Learning-Based Driver Workload Estimation

Yilu Zhang[1], Yuri Owechko[2], and Jing Zhang[1]

[1] R&D Center, General Motors Cooperation, 30500 Mound Road, Warren, MI, USA, yilu.zhang@gm.com, jing.zhang@gm.com

[2] HRL Laboratories, LLC., 3011 Malibu Canyon Road, Malibu, CA, USA, yowechko@hrl.com

A popular definition of workload is given by O'Donnell and Eggmeir, which states that "The term workload refers to that portion of the operator's limited capacity actually required to perform a particular task" [1]. In the vehicle environment, the "particular task" refers to both the vehicle control, which is the primary task, and other secondary activities such as listening to the radio. Three major types of driver workload are usually studied, namely, visual, manual, and cognitive. Auditory workload is not treated as a major type of workload in the driving context because the auditory perception is not considered as a major requirement to perform a driving task. Even when there is an activity that involves audition, the driver is mostly affected cognitively.

Lately, the advanced computer and telecommunication technology is introducing many new in-vehicle information systems (IVISs), which give drivers more convenient and pleasant driving experiences. Active research is being conducted to provide IVISs with both high functionality and high usability. On the usability side, driver's workload is a heated topic advancing in at least two major directions. One is the offline assessment of the workload imposed by IVISs, which can be used to improve the design of IVISs. The other effort is the online workload estimation, based on which IVISs can provide appropriate service at appropriate time, which is usually termed as *Workload Management*. For example, the incoming phone call may be delayed if the driver is engaged in a demanding maneuver.

Among the three major types of driver workload, cognitive workload is the most difficult to measure. For example, withdrawing hands from the steering wheel to reach for a coffee cup requires extra manual workload. It also may require extra visual workload in that the position of the cup may need to be located. Both types of workload are directly measurable through such observations as hands-off-wheel and eyes-off-road time. On the other hand, engaging in thinking (the so-called minds-off-road phenomenon) is difficult to detect. Since the cognitive workload level is internal to the driver, it can only be inferred based on the information that is observable. In this chapter, we report some of our research results on driver's cognitive workload estimation.[1] After the discussion of the existing practices, we propose a new methodology to design driver workload estimation systems, that is, using machine-learning techniques to derive optimized models to index workload. The advantage of this methodology will be discussed, followed by the presentation of some experimental results. This chapter concludes with discussion of future work.

1 Background

Driver Workload Estimation (DWE) refers to the activities of monitoring the driver, the vehicle, and the driving environment in real-time, and acquiring the knowledge of driver's workload level continuously. A typical DWE system takes sensory information of the driver, the vehicle and the driving environment as inputs, and generates an index to the driver's workload level as shown in Fig. 1. The central issue of DWE is to design the driver workload estimation algorithm that generates the workload index with high accuracy.

[1] To simplify the terminology, we use "workload" interchangeably with "cognitive workload" in this chapter.

Y. Zhang et al.: *Learning-Based Driver Workload Estimation*, Studies in Computational Intelligence (SCI) **132**, 1–17 (2008)
www.springerlink.com

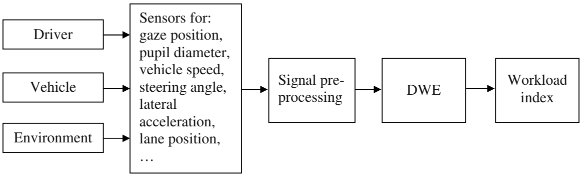

Driver
Vehicle
Environment
Sensors for: gaze position, pupil diameter, vehicle speed, steering angle, lateral acceleration, lane position, …
Signal pre-processing
DWE
Workload index

Fig. 1. The working process of a driver workload estimation system

A practical DWE system fulfills the following three requirements in order to identify driver's cognitive status while the driver is engaged in naturalistic driving practice.

- Continuously measurable: A DWE system has to be able to continuously measure workload while the driver is driving the vehicle so that workload management can be conducted appropriately to avoid overloading the driver.
- Residual capacity sensitive: The residual capacity of the driver refers to the amount of spare capacity of the driver while he/she is performing the primary task of maneuvering the vehicle, and, if applicable, engaging in secondary tasks such as drinking a cup of coffee or operating an IVIS. If the residual capacity is high, the driver is typically doing well in vehicle control and may be able to be engaging in even more secondary activities. Residual capacity is the primary interest for DWE.
- Highly non-intrusive: A DWE system should not interfere with the driver by any means.

