67+0.67+0+0+0+0.67+1.34)} \approx 11.34$$

Applying SAX to a univariate time series produces one or more sax words of equal length. Each word could be used as a feature along with a distance based classification algorithm such as K-Nearest-Neighbor. One has to bear in mind that if the numerosity reduction level is equals zero, then two consecutive windows will produce the same word. Some algorithms like bag of patterns [49], construct a matrix upon the frequency of the words. Thus if two consecutive windows produce the same words, some cells might contain larger frequency counts than the expected count. This document uses multivariate time series (see Definition 2) from two signals: SCL and Heart Rate. With that in mind, the distance measure $D\_MINDIST$ between two arrays P^d and Q^d containing n sax words of arbitrary length is defined as the average MINDIST value of each word.

$$D\_MINDIST = \frac{1}{n}\sum_{i=1}^{n} MINDIST(P_i, Q_i)$$

It is assumed that the words are concatenated, and that the length of each word in the array is known. Figure 4.4 illustrates the concept with two concatenated sax words, the first eight symbols in each line belong to the first word and the next six symbols to the second word.

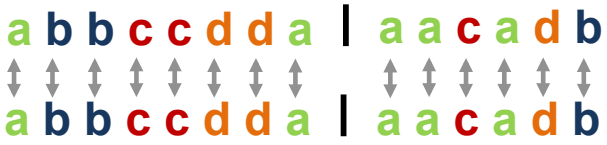

Figure 4.4: Illustration of the multidimensional sax words distance.

Fishel et al. [24] reports that the method used to detect the physiological baseline influences the capability of an automated system to recognize changes in human physiology. The main advantage of using a symbolic representation for physiological signals, as opposed as using a numerical feature extraction approach, is that the challenge of calculating a stable baseline to compare the changes against, is not required anymore. SAX is able to capture the shape of the signals by sliding a window throughout the whole sequence, which allows a classifier to categorize the physiological changes according to the shape instead of a numerical score.

On the other hand, SAX normalizes the input signal to have zero mean and one standard deviation, causing the numeric value to be meaningless. If the cognitive workload manager needs to be able to, not just identify that the cognitive workload level of the driver increased, but to determine the current level (“low”, “medium”, “high”), classifying using the shape of the signal is probably not the best approach unless there is a consistent change in shapes that characterizes each cognitive workload level.

4.1.3 *Towards real time classification: Time Series Bitmaps*

In an initial step this thesis establishes the feasibility to use SAX, a discretization approach, to characterize the cognitive workload level of a driver. However, another relevant goal of this paper is to find a suitable approach for real time driver's state detection. In order to preserve the shape of a long time series, SAX slides a window of arbitrary length over the entire sequence. In each iteration, the sliding window is discretized to a SAX word. In order to classify a time series as part of one category c, one simple approach is to compare a time series s (n *sax words*) to all time series in each category, i.e. measure the MINDIST of each word in the time series s to the corresponding word in the compared time series. The distance measure is the average of all MINDIST measures.

SAX is not intended to be used in a real time environment, where the data arrives continuously [43]. One of the assumptions of SAX is, that the length of the time series remains constantly. However, Kasetty et al. [36] propose an algorithm to maintain Time Series Bitmaps (TSBs) in constant time (see Table 4.1.3). This method propose an approach to deal with continuously incoming data. TSBs were originally proposed by Kumar et al. [43] as a visualization technique for discrete sequences such as SAX words. In the same paper, the authors successfully used TSBs for time series classification, clustering and anomaly detection. Kasetty et al. [36] on the other hand, monitored insects using senors streaming data. The authors used TSBs to keep a constant summary of the streaming data and to distinguish between different insect behaviors. This thesis considers using TSBs as a viable approach to summarize the continuous incoming signals (Heart rate and Electrodermal Activity).

Time Series Bitmaps (TSBs) are inspired by a representation used to draw fractals called *Chaos game representation.* In essence a TSB is a summarized representation of a SAX string which counts the frequency of SAX "subwords" of length L, and builds a squared grid of size $2^L$where each cell contains the count of a SAX subword (see Figure 4.5). It is important to note, that the *Chaos game representation* is defined for sequences with an alphabet size of four. Thus, using TSBs intrinsically sets the alphabet size for the discretization process of the time series to a value of four.

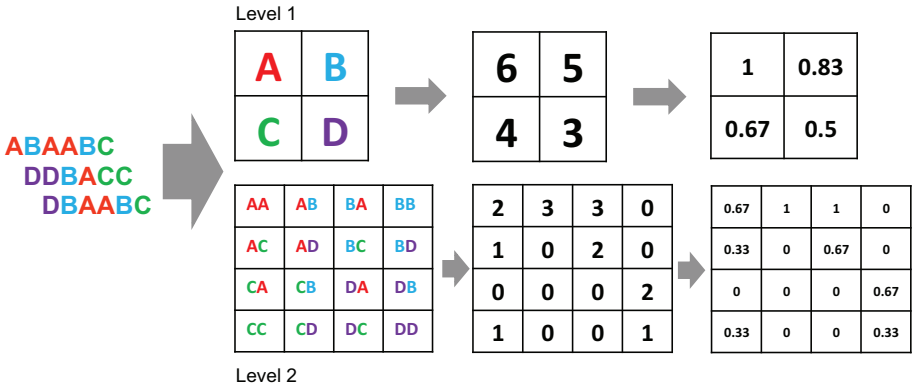

Figure 4.5: *Left*: Three SAX strings are mapped to time series bitmaps. *Middle*: Two different levels of time series bitmaps are constructed for the input strings. *Right*: time series bitmaps normalized by diving each cell by the largest value (adapted from Kumar et al. [43], Kasetty et al. [36]).

As depicted in Figure 4.5, creating TSBs is a three step process. First the desired length of the subwords has to be chosen. It is important to consider that increasing the length of the subwords will increase the size of the frequency grid. The second step is to count frequencies of subwords. For instance, if the level is set to one, frequencies are the raw counts of the four symbols. In the case where the level value is two, all the subwords of size two are counted (aa, ab, ac, etc). It is important to note that the count of the subwords are taken from individual subwords, i.e. in Figure 4.5 the last symbol of the first word is C and the first symbol of the second word is D, however this is not considered an occurrence of the subword "CD" [43]. Because the time series in a dataset might be from different lengths, the third and final step is to normalize the frequencies by dividing them by the largest value. Kumar et al. [43] perform another step that consisted on mapping the final values to colors. This thesis omits this last step, since the utility from the proposed time series bitmaps is for classification and not visualization.

In order to be able to compare time series bitmaps, a distance measure between two TSBs has to be defined. This thesis uses the euclidean distance to compare two time series A and B, i.e. the cell to cell distance value:

$$dist(A, B) = \sqrt{\sum_{i=1}^{2L}\sum_{j=1}^{2L}(A_{ij}, B_{ij})}$$

It is important to note that time series bitmap representation for SAX strings is abstract. As opposed to SAX, where the original time series shape was maintained, using time series bitmaps looses the structure of the time series. Another feature of time series bitmaps is the ability to represent a discretized time series in constant space and structure. The utility of the TSBs is their ability to efficiently compare and contrast them [43].

As already mentioned, Kasetty et al. [36] introduced an algorithm capable of maintaining TSBs in constant time. This thesis acknowledges the potential use of the proposed algorithm (see Table 4.1.3) to distinguish different levels of cognitive workload in a real time driving environment. Refer to Kasetty et al. [36] for a detailed description of the algorithm, as well as an illustration of how it updates the history buffer to save time and space.

4.1.4 *Evaluation and selection of the best model*

One of the fundamental task in machine learning is to construct a function that can accurately describe the data being analyzed. Various techniques in the field of machine learning can be used to learn an appropriate model for the data and problem at hand. One of the most popular problems in machine learning and data mining is the *classification* problem, that is the problem of predicting to which of a set of classes a new observation belongs, on the basis of a training set of observations that have a defined class (also called labeled observations). An algorithm that provides classification capabilities is known as *classifier*, which is a mathematical function that maps the input data to a defined class.

In general classification is a two step process; first comes the *learning step*, where the *classification model* is trained and then the *classification step*, where the model is

```
1  Function classifyTSBs(N,n,a,historySize) historyBuffer[historySize][n]
2  curtime series[N]
3  curTSB[a times a] = 0
4  input = getInput()
5  while curtimeseries.size() < N and
6              input != EndOfFile:
7        curtimeseries.append(input)
8        input = getInput()
9  curSAXWord = sax(curtimeseries,N,n,a) incrementTSB(curTSB,curSAXWord)
10 historyBuffer.append(curSAXWord)
11 while historyBuffer.size() < historySize and
12             input != EndOfFile:
13       curtimeseries.pop()
14       curtimeseries.append(input)
15       curSAXWord = sax(curtimeseries,N,n,a)
16       incrementTSB(curTSB,curSAXWord)
17       historyBuffer.append(curSAXWord)
18       input = getInput()
19 classify(curTSB)
20 while input != EndOfFile:
21         curtimeseries.pop()
22         curtimeseries.append(input)
23         curSAXWord = sax(curtimeseries,N,n,a)
24         removedWord = historyBuffer.pop()
25         decrementTSB(curTSB,removeWord)
26         historyBuffer.append(curSAXWord)
27         incrementTSB(curTSB,curSAXWord)
28         classify(curTSB)
29         input = getInput()
30
31
```

Table 4.4: Pseudocode to maintain TSBs in constant time (extracted from Kasetty et al. [36]).

used to predict the class labels for the input data [29, p. 328]. Several classification algorithms including Logistic Regression, Naive Bayes, Neural Networks, Decision Trees, etc, can be used to build a classification model. Once the classification model is obtained, it is important to estimate the performance of the selected model, i.e. estimate the number of correctly and incorrectly classified observations.

Performance metrics such as accuracy, sensitivity, precision and F-score (Section 4.1.4.1) are used to establish a classifier's performance. Model validation techniques such as Cross-Validation (section 4.1.4.2) use performance metrics to assess how a classifier can generalize to an independent dataset. Finally, with a trustworthy accuracy value, a statistical test can be employed to select the model that has the best generalization prospective, i.e. that will perform at least as good with the known test data as with unknown data (section 4.1.4.3).

4.1.4.1 *Performance metrics*

This section introduces various metrics that are utilized to measure and compare the performance of classifiers. Table **??** summarizes the most common performance metrics. They include accuracy, sensitivity, specificity, precision and F-score. The weight that each metric carries for a decision depends on the desired characteristic of the classifier. Results are most frequently evaluated using accuracy levels. However it is important to consider other metrics such as precision and sensitivity. Han et al. [29, p. 368] note that an inverse relationship between precision and sensitivity is often observed. Bearing that in mind is crucial for evaluating the desired behavior of a system. For example, if the system in question is safety relevant (e.g. airbag triggering), the cost of missing a instance of the class (low sensitivity) is very high. But in other application scenarios such as a credit card fraud detection system, having some transactions falsely detected as fraudulent (low precision) is a bearable situation.

In the case of a classifier used to detect the cognitive workload state of a driver, the decision maximizing the precision or the sensitivity rely on system that is using the information. So, if the classification results are being used by an autonomous deceleration system that is triggered if the driver is in a high cognitive workload level, then a perfect precision score is desired. However, when the system in question is in charge of deactivating the multimedia interface of the car (e.g. radio, navigation system), a balanced precision and sensitivity rate would make the system more useful.

In order to discuss this section, some terminology has to be introduced:

- **True positives** *(TP)*: The positive instances that were correctly identified by the classifier.
- **True negatives** *(TN)*: The negative instances that were correctly identified by the classifier.
- **False positives** *(FP)*: The negative instances that were incorrectly identified as positive by the classifier.
- **False negatives** *(FN)*: The positive instances that were incorrectly identified as negative by the classifier.

The rest of this section describes the performance measures depicted in Table **??**.

ACCURACY Is the proportion of correctly classified instances by the classifier. That is,

$$accuracy = \frac{TP + TN}{P + N}$$

Another name given to accuracy is recognition rate, since it indicates the degree of correctly recognized instances of the different classes. Accuracy levels can be misleading if the class distribution in the training set was unbalanced. For instance, if the training set contains 95% instances of class A and only 5% instances of class B a 95% accuracy might not be acceptable since the classifier could be misclassifying all instances of class B.