Before the discussion of existing DWE methodologies in the next section, it is helpful to give a brief introduction of cognitive workload assessment methods. There exist four major categories of cognitive workload assessment methods in present-day practice [2], namely primary-task performance measures, secondary-task performance measures, subjective measures, and physiological measures.

The primary-task performance measures evaluate cognitive workload based on driving performance, such as lane-position deviation, lane exceedences, brake pressure, and vehicle headway. These measures are usually direct and continuous.

In the secondary-task approach, the driver is required to perform one or multiple secondary tasks, e.g., pushing a button when flashing LED light is detected in the peripheral vision. The secondary-task performance such as reaction time is measured as the index of driver's cognitive workload level. Secondary-task performance may introduce tasks unnecessary to regular driving and is intrusive to the driver.

With the subjective measure approach, driver's personal opinion on his/her operative experience is elicited. After a trip or an experiment, the subject is asked to describe or rate several dimensions of effort required to perform the driving task. If the driver and the experimenter establish clear mutual understanding of the rating scale, the subjective measure can be very reliable. Examples of popular subjective workload rating index are NASA Task Load Index (TLX) [3] and Subjective Workload Assessment Technique (SWAT) [4].

Physiological measures include brain activities such as event-related potential (ERP) and Electroencephalogram (EEG), cardiac activities such as heart rate variance, as well as ocular activities such as eye closure duration and pupil diameter changes. Physiological measures are continuous and residual-capacity sensitive. The challenge, however, lies in reliably acquiring and interpreting the physiological data sets, in addition to user acceptance issues.

Among the four workload assessment methods, primary-task performance measures and physiological measures (obtained by non-intrusive sensors) fulfill the above-discussed DWE requirements and are generally appropriate for DWE applications. Although not directly suitable for real-time DWE, the secondary-task performance measures and the subjective measures are still valuable in developing DWE since they provide ways to calibrate the index generated by a DWE system.

2 Existing Practice and Its Challenges

Most existing researches on DWE follow this pattern. First, analyze the correlation between various features, such as lane position deviation, and driver's workload. The ground truth of driver's workload is usually assessed by subjective measures, secondary-task performance, or the analysis of the task. The features are usually selected according to the prior understanding of human behaviors and then tested using well-designed experiments. While there are attempts reported to analyze the features simultaneously [5], usually the analysis is done on individual features [6, 7]. Second, models are designed to generate workload index by combining features that have high correlation with driver workload. We refer to the above methodology as *manual analysis and modeling*. The manual DWE design process is illustrated in Fig. 2. Research along this line has achieved encouraging success. The well-known existing models include the steering entropy [8] and the SEEV model [9]. However, there are yet difficulties in developing a robust cognitive workload estimator for practical applications, the reasons of which are discussed below.

First, the existing data analysis methods very much rely on the domain knowledge in the field of human behavior. Although many studies have been conducted and many theories have been proposed to explain the way that human beings manage resources and workload [2, 10–12], the relationship between overt human behavior and cognitive activities is by and large unclear to the scientific community. It is extremely difficult to design the workload estimation models based on this incomplete domain knowledge.

Second, manual data analysis and modeling are not efficient. Until now, a large number of features related to driver's cognitive workload have been studied. A short list of them includes: lane position deviation, the number of lane departure, lane departure duration, speed deviation, lateral deviation, steering hold, zero-crossing and steering reversal rate, brake pressure, the number of brake presses, and vehicle headway. With the fast advancing sensing technology, the list is quickly expanding. It has been realized that while each individual feature may not index workload well under various scenarios, the fusion of multiple features tends to provide better overall performance. However, in the course of modeling the data to estimate workload, the models tend to be either relatively simple, such as the linear regression models, or narrowly scoped by covering a small number of features. It is usually expensive and time-consuming to iteratively design models over a large number of features and validate models on a huge data set.

Third, most researchers choose workload inference features by analyzing the *correlation* between the observations of driver's behavior and driver's workload level. This analysis requires the assumption of uni-mode Gaussian distribution, which is very likely to be violated in reality. In addition, a feature showing low correlation with the workload levels is not necessarily a bad workload indicator.

For example, driver's eye fixation duration is one of the extensively studied features for workload estimation. However, studies show contradictory findings in the relation between workload level and fixation duration. Some of them show positive correlation [13, 14] while others show negative correlation [15, 16]. Does this mean fixation duration is not a good workload indicator? Not necessarily. The fact, that the average fixation duration may become either longer or shorter when driver's workload is high, implies that

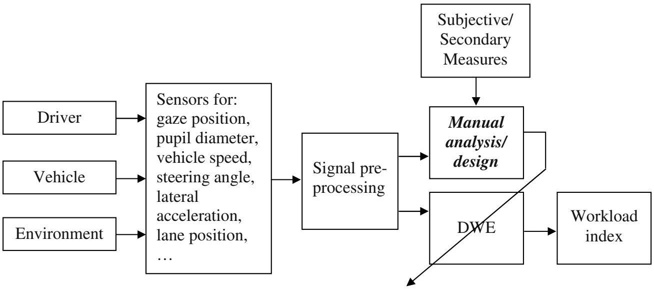


Fig. 2. The existing DWE system design process

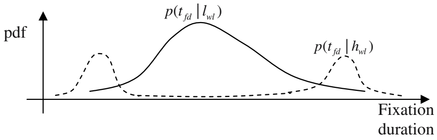


Fig. 3. The probability distribution function of fixation duration under high and low workload

the probability distribution function (pdf) of fixation duration under high workload ($p(t_{fd}|h_{wl})$) is multimodal, as shown in Fig. 3. With collected ocular data, one may estimate the conditional pdfs ($p(t_{fd}|h_{wl})$ and $p(t_{fd}|l_{wl})$) and the prior probabilities for high and low workload ($P(h_{wl})$ and $P(l_{wl})$). With this knowledge, standard Bayesian analysis will tell the probability of high workload given the fixation duration,

$$p(h_{wl}|t_{fd}) = \frac{p(t_{fd}|h_{wl})P(h_{wl})}{p(t_{fd}|h_{wl})P(h_{wl}) + p(t_{fd}|l_{wl})P(l_{wl})}.$$

3 The Proposed Approach: Learning-Based DWE

We proposed a learning-based DWE design process a few years ago [17, 18]. Under this framework, instead of manually analyzing the significance of individual features or a small set of features, the whole set of features are considered simultaneously. Machine-learning techniques are used to tune the DWE system, and derive an optimized model to index workload.

Machine learning is concerned with the design of algorithms that encode inductive mechanisms so that solutions to broad classes of problems may be derived from examples. It is essentially data-driven and is fundamentally different from traditional AI such as expert systems where rules are extracted mainly by human experts. Machine learning technology has been proved to be very effective in discovering the underlying structure of data and, subsequently, generate models that are not discovered from domain knowledge. For example, in the automatic speech recognition (ASR) domain, models and algorithms based on machine learning outperform all other approaches that have been attempted to date [19]. Machine learning has found increasing applicability in fields as varied as banking, medicine, marketing, condition monitoring, computer vision, and robotics [20].

Machine learning technology has been implemented in the context of driver behavior modeling. Kraiss [21] showed that a neural network could be trained to emulate an algorithmic vehicle controller and that individual human driving characteristics were identifiable from the input/output relations of a trained network. Forbes et al. [22] used dynamic probabilistic networks to learn the behavior of vehicle controllers that simulate good drivers. Pentland and Liu [23] demonstrated that human driving actions, such as turning, stopping, and changing lane, could be accurately recognized very soon after the beginning of the action using Markov dynamic model (MDM). Oliver and Pentland [24] reported a hidden Markov model-based framework to predict the most likely maneuvers of human drivers in realistic driving scenarios. Mitrović [25] developed a method to recognize driving events, such as driving on left/right curves and making left/right turns, using hidden Markov models. Simmons et al. [26] presented a hidden Markov model approach to predict a driver's intended route and destination based on observation of his/her driving habits.

Thanks to the obvious relation between driver behavior and driver workload, our proposal of learning-based DWE is a result of the above progress. Similar ideas were proposed by other researchers [27, 28] around the time frame of our work and many followup works have been reported ever since [29–31].