ERROR RATE Is also known as misclassification rate of a classifier, M, and can be calculated as 1 - accuracy(M). An alternative way to compute it is:

$$error\,rate = \frac{FP + FN}{P + N}$$

SENSITIVITY Is also referred as true positive rate, recall or hit rate. Given a class C, the true positive rate is the proportion of instances belonging to class C correctly identified by the classifier as such. Sensitivity can be calculated as:

$$sensitivity = \frac{TP_C}{P_C}$$

The true positive rate quantifies the completeness of the classifier, i.e. how many of the instances belonging to class C were classified as such.

FALSE POSITIVE RATE The false positive rate for a class C is the number of instances not belonging to class C incorrectly classified as class C divided by the number of all instances not belonging to C. It can be computed as:

$$false\,positive = \frac{FP_C}{N_C}$$

SPECIFICITY Is the proportion of instances not belonging to class C that are correctly identified by the classifier. It can be computed as:

$$specificity = \frac{FN_C}{N_C}$$

PRECISION The precision can be interpreted as a measure of exactness, i.e. the proportion of instances identified by the classifier as belonging to class C that are actually such. It can be calculated as:

$$precision = \frac{TP_C}{TP_C + FP_C}$$

F-SCORE A function that considers both precision and sensitivity when evaluating classifiers is the f-score. For a given class C, the F-score is computed as:

$$F - score = \frac{2 \times precision_C \times recall_C}{precision_C + recall_C}$$

Han et al. [29, p. 369] alludes additional aspects which can help to compare classifiers such as *speed, robustness, scalability* and *interpretability.*

4.1.4.2 *Assessing model performance*

In the fields of machine learning and data mining, evaluating the performance of a classifier is not a trivial task. Witten et al. [87, p. 147] suggests that evaluating the classifier's performance using the same data as the training data would result in a "hopelessly optimistic" performance. Thus, the most basic approach to evaluate an algorithm consist on splitting the dataset in two, the training and the test set. With the training set, the classifier will find the parameters that performs best, i.e. highest accuracy, highest sensitivity, etc. Once the parameters are set, the test data is used to evaluate how well the classifier is actually performing. The main idea is being able to predict how a classifier will perform in classifying new data based on old data. It is important that test set is not used in any form to create the classifier. Other approaches to predict model performance include holdout method, cross-validation and bootstrap [29, 87]. This section introduces tow of the most common methodologies for assessing a model performance.

HOLDOUT METHOD Evaluating a model involves splitting the dataset in two independent subsets: *training* and *test* set. The training set is used to derive a model; the test set uses the derived model to estimate the performance of the classifier using some performance metrics (see 4.1.4.1) [29, p. 370]. In the holdout method, the dataset is *randomly* partitioned into the training and the test set. Typically, two thirds of the data is allocated to the training set and one third to the test set. Using the training data, a model is derived and with the test set the model's accuracy is estimated.

Due to the randomized partitioning of the test and training set, the test set might be missing data of one or more classes. Thus, an advance version uses *stratification* to ensure that each class in the data set is represented with similar proportions in the train and test set.

CROSS VALIDATION Han et al. [29, p. 370] notes that the error estimate obtained using the holdout method is pessimistic since only part of the data was used to derive the model. Therefore, the *random subsampling* method proposes to repeat the holdout method k times. The overall accuracy is the average accuracy from each iteration. Using this scenario, a more reliable estimation is obtained, but there still the chance that two or more test sets overlap. A solution to this problem can be achieved using *cross-validation.*

In the k-fold cross-validation, the initial dataset is randomly partitioned into k mutually exclusive and equally sized subsets or "folds", $D_1, D_2, \ldots, D_k$. In order to estimate the classifier performance, training and testing is performed k times [29, p. 370]. In each iteration i, D_i is the test set and the rest folds conform the training set. This

means that, in the first iteration D_1 servers as the test set and $D_2, \ldots, D_k$ are used as the training set to build the model; in the second iterationD_2 servers as the test set and $D_1, D_3 \ldots, D_k$ are used as the training set to build the model and so on. Unlike the holdout method and random subsampling, here the folds are built as an initial step to guarantee independent test sets.

Similar to holdout and random subsampling, the folds are built in a randomizes fashion, allowing over and under-sampling to be present. Therefore, *stratified cross-validation* is typically a recommended approach [29, p. 371]. Witten et al. [87, p. 153] discuss that the recommended number of folds to use is 10 because several different studies, with different data sets and learning techniques, have shown that 10 folds produce a reliable error estimate.

4.1.4.3 *Model selection with cross-validation*

Model selection involves comparing the performance estimation (e.g. cross-validation accuracy results) to other models. This might seem like a trivial task of choosing the model that presents the lowest error rate or the highest sensitivity. Witten et al. [87, 156] notes that selecting the model with the best performance might be misleading because it heavily depends on the error estimation. Two main methodologies can be found in the literature [29, 75, 87], statistical tests and cost-benefit comparison. This thesis uses statistical tests due to its simplicity and understandability.

Han et al. [29, p. 372-373] use *paired Student's t-test* (if the data set used for both models is the same) to determine if the average accuracy of one model is statistical significantly greater than the average accuracy of other model. In this case, the term average means the cross-validation results that represent the average accuracy of the k-folds. Given two models M_1 and M_2, the authors propose to perform 10-fold cross-validation, 10 times, with a different data partitioning for each time but using the same data from training and testing the two models. They average the 10 error rates from each M_1 and M_2, to obtain the mean error rate ($\overline{err}(M_1)$ and $\overline{err}(M_2)$) for each model and the variance denoted as $var(M_1 - M_2)$. Using the *t-test* to compute the *t-statistic with k - 1 degrees of freedom* for k samples ($k = 10$ in this example) as follows:

$$t = \frac{\overline{err}(M_1) - \overline{err}(M_2)}{\sqrt{var(M_1 - M_2)/k}}$$

where

$$var(M1 - M2) = \frac{1}{k}\sum_{i=1}^{k}[err(M_1)_i - err(M_2)_i - (\overline{err}(M_1) - \overline{err}(M_2))]^2$$

The null hypothesis in this case is that the mean error rate $\overline{err}(M_1)$ does not differ significantly from $\overline{err}(M_2)$. To determine if the null hypothesis can be rejected, the t statistic needs to be computed and a significance level needs to be selected. In practice, a significance level of 5% or 1% is used. Using a table for the t-distribution the z value or confidence limit ($z = \frac{sig}{2}$) must be located. If $t > z$ or $t < -z$, then the null hypothesis can be rejected, meaning that the error means differ significantly. Otherwise, the conclusion is that any difference between M_1 and M_2 can attributed to chance.

Witten et al. [87, p. 157-159] follows a similar approach to compare models. Instead of using the standard version of the paired Student's t-test, the author proposes a variation of this statistic called *corrected resampled t-test.* In this case the t-statistic is computed as follow:

$$t = \frac{\overline{d}}{\sqrt{\left(\frac{1}{k} + \frac{n_2}{n_1}\right)\sigma_d^2}}$$

where $\overline{d}$ is mean difference of all 10 error rates (one error rate for each 10-fold cross validation run). For 10-fold cross-validation repeated 10 times *k=100*; n_1 is the percentage of instances used for training and n_2 for testing ($n_2/n_1 = 0.1/0.9$), and σ_d^2 is the variance based on 100 samples.

Salzberg [75] considers not only the accuracy levels to compare two models, but includes the number of examples that one model got right and the other wrong, the number that both got right and the number that both got wrong. The author uses a two-sided test with the binomial distribution to test is if one model performs better than other. Assuming that a series if independent tests (one test is one cross-validation iteration) is available. Let n be the number of examples for which the the models produce a different output. Let s (successes) be the number of times $M_1 > M_2$, and f (failures) be the number of times that $M_2 > M_1$. The expected value *E(s)* is equals $0.5n$ in case that the two models perform equally well. Suppose that $s > f$, meaning that M_1seems to be performing better than M_2. One would like to calculate the probability that M_1 is better than M_2 at least as many times as observe in the experiment. The probability of s successes in n trial can be computed as:

$$\frac{n!}{s!\,(n-s)!)} p_s q^{n-s}$$

If no difference between two models is expected, then $p = q = 0.5$. Suppose that in a total of 100 trials (10 times 10-fold cross-validation, in $n = 50$ meaning that the the algorithms differed in 50 examples; $s = 35$ means that M_1 was correct 35 cases where M_2 was wrong. Using this information, the probability of this results can be computed as follow:

$$\sum_{s=35}^{50} \frac{n!}{s!\,(n-s)!} (0.5)^n = 0.0032$$

Since $0.0032 \ll 0.05$ (assuming that a significance level of 5% is required), the null hypothesis can be rejected with high confidence. The author acknowledges using binomial test carries some disadvantages such that it doesn't handle quantitative differences between models, nor does it handle more than models and it doesn't consider the frequency of agreement between two models.

Salzberg [75] recommend the following approach to compare different models using cross-validation in a nested loop of cross-validation iterations:

1. Choose the algorithms to compare.
2. Choose a dataset that accentuates the strengths of the algorithms.
3. Divide the dataset D into k subset for cross-validation.

4. Run cross-validation as follow (outer cross-validation loop):
    a) For each k subset of the dataset D, create the training set $T = D - k$
    b) Divide each training set T into two smaller subsets T_1 and T_2. T_1 is used for training and T_2 for parameter *tuning* on each model. (Inner cross-validation loop)
    c) Use the optimized parameters and the complete training set T to train the classifiers.
    d) Measure the performance of each classifier using the test set $Y = D - T$
    e) Overall accuracy is averaged across all Y partitions. Using the Y values, one can also calculate the variance of the models.
5. Compare the models using the above described binomial test.

Comparing different classification schemes is not only useful for model selection, a similar approach can be followed to find the best parameters for an algorithm as showed in [75]. When using cross-validation one has to be careful that the training data is independent from the testing data in each iteration. This is particularly important in cases where sequential data is present (as in the case of this thesis analysis). To circumvent the data dependency problem approaches such as *hv-block* cross-validation [67] have been introduced. Although this thesis uses time series data, the design of the experiments (see Chapter 6) ensure that the data in the cross-validation training and test samples remains independent. Finally Salzberg [75] stress the importance of not running cross validation multiple times using the same dataset without adjusting the significance levels of the statistical tests accordingly, i.e. the classifier that based on the given performance metrics, is best suitable to categorize cognitive workload.

4.2 CLASSIFICATION ENGINE

So far, this chapter introduced several concepts including as Symbolic Aggregate Approximation (SAX) of time series, Time Series Bitmaps and an approach to select and tune a classifier. This section combines the approaches to build a classification engine that discretizes a driver's Heart Rate (HR) and Skin Conductance Level (SCL) signals in order to predict the cognitive workload state. Using TSBs, the the proposed engine could potentially deal with streaming data. The suggested classification engine is depicted in Figure 4.6.

The classification engine performs distinct operations to transform the raw incoming data (ECG, SCL) into a viable time series for the SAX discretization module. Several pre-processing operations take place in this initial phase (see red block in Figure 4.6). The SAX discretization module, take a time series and the corresponding discretization parameters such as the sliding window size, the number of discrete segments for each window, i.e. sax word length, and the alphabet size to generate one or more SAX words. The parameter generator module (see green block in Figure 4.6) produces several combinations of the SAX discretization input parameters. This allows the Validation Engine determine the best parameters based on the accuracy level.

The TSB builder (see light blue dotted block in Figure 4.6) is an optional module that creates Time Series Bitmaps if desired. In order to produce such bitmaps, it is necessary that the alphabet size of the input sax word is equals four.