3.1 Learning-Based DWE Design Process

The learning-based DWE design process is shown in Fig. 4. Compared to the one shown in Fig. 2, the new process replaces the module of manual analysis/design with a module of a machine learning algorithm, which is the key to learning-based DWE.

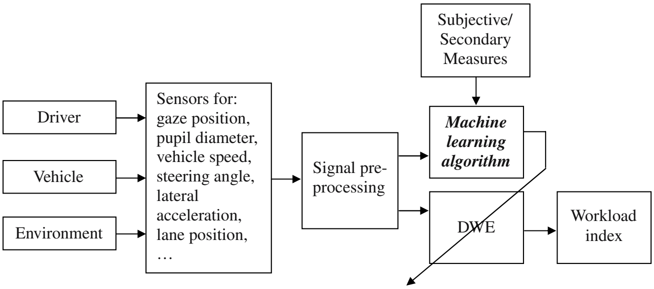

Subjective/ Secondary Measures
Driver
Vehicle
Environment
Sensors for: gaze position, pupil diameter, vehicle speed, steering angle, lateral acceleration, lane position, …
Signal pre-processing
Machine learning algorithm
DWE
Workload index

Fig. 4. The learning-based DWE design process

A well-posed machine learning problem requires a task definition, a performance measure, and a set of training data, which are defined as follows for the learning-based DWE:

Task: Identify driver's cognitive workload level in a time interval of reasonable length, e.g., every few seconds.
Performance measure: The rate of correctly estimating driver's cognitive workload level.
Training data: Recorded driver's behavior including both driving performance and physiological measures together with the corresponding workload levels assessed by subjective measures, secondary task performance, or task analysis.

In order to design the learning-based DWE algorithm, training data need to be collected while subjects drive a vehicle in pre-designed experiments. The data includes the sensory information of the maneuvering of the vehicle (e.g., lane position, which reflects driver's driving performance) and the driver's overt behavior (e.g., eye movement and heart beat), depending on the availability of the sensor on the designated vehicle. The data also includes the subjective workload ratings and/or the secondary-task performance ratings of the subjects. These ratings serve as the training labels.

After some preprocessing on the sensory inputs, such as the computation of mean and standard deviation, the data is fed to a machine-learning algorithm to extract the relationship between the noisy sensory information and the driver's workload level. The computational intelligence algorithm can be decision tree, artificial neural network, support vector machine, or methods based on discriminant analysis. The learned estimator, a mapping from the sensory inputs to the driver's cognitive workload level, can be a set of rules, a look-up table, or a numerical function, depending on the algorithm used.

3.2 Benefits of Learning-Based DWE

Changing from a manual analysis and modeling perspective to a learning-based modeling perspective will gain us much in terms of augmenting domain knowledge, and efficiently and effectively using data.

A learning process is an automatic knowledge extraction process under certain learning criteria. It is very suitable for a problem as complicated as workload estimation. Machine learning techniques are meant for analyzing huge amounts of data, discovering patterns, and extracting relationships. The use of machine-learning techniques can save labor-intensive manual process to derive combined workload index and, therefore, can take full advantage of the availability of various sensors. Finally, most machine learning techniques do not require the assumption of the unimode Gaussian distribution. In addition to the advantages discussed above, this change makes it possible for a DWE system to be adaptive to individual drivers. We will come back to this issue in Sect. 7.

Having stated the projected advantages, we want to emphasize that the learning-based approach benefits from the prior studies on workload estimation, which have identified a set of salient features, such as fixation duration, pupil diameter, and lane position deviation. We utilize the known salient features as candidate inputs.