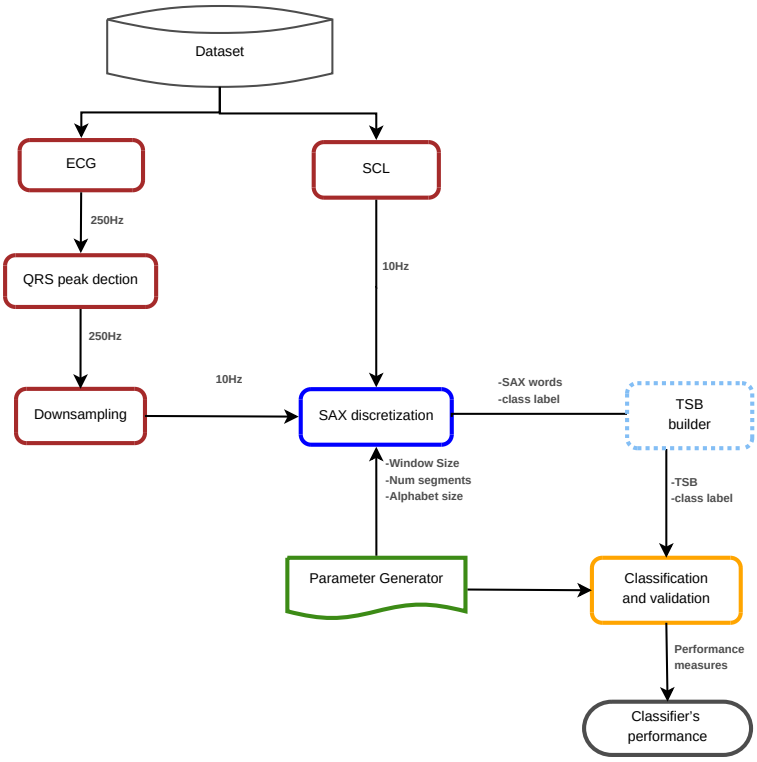

Figure 4.6: Flow chart of the classification engine to evaluate the feasibility of using SAX to categorize different cognitive workload levels in drivers.

Finally the classification and validation module (see yellow block in Figure 4.6) takes as input either one TSB or a list of sax words for each subject and class label (e.g. subject number 4 and class label "high cognitive" workload). This module is in charge of training and determining the performance of different models. This thesis uses the k-Nearest-Neighbor classifier with the MINDIST function (see Section 4.1) for sax words and the euclidean distance for TSBs. In essence, this module will train and test for each parameter combination (i.e. k and $SAXdiscretization$ parameters) and find the optimal the parameters as suggested by Salzberg [75]. The output will be the classification performance (e.g. accuracy, f-score, specificity, precision, etc.) of each model in each cross validation iteration, which can be compared to find the the best parameters using the aforementioned statistical tests.

5

EXPERIMENTAL SETUP

For assessing the capability of symbolic approaches to correctly classify cognitive workload in an field driving environment, empirical data was utilized from a study previously conducted by the MIT AgeLab. This chapter is intended to describe the apparatus, driving environment and procedure used to conduct the study. A complete description of the data can be found in [57].

5.1 PARTICIPANTS

Participants are required to have held a valid driver's license for at least three years and drive three or more times a week. Have a driving record free of accidents for the past year. The participant group was considered to be relatively healthy based on self-report and specified exclusion criteria such as: hospitalization in the past 6 moths, major cardiac conditions, treatment for mental disorder or neurological problems. Participants scoring less than 26 in the Mini-Mental State assessment [25] were also excluded. The Mini-Mental state assessment examines the cognitive mental status of a person using eleven questions.

Initially 165 participants were recruited from the grater Boston area. Of these, 57 participants were excluded because of reason suchlike: failure to meet requirements upon eligibility review, inability to perform the secondary tasks to criterion prior to driving, overt sleepiness while driving, heavy traffic among others. For a detailed description of the different reason see Reimer et al. [72].

Nine more participants were excluded due to missing data or poor measurement quality in the physiological signal domain. Therefore, a sample of 99 individuals (see Table 5.2) was equally balanced by gender and across the three age groups: 20-29, 40-49, and 60- 69. As in Reimer et al. [72] the age of men and women did not differ significantly within age groups.

5.2 APPARATUS

The on-road study conducted at the MIT AgeLab laboratory used a vehicle called "AwareCar" [70]. The "AwareCar" is an instrumented Volvo XC90 fitted with a customized data acquisition system designed for time synchronized capture and measurement of vehicle, driver and environmental data. The data acquisition system facilitate the synchronization of data from multiple sources. Each data point captured is linked to a specific periods or point in time during the experimental protocol. Pictures of the "AwareCar" are shown in figure 5.1. The data acquisition system used is made up of a number of embedded sensing systems that included a MEDAC System/3 instrumentation unit developed by the NeuroDyne Medical Corporation in Cambridge, Massachusetts. A FeceLAB® 4.5 eye tracking system (Seeing Machines Inc., Tucson,

Figure 5.1: On the left a picture of the AwareCar from the side. On the right a picture inside the AwareCar taken from the back sit.

AZ)[1] Vehicle performance data was logged at 10 Hz, physiological data was recorded at 250 Hz and eye tracking data was collected at 60 Hz. Table 5.1 describes the physiological data captured.

Signal	Description
PPG	Blood volume pulse wave
SCR	Skin conductance response
SCL	Skin conductance level
EKG	Electrocardiograph
HR	Heart rate
IBI	Inter-beat interval
T/R	Skin temperature

Table 5.1: Physiological data captured by MEDAC System/3

Besides capturing data, the data acquisition system included a manual and time-based event trigger that presented predefined secondary task to the driver.

5.2.1 *Driving environment*

Data were collected on Route 93 north of Boston, from south of the intersection with Route 495 and continuing into southern New Hampshire. The posted speed limit was 65 MPH and the number of travel lanes in each direction started at 4 and decreased to 2 over the course of the experiment.

Experiments were conducted mid-to-late morning or during the early afternoon to avoid commuter traffic. Cases where the driver encountered heavy traffic that blocked "steady state" flow or rain that impeded visibility were excluded from the analysis [57, 72].

1 Video recordings of driver behavior and vehicle surroundings were obtained but discarded for our analysis

Age group	Mean (SD in parenthesis)	Females	Males
20-29	23.60 (2.90)	17	18
40-49	45.61 (3.00)	16	16
60-69	62.83 (3.03)	16	15

Table 5.2: Participants sample demographics

5.2.2 *Secondary Tasks*

This thesis main purpose is to distinguish the physiological response of subjects under different cognitive workload levels. The cognitive workload level is not easy to observe, therefore increasingly difficult levels of secondary tasks were used to actively manipulate the workload level. Subjects were tasked to drive on a highway under smooth traffic conditions, and then exposed to a dual task (driving + subsidiary secondary task) to control their cognitive workload. In section 5.3 the exact followed procedure is explained.

There are many variations of subsidiary tasks to estimate the primary task workload [7, 8, 57, 56, 97] : mental arithmetic, digit span, delayed recall of random digits, self-paced generation of random digits among others.

Zeitlin [96] states that an effective subsidiary task has to meet the following requirements:

1. Minimally intrude on primary task performance within the range of conditions covered in the experiment,
2. Have more performance degradation as a function of decreased capacity than the primary task,
3. Show performance changes as a monotonic or predictable function of spare capacity,
4. Discriminate between resources being absorbed, and
5. Permit transient as well as continuous changes in workload to be measured.

Other practical attributed are also required for a field-useful subsidiary task [96]:

- Use alternative input-output modalities to the primary tasks. Because most man-machine control tasks are visual-motor, good subsidiary tasks tend to use auditory inputs and verbal outputs.
- Require minimal learning on the part of the subjecN-back Delayed Recall of Random Digitsts,
- Be resistant to change as a function of repetition,
- Require minimum equipment, and
- Be easy to score.

Delayed digit recall and random digit generation meet most of the above requirements [96]. This thesis uses delayed digit recall n-back[58] tasks which enables to systematically increase the cognitive demand level of an individual. Current research focuses on a binary state detection [30, 31, 37, 17, 79]: low and high load. By using multi-level tasks like the n-back-delayed-digit-recall task, the feasibility of a metric for cognitive load information can be developed. This metric enables detection of various cognitive workload levels. Next section provides a detailed description of the n-back task used in this thesis, this description was extracted from Mehler et al. [59].

N-BACK DELAYED RECALL OF RANDOM DIGITS During the course of the experiment, subjects were exposed to three levels of a delayed response task "n-back" to systematically increase demand on the driver without conflicting with the manual or visual tasks. Zeitlin [96] shows that delayed response tasks is sensitive to change in on-road driving conditions.

Each level of demand was presented to the driver as 4 series of 10 single digits (0 to 9). Each series of 10 stimuli was presented in random order, leaving a 30 second interval between the series. The inter-stimulus interval between each number in a series is of 2.25 seconds from the beginning of a stimulus until the next stimulus. There was a brief pause between each set of stimuli. The total testing period for each level is of 2 minutes. In section 5.3 the exact followed protocol is presented.

Three levels of difficulty of a "n-back" task were used to induce the divers with low, moderate and high levels of secondary workload [58]. In the "0-back" level, the subject had to simply repeat the number immediately after it was presented. The "0-back" test is the minimal cognitive level. Even though this test is not particularly difficult a statistically significant increase in physiological arousal is observed [58, 72]. Figure 5.2 depicts the "0-back" task graphically.

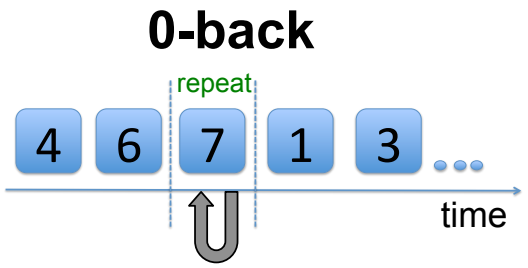

Figure 5.2: "0-back" task graphical representation

In the "1-back" condition, instead of repeating the current number immediately, the subject had to place it in memory and verbalize out loud the number that was presented just prior to the current number as depicted in figure 5.3. This represents an additional step up in the cognitive load since the individual must hold one digit in memory while recalling the previous item presented from memory. The "1- back" appears to have a moderate impact on the subject's physical arousal [72].

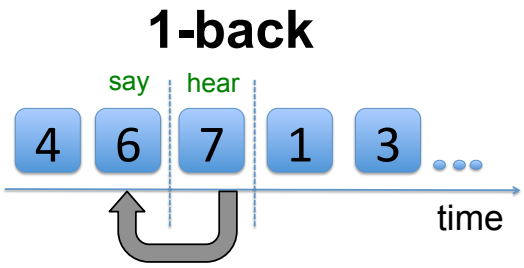

Figure 5.3: "1-back" task graphical representation

The hardest level of demand is the "2-back" form of the task. This level of the "n-back" required subjects to recall from memory and say out loud the number that was presented two numbers prior to the current one. The overall design of the task was meant to increase the cognitive load on the subject both in terms of absolute difficulty and sustained load[58]. Figure 5.4 provides a graphical representation of how the "2-back" test works.

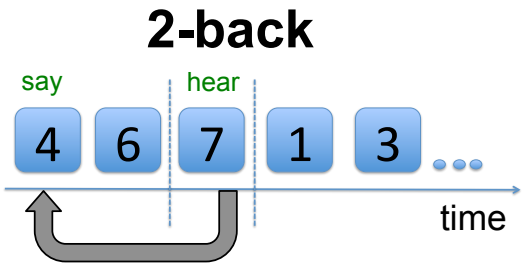

Figure 5.4: "2-back" task graphical representation

The order in which the different "n-back" tasks were presented to the subjects were counterbalanced [58]. Six different protocols originated from this randomization step. Table 5.3 shows an overview of the order in which the "n-back" test was presented depending on the assigned protocol, e.g. for a subject assigned to protocol number one, the first task to complete is "0-back" followed by the "1-back" and ending with the "2-back".

Protocol	**1st "n-back"**	**2nd "n-back"**	**3rd "n-back"**
1	0-back	1-back	2-back
2	2-back	0-back	1-back
3	0-back	2-back	1-back
4	2-back	1-back	0-back
5	1-back	2-back	0-back
6	1-back	0-back	2-back

Table 5.3: n-back task order for all different protocols

5.3 PROCEDURE

Participants signed an institutional review board–approved informed consent. Eligibility was then verified through a structured interview. Each participant was trained on

how to satisfactory complete the n-back tasks through a series of steps. First they received written instructions and were asked to read along as a research associate read a description of each task aloud. Participants were asked to complete a number of training trials. Training progressed from the low, to the moderate, to the difficult task level so that participants could develop conceptual understanding of the relationship between tasks. Repetitions of the instructions and practice sets were presented at each task level until participants demonstrated a minimum proficiency of 7 correct responses on the 0 and 1-back (out of 10 and 9 expected responses respectively) and of at least 4 (out of 8) on the 2-back. A maximum of 9 practice sets were allowed for the 2-back. Participants who failed to meet the training requirements were excluded from the analysis.