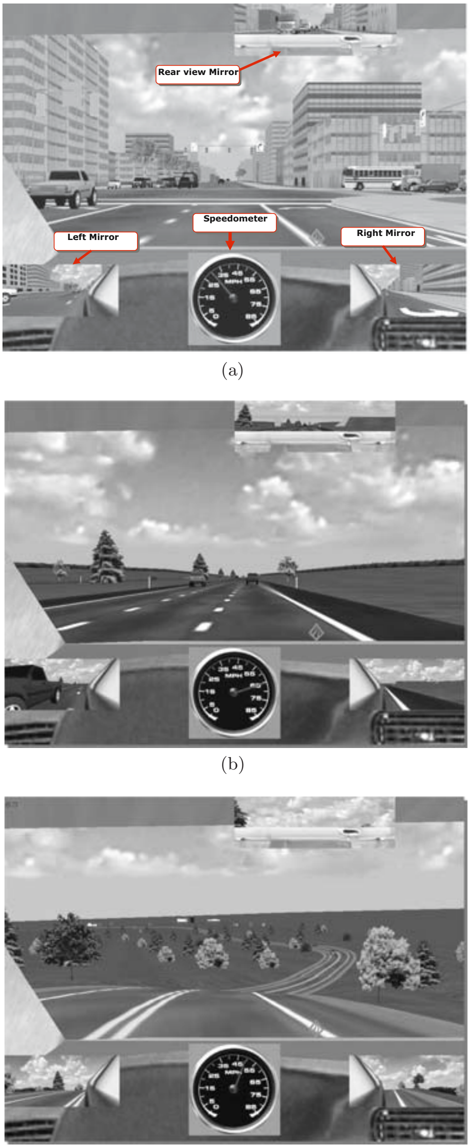
4 Experimental Data

Funded by GM R&D under a contract, researchers from the University of Illinois at Urbana-Champaign conducted a driving simulator study to understand driver's workload. The data collected in the simulator study was used to conduct some preliminary studies on learning-based DWE as presented in this chapter.

The simulator system has two Ethernet-connected PCs running GlobalSim's Vection Simulation Software version 1.4.1, a product currently offered by DriveSafety Inc. (http://www.drivesafety.com). One of the two computers (the subject computer) generates the graphical dynamic driving scenes on a standard 21-in monitor with the resolution of 1,024 × 768 (Fig. 5). Subjects use a non-force feedback Microsoft Sidewinder USB steering wheel together with the accelerator and brake pedals to drive the simulator. The second computer (the experimental computer) is used by an experimenter to create driving scenarios, and collect and store the simulated vehicle data, such as vehicle speed, acceleration, steering angle, lateral acceleration, lane position, etc. To monitor the driver's behavior closely, a gaze tracking system is installed on the subject computer and running at the same time as the driving simulation software. The gaze tracking system is an Applied Science Lab remote monocular eye tracker, Model 504 with pan/tilt optics and a head tracker. It measures the pupil diameter and the point of gaze at 60 Hz with an advertised tracking accuracy of about ±0.5 degree [32]. The gaze data is also streamed to and logged by the experimenter computer. A complete data table is shown in Table 1.

Twelve students participated in the experiment. Each participant drove the simulator in three different driving scenarios, namely, highway, urban, and rural (Fig. 5). There were two sessions of driving for each scenario, each lasting about 8–10 min In each session, the participants were asked to perform secondary tasks (two verbal tasks and two spatial-imagery tasks) during four different 30-s periods called *critical periods*. In the verbal task, the subjects were asked to name words starting with a designated letter. In the spatial-imagery task, the subjects were asked to imagine the letters from A to Z with one of the following characteristics: (a) remaining unchanged when flipped sideways, (b) remaining unchanged when flipped upside down, (c) containing a close part such as "A", (d) having no enclosed part, (e) containing a horizontal line, (f) containing a vertical line. Another four 30-s critical periods were identified as control sessions in each session, during which no secondary tasks were introduced. In the following analysis, we concentrate on the data during the critical periods. In total, there were $12\ subjects \times 3\ scenarios \times 2\ sessions \times 8\ critical\ periods/session = 576$ critical periods. Because of some technical difficulties during the experiment, the data from some of the critical periods were missing, which ended up with a total of 535 critical periods for use. The total number of data entries is $535\ critical\ periods \times 30\,s \times 60\,Hz = 1{,}036{,}800$.