After a short break, participants reported to the instrumented vehicle. A research associate was seated in the rear of the vehicle for the entire experiment to ensure safety, provide driving directions and answer questions as needed. Participants start driving for 10 minutes to reach the interstate I-93. Subsequently, participants drove for 20 minutes or more on the highway to familiarize themselves with the vehicle. After this period the first two minute *reference period* of single driving task started . Following a 30 second separation interval, 18 seconds of recorded instructions then indicate the upcoming *n-back* task. Table 5.4 shows an overview of the procedure followed by the participants.

Period	Duration (min:s)	Description
Adaptation	~ 30:00	Single task driving
Driving (REFERENCE)	2:00	Single task driving
	0:30	Separation interval
Driving + 1sttask	0:18	Task instructions
	2:00	Four 10-item blocks
	2:00	Recovery period
Driving + 2nd task	0:18	Task instructions
	2:00	Four 10-item blocks
	2:00	Recovery period
Driving + 3rd task	0:18	Task instructions
	2:00	Four 10-item blocks
	0:30	Separation interval
Driving (recovery)	2:00	Single task driving

Table 5.4: Procedure overview (adapted from Mehler et al. [59])

In the consideration of demand periods Mehler et al. [59] defined each period as 2 minutes of sustained demand. For the purposes of estimating driving workload, this thesis includes the task instructions as part of the demand period. This is an important assumption because the physiological arousal of the participant starts changing when the task instructions are played. Since this thesis applies a shape-bases similarity classification approach, the 18 seconds instructions periods modifies the shape of the

input signals. Chapter 4 illustrates how the classification accuracy improves when the instruction period is considered part of the task.

6

EXPERIMENTAL WORK AND RESULTS

This chapter evaluates the feasibility of using data from two physiological signals, named Skin Conductance Level (SCL) and Heart Rate (HR), to classify distinct cognitive workload levels. The performance of the Symbolic Aggregate Approximation (SAX) using the k-Nearest-Neighbor classifier is evaluated on empirical study data. In general, supervised learning algorithms build a learning model using labeled training data, i.e. each category being classified has a class label, and each example in the training data has one and only one class label. In this case, the k-Nearest-Neighbor will store the training data in a model repository. In the classification phase, (when unlabeled data are classified), for each new example of unknown class, the classifier counts the number of k nearest observations to each class in the training set. Finally, the new example gets assigned to the class having the highest number of counts.

As explained in Section 4.1.4, finding the best parameters and evaluating the classifier is not a trivial process. Overfitting, a situation in which the classifier learns the peculiarities of the given training data instead of finding the underlying patterns, is one of the main risks of an excessive parameter optimization. Overfitting is . To improve the generalizability of the results of this thesis, 10-fold cross validation as proposed by Salzberg [75] is conducted. In order to find the best parameters for the algorithm (e.g. number of neighbors k) and for the discretization approach (e.g. number of segments, alphabet size, window size), in each cross-validation fold, the best parameters were found using a second 10-fold cross-validation run.

Four groups of experiments were conducted to evaluate the classification power of SAX and k-Nearest-Neighbor to detect driver cognitive workload:

1. 2-Back vs. reference
2. 2-Back vs. recovery
3. 1-Back vs. reference
4. 1-Back vs. recovery

The best SAX discretization parameters were determined for each group of experiments. Experiments 1 to 4 use discrete segments where a constant demand is imposed on the subject, e.g. 2-back, 1-back, recovery and single-task driving (reference) periods. The aim of those four experiments is to investigate if the algorithm is able to categorize each demand period based on the shape structure of heart rate and SCL signals. Furthermore, a statistical test, as proposed in Witten et al. [87, p. 157-159], is applied to the classification accuracy results for each parameter combination (i.e. number of segments, sliding window size, alphabet size, number of neighbors). The feasibility of using TSBs for cognitive workload discrimination is also explored. Finally a summary of the results of each experimental setting is presented.

6.1 EXPERIMENTS

This section describes the data sets used for classification as well as the experimental work. Classification and evaluation experiments were conducted using MATLAB. The SAX discretization procedure was adapted from [50]. Figure 6.1 shows an overview of the protocol timeline and the segments extracted from each subject for each classification experiment. In all the experiments, the SAX words are built using discrete segment data from two minutes of driving. For the high demand periods (e.g. 1-Back and 2-Back), the arousal begins when the subject hears the instructions, therefore the 18 second instruction period was included as part of the n-back segments. In this case, using only SAX words to distinguish cognitive workload levels does not require a numerosity reduction technique, since the frequency of the words is not taken into account.

The purpose of the experiments is to assess whether the 2-back and 1-back periods can be separated from the reference and recovery periods. This is the first step towards developing a system capable of driver state detection in a real time environment.

Several authors have suggested that training a classifier on an individual basis yields higher classification performance [16, 86]. The experiment design of the study used in this thesis dictates that each subject is exposed to three n-back levels (0-back, 1-back, 2-back) only once. Thus, the classifier does not have enough labeled data for each subject in order to build an individual model. Therefore, the experiments conducted in this thesis use data from all 99 subjects to build the classification model. This thesis employed 10-fold cross-validation to estimate the out-of-sample error of each model. The folds were broken per subject, i.e. nine folds contained 10 subjects' data and one fold contained only 9 subjects' data. As usual in 10-fold cross-validation, in each iteration nine folds are used for training and one for testing. One of the most important assumptions for the cross-validation folds is that the folds are independent. Therefore, constructing the folds on a subject basis yields a more accurate out-of-sample error. An alternative procedure would be to randomly select 10% of the data for each fold regardless of the subject the data belong to. In the latter case, data from the same subject might be contained in both the training and the test set, breaking the independence assumption.

Finding an appropriate set of parameters for the SAX discretization procedure and the k-Nearest-Neighbor algorithm is a compromise between computational cost and classification accuracy. For instance, discretizing a time series to SAX words of length 4, 8, 16, 32, 64 requires 4, 8, 16, 32, 64 bytes, respectively (assuming they are encoded as ASCII characters). While increasing the length of the SAX words usually achieves higher accuracy, more space is required to store the discretized sequence. Some other parameters, like the number k of nearest neighbors, have an impact on the number of calculations needed determine the class that a sequence belongs to.

The classification engine used in this thesis has various parameters which affect the classification performance. These parameters were optimized through experimental work using cross-validation. The classification accuracy was optimized using the training set on each cross-validation iteration. The error rate is estimated using the best parameters (i.e., the best model) to classify the test set. The parameters optimized include the sliding window size, the length of the SAX words, the alphabet size and the number of neighbors. The next sections provide an empirical analysis on the

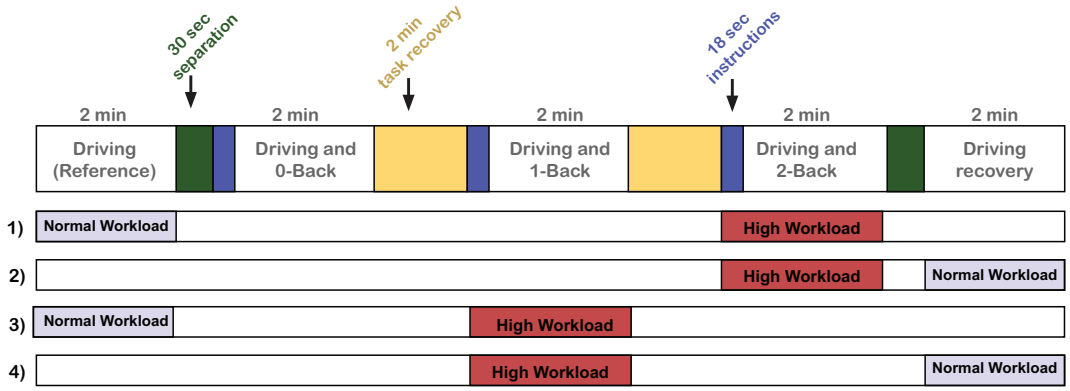

Figure 6.1: Overview of the datasets used in each experiment. The top row shows the protocol timeline with annotations for each period (see Table 5.4). The following four rows show the discrete segments and the corresponding labels used in each experiment. 1) Considers the 2-back and the reference period. 2) Uses the 2-back and the recovery period. 3) Considers the 1-back and the reference period. 4) Uses the 1-back and the recovery period. Note that the order for the n-back tasks was counterbalanced for each subject in the sample (adapted from Zec [95]).

parameters' impacts onclassification accuracy. The following experiments are only used to exemplify the impact of each of the parameters. To determine the best set of parameters, a separate 10-fold cross-validation run with a nested 10-fold cross-validation run in each iteration was used.

6.1.1 *Impact of window size and number of segments*

The sliding window length and the number of symbols in each window (i.e. the SAX word length), have a significant impact on the representation of the time series. These two parameters have also been shown to have an impact on classification accuracy. These two parameters are tested together, since they are correlated with each other. The window to symbol rate or compression rate R is calculated as $R = \frac{w}{n}$, where w is the length of the sliding window and n is the number of symbols in each window.

The compression rate was tested in the range of 2 to 480 (e.g. a compression rate of two means that two observations are compressed in one symbol) and the sliding window length was tested with values of 40, 6, 80, 120, 240 and 480. In these experiments the alphabet size was set to 4 and k was set to 3 neighbors. Figure 6.2 depicts the effects of varying the sliding window size and the number of symbols per window. It can be appreciated that a compression rate equal to the window length value (i.e. out of a window of length w only one symbol is generated) decreases the accuracy to 50%, meaning that the class selection is equivalent to a random guess (see yellow bars in Figure 6.2).

An explanation for this phenomenon is that both the skin conductance level and the heart rate measures change constantly. Therefore, using a high compression rate diminishes the changes of the time series data, which makes it harder for the classifier to distinguish differences between the two cognitive states.

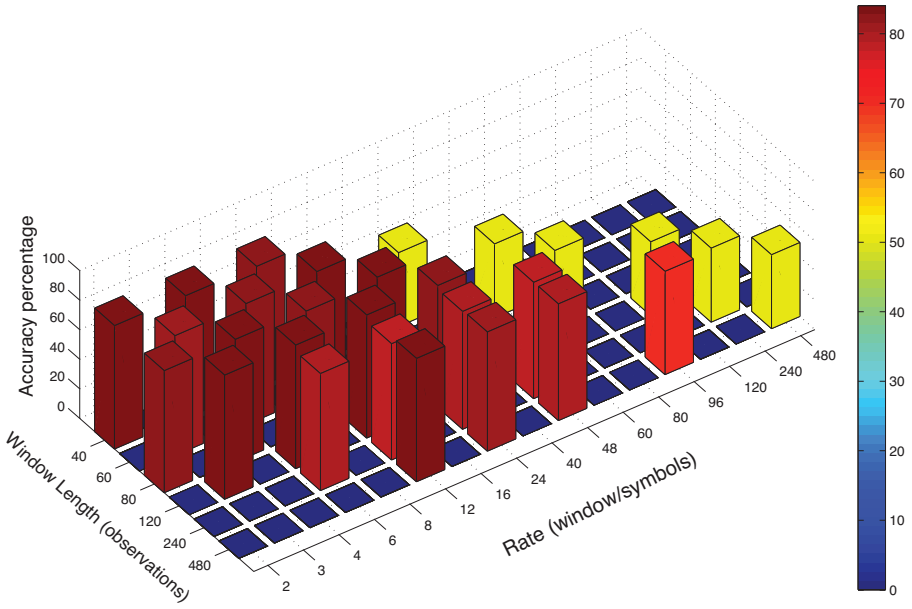

Figure 6.2: Effect of sliding window length and number of segments per window on the accuracy of the k-Nearest-Neighbor classifier. In these experiments the alphabet size was set to 4 and k was set to 3.

The accuracy values range from 76% to 82% when the compression rate is between 2 and 12. Thus, regardless of the window length, as long as the compression rate remains in this range, high accuracy values can be expected.

6.1.2 *Impact of the number of neighbors*

Choosing the optimal value of k depends upon the data at hand. Using a large value of k helps reduce the effects of noisy data in classification [87, p. 568], but it also causes the boundaries between the classes to be less defined, i.e. a high k could causethe classifier's performance to deteriorate. Figure 6.3 shows how mean accuracy is affected as k is varied.

In the first experiment (see blue line in Figure 6.3), the alphabet size was set to five and the window to symbols ratio to two. It can be seen that the accuracy of the classifier improves gradually when the number of neighbors increases from two to five. In the range of 7 to 23 neighbors, the performance decays slightly. On the other hand, in the second experiment (see red line in Figure 6.3), the alphabet size was set to four and the window to symbols ratio to 30. In this case, the classification accuracy improves for up to three nearest neighbors but decreases significantly when the value of k is increased beyond that.

The results of the experiments lead to two conclusions. First, both suggest that an extremely high value of k can reduce the classification accuracy. Second, they provide evidence that changing one or more input parameters alters the behavior of the classifier. Thus, it is important to assess the input parameters as a set, rather than in separate experiments.

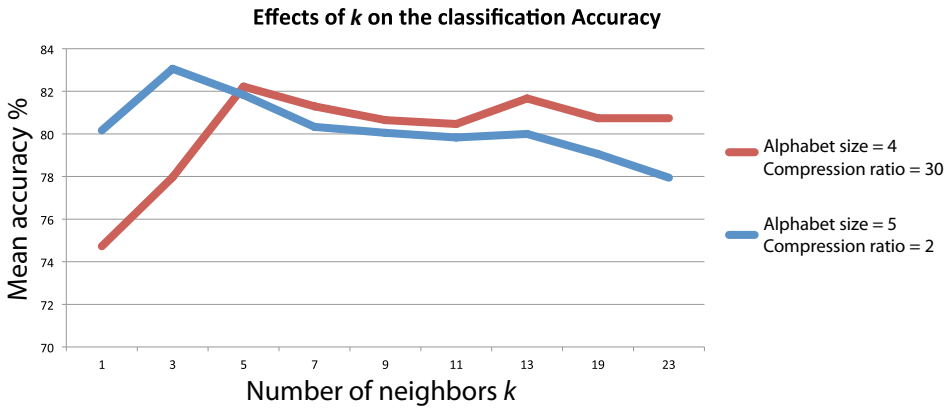

Figure 6.3: Effect of the number of neighbors on the accuracy of the k-Nearest-Neighbor classifier. In these experiments the alphabet size and compression ratio were set to fourwith sliding windows of size 40.

6.1.3 *Feasibility of time series bitmaps (TSBs)*

As previously discussed, time series bitmaps have been proposed as a viable approach to classifying insect behaviors [36]. This thesis proposes an experiment to establish the feasibility of using TSBs for cognitive workload classification. A TSB is a summary representation of a discretized time series using SAX. The main idea behind TSBs is to count the frequency of subsequences of length n and build a grid of length n^2, where the cells represent all possible permutations of n symbols using an alphabet of size a (see 4.1.3).

Kasetty et al. [36] propose an algorithm to maintain a TSB in constant time, which can be used to classify distinct workload levels in real time. As an initial step, one has to provide a template for each activity in question. In this case, a template of the "high" demand level could be the average of the TSBs across all the subjects in the sample for the 2-back period, i.e. the average value of each cell for all TSBs (one TSB per subject's class label). Moreover, using a template to classify the TSBs reduces the time complexity of the k-Nearest-Neighbor to constant time, since only t distances are calculated each time, where t is the number of class templates. Since the aforementioned method relies on how accurately a template represents a given class, this experiment aimed to show that across subjects, different cognitive workload levels will produce different TSBs.

Similar to the experiments for SAX, this Section proposes to use discrete data segments (e.g. 1-back, 2-back, reference segments) to build a TSB for each subject in each segment. Once all the TSBs are available, the k-Nearest-Neighbor classifier is used to group the TSBs according to their Euclidean distance to each other. Two different settings are used in this experiment, corresponding to experiments 1 and 3 from the SAX experiment group:

1. 2-Back vs. reference
2. 2-Back vs. recovery
3. 1-Back vs. reference
4. 1-Back vs. recovery

Building the best template involves adjusting the parameters of the underlying SAX words, and using cross-validation to see which parameter set performs best. An approach similar to TSBs is proposed by Lin and Li [49], where the authors advise that: "time series with smooth patterns can be described with a small w, and those with rapidly changing patterns prefer large w to capture the critical changes". This work tries to show the feasibility of using TSBs, therefore parameters are optimized towards better k-Nearest-Neighbor classification.

6.2 RESULTS

The following sections present a summary of the classification results and provide key observations for each approach. Moreover, detailed information for each experiment and each approach is provided: labeled categories considered, window length, total number of SAX words, and overall classification accuracy. For each class, true positive rate (TP rate), false positive rate (FP rate), precision and F-scores are provided.

Classification approaches using machine learning are not perfect. A compromise must be struck between the most important performance metrics. For instance, in the experiment results shown below, true positive rate of the "high" workload class indicates the completeness for that class (i.e. the rate of instances that were labeled as "high" workload and identified as such by the classifier). Precision of the "high" workload is a measure of exactness (i.e. the proportion of instances classified as "high" workload, that correspond to a "high" workload level). The F-score measures the compromise between exactness and completeness. The overall accuracy identifies the proportion of correctly classified instances. As previously discussed (see Section 4.1.4.1), the importance of each metric depends on the application.

6.2.1 *Symbolic Aggregation Approximation (SAX)*

The results presented in this section were obtained by using 10-fold cross-validation. In each iteration an inner loop was used to compare different parameters through another 10-fold cross-validation run. The adjusted parameters include the alphabet size, the sliding window length, the window to symbol rate and the number of neighbors. As already discussed, parameter optimization might lead to overfitting the model to the training data. The parameter set selected for the experiments was obtained as follows:

1. Select different parameters to compare (e.g. alphabet size range from 3 to 8, number of neighbors range from 1 to 15, etc.)
2. Run 10-fold cross-validation for each experiment with an inner loop for model comparison
3. Use the *t-test* to compare each model from each experiment to find the best parameters
4. Find a parameter set that maximizes the accuracy percentage across experiments

The parameter set used in the experiments is not an optimal set for each experiment, but a set of parameters that would maximize the accuracy across all experiments. Table 6.1 shows the results for each experiment using the following parameters: sliding

Dataset	Accuracy	Elevated Workload			
		TP rate	FP rate	Precision	F-Measure
2-Back vs. reference	85%	83%	15%	83%	1.53
2-Back vs. recovery	84%	88%	22%	80%	1.57
1-Back vs. reference	78%	82%	25%	76%	1.46
1-Back vs. recovery	78%	82%	26%	75%	1.45

Table 6.1: Results summary with SAX symbolic representation of time series using k-Nearest-Neighbor classifier. The optimal parameter set of window length $w = 12$ seconds, compression rate $r = 4$, alphabet size $a = 4$ and number of neighbors $n = 3$ is presented.

window size $w = 120$ (equivalent to 12 seconds of data), compression rate $r = 4$, alphabet size $a = 4$ and number of neighbors $n = 3$.

Results show that the highest workload level (i.e. 2-back) can be distinguished from both the reference period (i.e. only driving) and the recovery period (i.e. only driving after task completion) significantly better than random guessing (85% and 84% accuracy, respectively). The true positive rates are 83% and 85% for the 2-back vs. reference period and 2-back vs. recovery experiments, respectively. False positive rate values are in the range of 15% and 22% and the precision value is at least 80%.

The classifier's performance decreases when discriminating the middle workload level (i.e. 1-back) from the reference and the recovery periods, yielding a 78% accuracy. True positive rate in both cases is 82% and the false positive rate is 25% for the 1-back vs. reference experiment and 26% for the 1-back vs. recovery. Precision ranges from 75% to 76%.

6.2.2 *Time series bitmaps (TSBs)*

Time series bitmaps can be used to assess the cognitive workload level of a driver in real time. The experiments conducted in this thesis aim to establish the feasibility of distinguishing different cognitive workload levels by comparing the Euclidean distances of the TSBs.

The parameters used to generate and classify the TSBs were selected based on a series of observations. Kumar et al. [43] notes that the sliding window length N should be chosen such that it reflects the natural scale at which events occur in the time series. The number of segments n depends on the complexity of the signal. One would like to achieve a good compromise between fidelity of the approximation and dimensionality reduction. The alphabet size in this case is 4 since the TSBs are defined for DNA sequences which are composed of exactly four symbols.

Note that the numerosity reduction level to generate the SAX strings was set to 1 (i.e. remove the SAX string if the value is the same as the last one). This helps to

Dataset	Accuracy	Elevated Workload			
		TP rate	FP rate	Precision	F-Measure
2-Back vs. reference	80%	80%	21%	78%	1.43
2-Back vs. recovery	80%	78%	21%	77%	1.40
1-Back vs. reference	74%	74%	22%	74%	1.32
1-Back vs. recovery	72%	72%	28%	71%	1.30

Table 6.2: Results summary using TSBs and k-Nearest-Neighbor classifier using following parameter set: window length $w = 64$, compression rate $r = 16$, and number of neighbors $n = 3$

avoid over-counting subsequences from two consecutive sliding windows that use nearly identical data.

Generating TSBs requires normalization of the frequency counts. In these experiments, the same normalization parameters used to normalize the training set were used for the test set, as noted in Witten et al. [87, p. 575]. Similar to the experiments conducted using SAX words, the final parameter set is common for all experiments and was obtained using the same methodology (see 6.2.1).

Results in Table 6.2 show that the performance of TSBs is lower compared to SAX but still significantly better than chance. The accuracy, true positive rate, false positive rate and precision when distinguishing elevated workload (i.e. 2-back) vs. baseline period is 80%, 80%, 21% and 78%, respectively. Similar to SAX results, performance decreases if the algorithm attmeps to distinguish the 1-back task from the reference or the recovery period. In these cases, the accuracy and the true positive rate range from 72% to 74%, the false positive rate from 22% to 28% and the precision from 71% to 74%. Nevertheless, better performance might be achieved if the training and the test data were drawn from the same individual.

7

DISCUSSION

Cognitive workload state detection is challenging. Unlike visual or manual workload, cognitive workload can not be observed from external behavior. The ability to constantly monitor the driver's cognitive state could enhance current safety systems to provide feedback to the driver in cases where driving performance may be affected by high workload levels. Moreover, a cognitive workload manager could be developed to optimally manage and adapt the demands that the In-Vehicle Information Systems (IVIS) impose on the driver, thus preventing accidents.

This thesis presented a thorough literature review on cognitive workload, starting with attentional capacity models and their attendant concepts of task demand and capacity. Several definitions of workload were provided, leading to a definition of cognitive workload. Furthermore, three main empirical methods to measure cognitive workload were analyzed: performance measures, subjective measures and physiological measures.

A classification engine was proposed and evaluated using data collected from an on-road study involving the AwareCar, an instrumented vehicle able to monitor physiological signals from a driver. A total of 99 subjects were included in the analysis. The classification engine used a discretization technique called Symbolic Aggregate Approximation (SAX) of sequences of physiological measurements: Heart Rate and Skin Conductance Level. Using a k-Nearest-Neighbor classifier, a series of experiments were conducted to evaluate the feasibility of using SAX for cognitive workload state estimation. SAX is only defined to deal with univariate time series; this work used multivariate time series (HR, SCL), and the distance measure of the time series was therefore calculated as the mean distance of the SAX words from each one of the time series. One can find more sophisticated approaches to deal with multivariate time series, but with this simplistic approach one is able to determine the feasibility of applying SAX with multivariate time series.

Being able to rapidly classify the cognitive workload level of a driver in real time is a desirable feature of the classification engine. This work also proposed the use of Time Series Bitmaps (TSBs), and an evaluation experiment classifying the discrete segments of elevated and normal cognitive workload was also conducted.

7.1 CONCLUSIONS

The results show that classifying cognitive workload using the structural shape of the heart rate and skin conductance data is a suitable approach. Applying cross-validation with a nested loop for parameter selection enhances the out-of-sample error estimation [75]. Thus it is safe to say that the k-Nearest-Neighbor classifier with the given parameters performed substantially better than random guessing.

In the experimental work conducted using SAX, the results show that using discrete segments of data (2-back, 1-back, recovery, reference), the classifier can successfully distinguish elevated levels of cognitive workload from the recovery and reference periods. The 2-back task can be distinguished from the reference period with 85% accuracy

and from the recovery period with 84% accuracy. The 1-back task imposes less mental workload on the subjects. Therefore, discriminating it from the reference and the recovery periods yields a lower accuracy percentage. In both cases—the 1-back vs. recovery and the 1-back vs. baseline experiments—the classification accuracy is 78%.

Using TSBs instead of SAX words to summarize the time series data resulted in an average 6% decrease in classification accuracy. The classifier's performance may be increased by building TSB templates that take into account individual differences among participants. The main advantage of using TSBs is their ability to classify incoming physiological measurements in real time.

Zec [95], Tan [80] performed an analysis of the same study data used in this thesis using different approaches. While this work focuses on physiological measures, both authors considered visual and vehicle metrics for their analyses. On the one hand, Zec [95] used feature based classification methods including k-Nearest-Neighbor, logistic regression, Naive-Bayes and Neural Networks. The author reported that the 1-Nearest-Neighbor classifier using a 30 second sliding window yielded 91.7%, 87.6%, 87.8% and 86.1% accuracy when classifying the reference vs. 2-back, recovery vs. 2-back, reference vs. 1-back and recovery vs. 1-back, respectively. While the results indicate a higher accuracy level, one has to consider that the error is estimated with a separate cross-validation run for each classifier and each sliding window length. Thus, it is possible that the error estimation is overly optimistic.

On the other hand, Tan [80] used Support Vector Machines with a Radial Basis Function (RBF) kernel to estimate the cognitive workload level in drivers. The dataset used in Tan's study is the same as the one in this thesis, but the author evaluated the performance of the classifier using the *d' value,* which is a parameter derived from a Receiver Operating Characteristic (ROC) graph analysis. Therefore, a direct comparison of results is difficult.

While the results of this work provide evidence that SAX and TSBs can be used successfully to identify the physiological responses caused by secondary tasks imposed on an individual while driving, some consideration of limitations is in order. On the one hand, a drawback of SAX is that discretizing a time series requires three input parameters to be set. Keogh et al. [40] notes that working with parameter-laden algorithms can prevent pattern discovery if the parameters were improperly chosen, or may result in the discovery of spurious (invalid) patterns. On the other hand, a crucial benefit of this approach is that training the classifier doesn't involve calculating the physiological baseline of an individual.

In conclusion, the experimental results from this thesis provide evidence that classification and detection of cognitive workload level in drivers is a challenging but achievable task. Through the use of a time series discretization approach and a machine learning classifier, different workload levels in drivers can be detected. Moreover, a second approach which enables real-time state detection was also introduced and evaluated. Future smart vehicles equipped with intelligent systems might benefit from a cognitive workload manager that enhances the driving experience, and more importantly, enhances the driver's safety by managing and adapting the IVIS interaction with the driver. A major contribution of this work is that it establishes that physiological reactions to elevated workload demands share structural similarities. Machine learning approaches that consider the shape of the incoming time series can be used to identify changes in the cognitive workload level of a driver.

7.2 FUTURE WORK

This work establishes the feasibility of using a discretization approach to analyze multivariate time series data applied to cognitive workload state detection. Furthermore, experiments to establish the viability of real-time state detection were also conducted. Future research on this area includes:

- Validation of the proposed approach with other data sets, including different driving conditions and physiological measurements form individuals with compromised cardiac health. Furthermore, a dataset designed to train and classify on a per-subject basis would be interesting to analyze.

- Use of Support Vector Machines (SVMs) with a local alignment kernel, as proposed by Xing et al. [89], Saigo et al. [74]. SVMs are a more powerful and more complex machine learning approach than k-Nearest-Neighbor. Tan [80] used SVMs with a Radial Basis Function (RBF) kernel on the same dataset in this thesis. The reported performance is acceptable, but it is the belief of the author that the framework can be improved using other kernel functions.

- Use of unlabeled data to improve classification performance. Most of the experiments studying the use of physiological signals to infer the cognitive workload state of a driver possess a high volume of unlabeled data. Wei and Keogh [82] successfully applied an algorithm to label the data. Having a higher quantity of labeled data will improve the accuracy of the classification model.

- Application of a discretization technique such as SAX to time series data to integrate them with other measurements such as driving behavior metrics. Ensemble methods might potentially assist the use of different classifiers to deal with time series and non-time series data simultaneously.

BIBLIOGRAPHY

[1] R. W. Backs and W. Boucsein. *Engineering psychophysiology: issues and applications*. Lawrence Erlbaum Assoc Inc, 2011. ISBN 0805824537. (Cited on pages xv and 16.)

[2] W. Boucsein. Applications of Electrodermal Recording. In *Electrodermal Activity*, number 1973, pages 259–523. Springer US, Boston, MA, 2012. ISBN 978-1-4614-1125-3. doi: 10.1007/978-1-4614-1126-0. URL http://www.springerlink.com/index/10.1007/978-1-4614-1126-0. (Cited on pages 27, 28, and 29.)

[3] W. Boucsein. Principles of Electrodermal Phenomena. In *Electrodermal Activity*, pages 1–86. Springer US, Boston, MA, 2012. ISBN 978-1-4614-1125-3. doi: 10.1007/978-1-4614-1126-0. URL http://www.springerlink.com/index/10.1007/978-1-4614-1126-0. (Cited on pages 27 and 29.)

[4] K. a. Brookhuis and D. de Waard. The use of psychophysiology to assess driver status. *Ergonomics*, 36(9):1099–110, Sept. 1993. ISSN 0014-0139. doi: 10.1080/00140139308967981. URL http://www.ncbi.nlm.nih.gov/pubmed/8404838. (Cited on page 17.)

[5] K. A. Brookhuis and D. de Waard. Assessment of Drivers' Workload: Performance and Subjective and Physiological Indexes. *Stress, workload, and fatigue*, 2000. (Cited on pages 17 and 29.)

[6] K. A. Brookhuis, G. de Vries, and D. de Waard. The effects of mobile telephoning on driving performance. *Accident Analysis & Prevention*, 23(4): 309–316, 1991. URL http://www.sciencedirect.com/science/article/pii/000145759190008S. (Cited on page 14.)

[7] I. D. Brown. Measuring the 'spare mental capacity' of car drivers by a subsidiary auditory task. *Ergonomics*, (November 2012):37–41, 1962. URL http://www.tandfonline.com/doi/abs/10.1080/00140136208930580. (Cited on pages 29 and 55.)

[8] I. D. Brown and E. C. Poulton. Measuring the spare 'mental capacity'of car drivers by a subsidiary task. *Ergonomics*, (November 2012):37–41, 1961. URL http://www.tandfonline.com/doi/abs/10.1080/00140136108930505. (Cited on pages 17 and 55.)

[9] B. Cain. A Review of the Mental Workload Literature. *Defence research and development Toronto (Canada)*, pages 4–1–4–34, 2007. URL http://www.dtic.mil/cgi-bin/GetTRDoc?Location=U2&doc=GetTRDoc.pdf&AD=ADA474193. (Cited on pages 8, 10, 15, 17, and 18.)

[10] K.-p. Chan and A. W.-c. Fu. Efficient Time Series Matching by Wavelets. *Proceedings 15th International Conference on Data Engineering (Cat. No.99CB36337)*, pages 126–133, 1999. doi: 10.1109/ICDE.1999.754915. URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=754915. (Cited on page 38.)

[11] M. J. Christie. Electrodermal activity in the 1980s: a review. *Journal of the Royal Society of Medicine*, 74(8):616–22, Aug. 1981. ISSN 0141-0768. URL http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=1438893&tool=pmcentrez&rendertype=abstract. (Cited on page 17.)

[12] G. D. Clifford, F. Azuaje, and P. E. McSharry. ECG Statistics , Noise , Artifacts , and Missing Data. In *Advanced methods and tools for ECG data analysis*, chapter 3, pages 55–99. Artech House, 2006. ISBN 1580539661. (Cited on pages xiii, xv, 30, and 31.)

[13] J. F. Coughlin, B. Reimer, and B. Mehler. Driver wellness, safety & the development of an awarecar. *AgeLab, Mass Inst. Technol . . .*, pages 1–15, 2009. URL https://www.foundationcenter.org/grantmaker/santos/agelab.pdf. (Cited on pages 1 and 20.)

[14] J. F. Coughlin, B. Reimer, and B. Mehler. Monitoring, managing, and motivating driver safety and well-being. *IEEE Pervasive Computing*, 10(3):14–21, 2011. ISSN 1536-1268. doi: 10.1109/MPRV.2011.54. URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5958684. (Cited on pages xiii, 1, 2, and 21.)

[15] M. E. Dawson, A. M. Schell, and D. L. Filion. The electrodermal system. In J. T. Cacioppo, L. G. Tassinary, and G. G. Berntson, editors, *Handbook of Psychophysiology*, pages 200–223. Cambridge University Press, 2nd. edition, 2000. ISBN 62634X. (Cited on pages xiii, 27, and 28.)

[16] A. de Santos Sierra, C. Avila Sanchez, J. Guerra Casanova, and G. Bailador del Pozo. Real-Time Stress Detection by Means of Physiological Signals. In Y. D. Jucheng, editor, *Recent Application in Biometrics*, chapter 2. InTech, 2011. ISBN 978-953-307-488-7. URL http://www.intechopen.com/books/recentapplication-in-biometrics/hand-biometrics-in-mobile-devices. (Cited on pages 22, 26, and 62.)

[17] A. de Santos Sierra, C. Sanchez Avila, J. Guerra Casanova, and G. Bailador del Pozo. A Stress-Detection System Based on Physiological Signals and Fuzzy Logic. *IEEE Transactions on Industrial Electronics*, 58(10):4857–4865, 2011. URL http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5680639. (Cited on pages 21, 26, and 56.)

[18] D. de Waard. *The measurement of drivers' mental workload.* PhD thesis, University of Groningen, 1996. URL http://people.usd.edu/~schieber/pdf/deWaard-Thesis.pdf. (Cited on pages 16, 17, and 18.)

[19] M. Deshpande and G. Karypis. Evaluation of techniques for classifying biological sequences. *Advances in Knowledge Discovery and Data Mining*, pages 1–14, 2002. URL http://www.springerlink.com/index/P7AT0774RWA2KRK0.pdf. (Cited on pages 33 and 35.)

[20] J. a. Deutsch and D. Deutsch. Attention: Some Theoretical Considerations. *Psychological Review*, 70(1):80–90, 1963. ISSN 0033-295X. doi: 10.1037/h0039515. URL http://content.apa.org/journals/rev/70/1/80. (Cited on page 6.)

[21] J. Engström, E. Johansson, and J. Östlund. Effects of visual and cognitive load in real and simulated motorway driving. *Transportation Research Part F: Traffic Psychology and Behaviour*, 8(2):97–120, Mar. 2005. ISSN 13698478. doi: 10.1016/j.trf.2005.04.012. URL http://linkinghub.elsevier.com/retrieve/pii/S1369847805000185. (Cited on pages 14 and 23.)

[22] S. H. Fairclough and L. Venables. Prediction of subjective states from psychophysiology: a multivariate approach. *Biological psychology*, 71(1):100–10, Jan. 2006. ISSN 0301-0511. doi: 10.1016/j.biopsycho.2005.03.007. URL http://www.ncbi.nlm.nih.gov/pubmed/15978715. (Cited on page 29.)

[23] C. Faloutsos, M. Ranganathan, and Y. Manolopoulos. Fast subsequence matching in time-series databases. *Proceedings of the 1994 ACM SIGMOD international conference on Management of data - SIGMOD '94*, pages 419–429, 1994. doi: 10.1145/191839.191925. URL http://portal.acm.org/citation.cfm?doid=191839.191925. (Cited on page 38.)

[24] S. Fishel, E. Muth, and A. Hoover. Establishing appropriate physiological baseline procedures for real-time physiological measurement. *Journal of Cognitive Engineering and Decision Making*, 1(3):286–308, 2007. doi: 10.1518/155534307X255636. URL http://edm.sagepub.com/content/1/3/286.short. (Cited on page 41.)

[25] M. Folstein, S. Folstein, and P. McHugh. *Mini-Mental State: a practical method for grading the cognitive state of patients for the clinician*, volume 12. 1975. URL http://www.health.fgov.be/internet2Prd/groups/public/@public/@dg1/@acutecare/documents/ie2divers/19074388_nl.pdf. (Cited on page 53.)

[26] a. Fong, C. Sibley, a. Cole, C. Baldwin, and J. Coyne. A Comparison of Artificial Neural Networks, Logistic Regressions, and Classification Trees for Modeling Mental Workload in Real-Time. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, 54(19):1709–1712, Sept. 2010. ISSN 1071-1813. doi: 10.1177/154193121005401973. URL http://pro.sagepub.com/lookup/doi/10.1177/154193121005401973. (Cited on pages 22 and 26.)

[27] D. Gopher and E. Donchin. Workload: An examination of the concept. In K. R. Boof, L. Kaufman, and J. P. Thomas, editors, *Handbook of Perception and Human Performance*, volume 2, pages 41–1—-41–49. 1986. URL http://psycnet.apa.org/psycinfo/1986-98619-019. (Cited on pages 1, 6, and 15.)

[28] K. S. Gould, B. K. Rø ed, E.-R. Saus, V. F. Koefoed, R. S. Bridger, and B. E. Moen. Effects of navigation method on workload and performance in simulated high-speed ship navigation. *Applied ergonomics*, 40(1):103–14, Jan. 2009. ISSN 1872-9126. doi: 10.1016/j.apergo.2008.01.001. URL http://www.ncbi.nlm.nih.gov/pubmed/18295184. (Cited on page 29.)

[29] J. Han, M. Kamber, and J. Pei. *Data Mining: Concepts and Techniques: Concepts and Techniques*. Elsevier, 3 edition, 2011. (Cited on pages 33, 45, 47, and 48.)

[30] J. Healey, J. Seger, and R. W. Picard. Quantifying driver stress: developing a system for collecting and processing bio-metric signals in natural situations. *Biomedical sciences instrumentation*, 35(483):193–8, Jan. 1999. ISSN 0067-8856. URL http://www.ncbi.nlm.nih.gov/pubmed/11143346. (Cited on pages 17, 29, and 56.)

[31] J. A. Healey and R. W. Picard. Detecting Stress During Real-World Driving Tasks Using Physiological Sensors. *IEEE Transactions on Intelligent Transportation Systems*, 6(2):156–166, June 2005. ISSN 1524-9050. doi: 10.1109/TITS.2005.848368. URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1438384. (Cited on pages 21, 26, 29, and 56.)

[32] D. Heger and F. Putze. Towards Continuous Monitoring of Mental Workload. *Advances in Artificial Intelligence*, 2010. URL http://www.springerlink.com/index/94M536312355KR4K.pdf. (Cited on pages 22 and 26.)

[33] E. Jakobsson, A. Beutner, S. Pettersson, A. Bartels, F. Seglo, A. Lindqvist, and A. Nilsson. Deliverable D12.1 Architecture. Technical report, 2009. (Cited on page 24.)

[34] H. Jex. Measuring mental workload: Problems, progress, and promises. *Advances in Psychology*, 1988. URL http://www.sciencedirect.com/science/article/pii/S016641150862381X. (Cited on page 15.)

[35] D. Kahneman. *Attention and Effort*, volume 88. June 1975. ISBN 0130505188. doi: 10.2307/1421603. URL http://www.jstor.org/stable/1421603?origin=crossref. (Cited on page 6.)

[36] S. Kasetty, C. Stafford, G. P. Walker, X. Wang, and E. Keogh. Real-Time Classification of Streaming Sensor Data. *2008 20th IEEE International Conference on Tools with Artificial Intelligence*, pages 149–156, Nov. 2008. doi: 10.1109/ICTAI.2008.143. URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4669683. (Cited on pages xiv, xv, 42, 43, 44, and 65.)

[37] E. Kawakita and M. Itoh. Estimation of driver's mental workload using visual information and heart rate variability. *Transportation Systems (ITSC),*, pages 765–769, 2010. URL http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5625079. (Cited on pages 22, 26, and 56.)

[38] E. Keogh and S. Kasetty. On the need for time series data mining benchmarks: a survey and empirical demonstration. *Data Mining and Knowledge Discovery*, 2003. URL http://www.springerlink.com/index/G7535342U0781722.pdf. (Cited on page 36.)

[39] E. Keogh, K. Chakrabarti, S. Mehrotra, and M. J. Pazzani. Locally adaptive dimensionality reduction for indexing large time series databases. *Proceedings of the 2001 ACM SIGMOD international conference on Management of data - SIGMOD '01*, pages 151–162, 2001. doi: 10.1145/375663.375680. URL http://portal.acm.org/citation.cfm?doid=375663.375680. (Cited on page 38.)

[40] E. Keogh, S. Lonardi, and C. A. Ratanamahatana. Towards parameter-free data mining. *Proceedings of the 2004 ACM SIGKDD international conference*

on Knowledge discovery and data mining - KDD '04, page 206, 2004. doi: 10.1145/1014052.1014077. URL http://portal.acm.org/citation.cfm?doid=1014052.1014077. (Cited on page 70.)

[41] E. Keogh, J. Lin, and A. Fu. HOT SAX : Finding the Most Unusual Time Series Subsequence : Algorithms and Applications. *IEEE International Conference on Data Mining - ICDM*, pages 226—-233, 2005. (Cited on page 38.)

[42] B.-U. Köhler, C. Hennig, and R. Orglmeister. The principles of software QRS detection. *IEEE engineering in medicine and biology magazine : the quarterly magazine of the Engineering in Medicine & Biology Society*, 21(1):42–57, 2002. ISSN 0739-5175. URL http://www.ncbi.nlm.nih.gov/pubmed/11935987. (Cited on page 30.)

[43] N. Kumar, N. Lolla, and E. Keogh. Time-series bitmaps: a practical visualization tool for working with large time series databases. *SIAM 2005 Data Mining ...*, 2005. URL http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.75.8858. (Cited on pages xiv, 42, 43, and 67.)

[44] M. H. Kutila, M. Jokela, T. Mäkinen, J. Viitanen, G. Markkula, and T. W. Victor. Driver cognitive distraction detection: Feature estimation and implementation. *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, 221(9):1027–1040, Jan. 2007. ISSN 0954-4070. doi: 10.1243/09544070JAUTO332. URL http://journals.pepublishing.com/openurl.asp?genre=article&id=doi:10.1243/09544070JAUTO332. (Cited on page 24.)

[45] J. O. Laguna, A. Olaya, and D. Borrajo. A dynamic sliding window approach for activity recognition. *Proc. of the 19th intl. conf. on User modeling, adaption, and personalization*, pages 219–230, 2011. URL http://www.springerlink.com/index/V1747336094Q08X0.pdf. (Cited on page 35.)

[46] R. W. Levenson. Autonomic nervous system differences among emotions. *Psychological science*, 3(1):23–27, 1992. URL http://pss.sagepub.com/content/3/1/23.short. (Cited on page 29.)

[47] Y. Liang and J. Lee. Driver cognitive distraction detection using eye movements. *Passive Eye Monitoring*, 2008. URL http://www.springerlink.com/index/V55V06K21M7724V5.pdf. (Cited on page 1.)

[48] Y. Liang, J. D. Lee, and M. L. Reyes. Nonintrusive Detection of Driver Cognitive Distraction in Real Time Using Bayesian Networks. *Transportation Research Record: Journal of the Transportation Research Board*, 2018(1):1–8, 2007. doi: 10.3141/2018-01. URL http://trb.metapress.com/content/5211G37261H0551Q. (Cited on pages 23 and 26.)

[49] J. Lin and Y. Li. Finding structural similarity in time series data using Bag-of-Patterns representation. *Scientific and Statistical Database Management*, pages 461–477, 2009. URL http://www.springerlink.com/index/G670227067201725.pdf. (Cited on pages 41 and 66.)

[50] J. Lin, E. Keogh, S. Lonardi, and B. Chiu. A symbolic representation of time series, with implications for streaming algorithms. *Proceedings of the 8th ACM SIGMOD workshop on Research issues in data mining and knowledge discovery - DMKD '03*, page 2, 2003. doi: 10.1145/882085.882086. URL http://portal.acm.org/citation.cfm?doid=882082.882086. (Cited on pages xiv, xv, 37, 38, 39, 40, and 62.)

[51] J. Lin, E. Keogh, and W. Truppel. Clustering of streaming time series is meaningless. *Proceedings of the 8th ACM SIGMOD workshop on Research issues in data mining and knowledge discovery - DMKD '03*, page 56, 2003. doi: 10.1145/882095.882096. URL http://portal.acm.org/citation.cfm?doid=882082.882096. (Cited on page 35.)

[52] J. Lin, E. Keogh, L. Wei, and S. Lonardi. Experiencing SAX: a novel symbolic representation of time series. *Data Mining and Knowledge Discovery*, 15 (2):107–144, Apr. 2007. ISSN 1384-5810. doi: 10.1007/s10618-007-0064-z. URL http://www.springerlink.com/index/10.1007/s10618-007-0064-z. (Cited on pages xiii, 37, 38, and 40.)

[53] P. M. Linton, B. D. Plamondon, A. O. Dick, A. C. Bittner, and R. E. Christ. Operator workload for military system acquisition. *Applications of Human Performance Models to System Design*, 2:21–45, 1989. URL http://www.dtic.mil/cgi-bin/GetTRDoc?AD=ADA215322#page=29. (Cited on page 7.)

[54] R. J. Lysaght, S. G. Hill, A. Dick, B. D. Plamondon, P. M. Linton, W. W. Wierwille, A. L. Zaklad, A. C. Bittner, and R. J. Wherry. Operator workload: Comprehensive review and evaluation of operator workload methodologies. Technical report, 1989. URL http://oai.dtic.mil/oai/oai?verb=getRecord&metadataPrefix=html&identifier=ADA212879. (Cited on pages xiii, 11, and 12.)

[55] A. McGovern, D. H. Rosendahl, R. a. Brown, and K. K. Droegemeier. Identifying predictive multi-dimensional time series motifs: an application to severe weather prediction. *Data Mining and Knowledge Discovery*, 22(1-2):232–258, July 2010. ISSN 1384-5810. doi: 10.1007/s10618-010-0193-7. URL http://www.springerlink.com/index/10.1007/s10618-010-0193-7. (Cited on page 34.)

[56] B. Mehler, B. Reimer, J. F. Coughlin, and J. A. 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*, 2138(-1):6–12, Dec. 2009. ISSN 0361-1981. doi: 10.3141/2138-02. URL http://trb.metapress.com/openurl.asp?genre=article&id=doi:10.3141/2138-02. (Cited on pages 7, 17, 29, and 55.)

[57] B. Mehler, B. Reimer, and J. F. Coughlin. Physiological Reactivity to Graded Levels of Cognitive Workload across Three Age Groups: An On-Road Evaluation. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, 54(24): 2062–2066, Sept. 2010. ISSN 1071-1813. doi: 10.1177/154193121005402409. URL http://pro.sagepub.com/lookup/doi/10.1177/154193121005402409. (Cited on pages 7, 29, 53, 54, and 55.)

[58] B. Mehler, B. Reimer, and J. A. Dusek. MIT AgeLab delayed digit recall task (n-back). *MIT AgeLab White Paper Number 2011-3B*, 2011. URL http://agelab.mit.edu/system/files/Mehler_et_al_n-back-white-paper_2011_B.pdf. (Cited on pages 7, 56, and 57.)

[59] 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*, 54(3):396–412, Apr. 2012. ISSN 0018-7208. doi: 10.1177/0018720812442086. URL http://hfs.sagepub.com/cgi/doi/10.1177/0018720812442086. (Cited on pages xv, 7, 29, 56, and 58.)

[60] B. Mehler, B. Reimer, and M. Zec. Defining workload in the context of driver state detection and HMI evaluation. *Proceedings of the 4th International Conference on Automotive User Interfaces and Interactive Vehicular Applications - AutomotiveUI '12*, (c):187, 2012. doi: 10.1145/2390256.2390288. URL http://dl.acm.org/citation.cfm?doid=2390256.2390288. (Cited on pages 6, 7, 9, 10, and 17.)

[61] N. Merat, V. Anttila, and J. Luoma. Comparing the driving performance of average and older drivers: the effect of surrogate in-vehicle information systems. *... Research Part F: Traffic Psychology and ...*, 8:147–166, 2005. URL http://www.sciencedirect.com/science/article/pii/S1369847805000173. (Cited on page 10.)

[62] M. Miyaji, H. Kawanaka, and K. Oguri. Driver's cognitive distraction detection using physiological features by the adaboost. *2009 12th International IEEE Conference on Intelligent Transportation Systems*, pages 1–6, Oct. 2009. doi: 10.1109/ITSC.2009.5309881. URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5309881. (Cited on pages 22 and 26.)

[63] G. Morgan, M. Mikhail, M. Murray, and C. Larson. Patient Motions. In *Clinical anesthesiology*, chapter 6. (Cited on pages xiii and 29.)

[64] L. J. Mulder. Measurement and analysis methods of heart rate and respiration for use in applied environments. *Biological psychology*, 34(2-3):205–36, Nov. 1992. ISSN 0301-0511. URL http://www.ncbi.nlm.nih.gov/pubmed/1467394. (Cited on pages 17, 29, and 30.)

[65] R. D. O'donnell and F. T. Eggemeier. Workload assessment methodology. In K. R. Boof, L. Kaufman, and J. P. Thomas, editors, *Handbook of Perception and Human Performance*, volume 2, pages 42–1—-42–48. 1986. URL http://people.usd.edu/~schieber/pdf/ODonnell-Eggemeier.pdf. (Cited on pages xiii, xv, 1, 10, 11, 13, 15, 17, and 18.)

[66] J. Östlund, B. Peters, B. Thorslund, J. Engström, G. Markkula, A. Keinath, D. Horst, S. Juch, S. Mattes, and U. Foehl. Driving performance assessment - methods and metrics. Technical Report March 2004, Information Society Technologies (IST) Programme, Gothenburg, Sweden., 2005. URL http://www.citeulike.org/group/8271/article/3878230. (Cited on pages xv and 14.)

[67] J. Racine. Consistent cross-validatory model-selection for dependent data: hv-block cross-validation. *Journal of econometrics*, 99(1):39–61, Nov. 2000. ISSN 03044076. doi: 10.1016/S0304-4076(00)00030-0. URL http://www.sciencedirect.com/science/article/pii/S0304407600000300. (Cited on page 50.)

[68] N. Rauch, A. Kaussner, H.-P. Krüger, S. Boverie, and F. Flemisch. The Importance of Driver State Assessment Within Highly Automated Vehicles. *16th ITS World ...*, pages 1–8, 2009. URL http://haveit-eu.org/LH2Uploads/ItemsContent/25/3117-FULL-PAPER-THE-IMPORTANCE.pdf. (Cited on page 24.)

[69] M. a. Regan, C. Hallett, and C. P. Gordon. Driver distraction and driver inattention: definition, relationship and taxonomy. *Accident; analysis and prevention*, 43(5):1771–81, Sept. 2011. ISSN 1879-2057. doi: 10.1016/j.aap.2011.04.008. URL http://www.ncbi.nlm.nih.gov/pubmed/21658505. (Cited on page 7.)

[70] B. Reimer. Impact of Cognitive Task Complexity on Drivers' Visual Tunneling. *Transportation Research Record: Journal of the Transportation Research Board*, 2138(-1):13–19, Dec. 2009. ISSN 0361-1981. doi: 10.3141/2138-03. URL http://trb.metapress.com/openurl.asp?genre=article&id=doi:10.3141/2138-03. (Cited on pages 7, 14, 29, and 53.)

[71] 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. *Proceedings of the 1st International Conference on Automotive User Interfaces and Interactive Vehicular Applications - AutomotiveUI '09*, (AutomotiveUI):115–118, 2009. doi: 10.1145/1620509.1620531. URL http://portal.acm.org/citation.cfm?doid=1620509.1620531. (Cited on pages 7, 14, 18, and 28.)

[72] 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: The Journal of the Human Factors and Ergonomics Society*, 54(3):454–468, Feb. 2012. ISSN 0018-7208. doi: 10.1177/0018720812437274. URL http://hfs.sagepub.com/cgi/doi/10.1177/0018720812437274. (Cited on pages 7, 14, 17, 53, 54, and 56.)

[73] G. Rigas and Y. Goletsis. Real-Time Driver's Stress Event Detection. *Systems, IEEE Transactions*, 13(1):221–234, 2012. URL http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=6036175. (Cited on pages 21 and 26.)

[74] H. Saigo, J.-P. Vert, N. Ueda, and T. Akutsu. Protein homology detection using string alignment kernels. *Bioinformatics (Oxford, England)*, 20(11):1682–9, July 2004. ISSN 1367-4803. doi: 10.1093/bioinformatics/bth141. URL http://www.ncbi.nlm.nih.gov/pubmed/14988126. (Cited on page 71.)

[75] S. L. Salzberg. On comparing classifiers: Pitfalls to avoid and a recommended approach. *Data mining and knowledge discovery*, 328:317–328, 1997. URL http://www.springerlink.com/index/LJ163588K4QXH427.pdf. (Cited on pages 48, 49, 50, 51, 61, and 69.)

[76] J. Shieh and E. Keogh. iSAX : Indexing and Mining Terabyte Sized Time Series. *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 623–631, 2008. doi: 10.1145/1401890.1401966. URL http://doi.acm.org/10.1145/1401890.1401966. (Cited on page 38.)

[77] R. R. Singh, S. Conjeti, and R. Banerjee. An approach for real-time stress-trend detection using physiological signals in wearable computing systems for automotive drivers. *2011 14th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pages 1477–1482, Oct. 2011. doi: 10.1109/ITSC.2011.6082900. URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6082900. (Cited on pages 22 and 26.)

[78] M. R. Smith, G. J. Witt, and D. L. Bakowski. A Final Report of SAfety VEhicles using adaptive Interface Technology (Task 15): SAVE-IT Summary and Benefits Estimation. Technical report, Delphi Electronics & Safety, 2008. (Cited on page 24.)

[79] F.-T. Sun, C. Kuo, and H.-T. Cheng. Activity-Aware Mental Stress Detection Using Physiological Sensors. *Mobile Computing, ...*, 76:211–230, 2012. doi: 10.1007/978-3-642-29336-8_12. URL http://www.springerlink.com/index/U121065535M718J7.pdf. (Cited on page 56.)

[80] C. Tan. *A Multi-Parametric Framework for the Classification of Cognitive Workload Levels in Drivers*. PhD thesis, Technical University Munich, Munich, 2010. (Cited on pages 23, 70, and 71.)

[81] A. M. Treisman. Contextual cues in selective listening. *Quarterly Journal of Experimental Psychology*, (December 2012):37–41, 1960. URL http://www.tandfonline.com/doi/abs/10.1080/17470216008416732. (Cited on page 6.)

[82] L. Wei and E. Keogh. Semi-supervised time series classification. *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '06*, page 748, 2006. doi: 10.1145/1150402.1150498. URL http://portal.acm.org/citation.cfm?doid=1150402.1150498. (Cited on pages 33 and 71.)

[83] C. Wickens. Multiple resources and performance prediction. *Theoretical issues in ergonomics science*, 2002. URL http://www.tandfonline.com/doi/abs/10.1080/14639220210123806. (Cited on page 9.)

[84] C. D. Wickens. Multiple resources and mental workload. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 50(3):449–455, 2008. doi: 10.1518/001872008X288394. URL http://hfs.sagepub.com/content/50/3/449.short. (Cited on page 6.)

[85] G. F. Wilson and F. T. Eggemeier. Mental Workload Measurement. In *International Encyclopedia of Ergonomics and Human Factors, Second Edition - 3 Volume Set*. CRC Press, Mar. 2006. ISBN 978-0-415-30430-6. doi: doi:10.1201/9780849375477.ch167. URL http://dx.doi.org/10.1201/9780849375477.ch167. (Cited on pages 8, 11, and 12.)

[86] G. F. Wilson and C. A. Russell. Real-Time Assessment of Mental Workload Using Psychophysiological Measures and Artificial Neural Networks. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 45(4):635–643, 2003. ISSN 1547-8181. doi: 10.1518/hfes.45.4.635.27088. URL http://hfs.sagepub.com/cgi/doi/10.1518/hfes.45.4.635.27088. (Cited on page 62.)

[87] I. H. Witten, E. Frank, and M. A. Hall. *Data Mining: Practical machine learning tools and techniques*. Elsevier, 3 edition, 2011. ISBN 978-0-12-374856-0. (Cited on pages 47, 48, 49, 61, 64, and 68.)

[88] B. Xie and G. Salvendy. Review and reappraisal of modelling and predicting mental workload in single- and multi-task environments. *Work & Stress*, 14(1):74–99, Jan. 2000. ISSN 0267-8373. doi: 10.1080/026783700417249. URL http://www.tandfonline.com/doi/abs/10.1080/026783700417249. (Cited on pages xiii, 8, 9, 10, 11, and 12.)

[89] Z. Xing, J. Pei, and E. Keogh. A brief survey on sequence classification. *ACM SIGKDD Explorations Newsletter*, 12(1):40, Nov. 2010. ISSN 19310145. doi: 10.1145/1882471.1882478. URL http://portal.acm.org/citation.cfm?doid=1882471.1882478. (Cited on pages 33, 35, 36, 37, and 71.)

[90] L. Ye and E. Keogh. Time series shapelets: a new primitive for data mining. *KDD '09: Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2009. URL http://dl.acm.org/citation.cfm?id=1557019.1557122. (Cited on page 35.)

[91] R. M. Yerkes and J. D. Dodson. The relation of strength of stimulus to rapidity of habit - formation. *Journal of Comparative Neurology and Psychology*, 18 (5):459–482, 1908. URL http://onlinelibrary.wiley.com/doi/10.1002/cne.920180503/abstract. (Cited on page 1.)

[92] M. S. Young and N. a. Stanton. Malleable Attentional Resources Theory: A New Explanation for the Effects of Mental Underload on Performance. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 44(3):365–375, July 2002. ISSN 00187208. doi: 10.1518/0018720024497709. URL http://hfs.sagepub.com/cgi/doi/10.1518/0018720024497709. (Cited on page 6.)

[93] M. S. Young and N. A. Stanton. Mental Workload. In N. Stanton, H. Alan, K. Brookhuis, E. Salas, and H. Hendrick, editors, *Handbook of Human Factors and Ergonomics Methods*, number 2001, chapter Chapter 39, pages 39–1–39–9. 2005. ISBN 978-0-415-28700-5. doi: 10.1201/9780203489925.ch39. URL http://www.crcnetbase.com/doi/pdf/10.1201/9780203489925.ch39. (Cited on page 8.)

[94] M. S. Young and N. A. Stanton. Mental Workload: Theory, Measurement and Application. In *International Encyclopedia of Ergonomics and Human Factors, Second Edition - 3 Volume Set*. CRC Press, Mar. 2006. ISBN 978-0-415-30430-6. doi: doi:10.1201/9780849375477.ch168. URL http://dx.doi.org/10.1201/9780849375477.ch168. (Cited on page 17.)

[95] M. Zec. *A Machine Learning Approach towards Online Detection of Driver Mental Workload*. Master thesis, Technical University Munich, 2012. (Cited on pages xiv, 23, 63, and 70.)

[96] L. R. Zeitlin. Subsidiary task measures of driver mental workload. *Transportation Research Record: Journal of the Transportation Research Board*, 1403:23–27, 1993. (Cited on pages 17, 55, and 56.)

[97] L. R. Zeitlin. Estimates of driver mental workload: A long-term field trial of two subsidiary tasks. *Human Factors: The Journal of the Human Factors ...*, 37(3):611–621, 1995. URL http://hfs.sagepub.com/content/37/3/611.short. (Cited on pages 17 and 55.)

[98] Y. Zhang, Y. Owechko, and J. Zhang. Learning-based driver workload estimation. *Computational Intelligence in Automotive Applications*, 132:1–24, 2008. doi: 10.1007/978-3-540-79257-4_1. URL http://www.springerlink.com/index/41124802v8470043.pdf. (Cited on page 5.)