In the simulator study, there was no direct evidence of driver's workload level, such as the subjective workload assessment. However, workload level for each data entry was needed in order to evaluate the idea of learning-based DWE. We made an assumption that drivers bear more workload when engaging in the secondary tasks. As we know, the primary driving task includes vehicle control (maintaining the vehicle in a safe location with an appropriate speed), hazard awareness (detecting hazards and handling the elicited problems), and navigation (recognizing landmarks and taking actions to reach destination) [33]. The visual perception, spatial cognitive processing, and manual responses involved in these subtasks all require brain resources. In various previous studies, many secondary mental tasks, such as verbal and spatial-imagery tasks, have been shown to compete for the limited brain resources with the primary driving task. The secondary mental tasks affect the drivers by reducing their hazard detection capability and delaying the decision-making time [13, 14, 34]. Although a driver may respond to the multi-tasks by changing resource allocation strategy to make the cooperation more efficient, in general, the more tasks a driver is conducting at a time, the more resources he/she is consuming and, therefore, the higher workload he/she is bearing. Based on this assumption, we labeled all the sensor inputs falling into the dual-task critical periods with high workload. The sensor inputs falling into the control critical periods were labeled with low workload. We understand that driver's workload may fluctuate during a critical period depending on the driving condition and her actual involvement in the secondary task.



(a)

(b)

(c)

Fig. 5. The screen shots of the driving scene created by the GlobalSim Vection Simulation Software in three different driving scenarios: (**a**) urban, (**b**) highway, and (**c**) rural

Table 1. The vehicle and gaze data collected by the simulator system

Vehicle	Velocity, lane position, speed limit, steer angle, acceleration, brake, gear, horn, vehicle heading, vehicle pitch, vehicle roll, vehicle X, vehicle Y, vehicle Z, turn signal status, latitudinal acceleration, longitudinal acceleration, collision, vehicle ahead or not, headway time, headway distance, time to collide, terrain type, slip.
Gaze	Gaze vertical position, gaze horizontal position, pupil diameter, eye to scene point of gaze distance, head x position, head y position, head z position, head azimuth, head elevation, head roll.

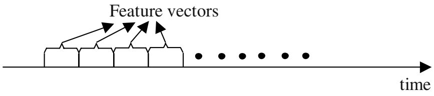


Fig. 6. The rolling time windows for computing the feature vectors

Table 2. The features used to estimate driver's workload

Feature number	Features
1	m_{spd}: mean vehicle velocity
2	V_{spd}: standard deviation of vehicle velocity
3	m_{lp}: mean lane position
4	V_{lp}: standard deviation of vehicle lane position
5	m_{str}: mean steering angle
6	V_{str}: standard deviation of steering angle
7	m_{acc}: mean vehicle acceleration
8	V_{acc}: standard deviation of vehicle acceleration
9	m_{pd}: mean pupil diameter
10	V_{pd}: standard deviation of pupil diameter
11–18	n_i: number of entries for the gaze moving into region i, $i = 1, 2, ..., 8$
19–26	ts_i: portion of time the gaze stayed in region i, $i = 1, 2, ..., 8$
27–34	tv_i: mean visit time for region i, $i = 1, 2, ..., 8$

5 Experimental Process

We preprocessed the raw measurements from the sensors and generated vectors of features over the fixed-size rolling time windows as shown in Fig. 6. Table 2 lists all the features we used. The "regions" in Table 2 refer to the eight regions of driver's front view as shown in Fig. 7. It is desirable to estimate the workload at a frequency as high as possible. However, it is not necessary to assess it at a frequency of 60 Hz because the driver's cognitive status does not change at that high rate. In practice, we tried different time window sizes, of which the largest was 30 s, which equals the duration of a critical period.

While many learning methods can be implemented, such as Bayesian learning, artificial neural networks, hidden Markov models, case based reasoning, and genetic algorithms, we used decision tree learning , one of the most widely used methods for inductive inference, to show the concept.

A decision tree is a hierarchical structure, in which each node corresponds to one attribute of the input attribute vector. If the attribute is categorical, each arc branching from the node represents a possible value of that attribute. If the attribute is numerical, each arc represents an interval of that attribute. The leaves of the tree specify the expected output values corresponding to the attribute vectors. The path from the root to a leaf describes a sequence of decisions made to generate the output value corresponding to an attribute vector. The goal of decision-tree learning is to find out the attribute and the splitting value for each node of the decision tree. The learning criterion can be to reduce entropy [35] or to maximize t-statistics [36], among many others.

For the proof-of-concept purpose, we used the decision-tree learning software, See5, developed by Quinlan [35]. In a See5 tree, the attribute associated with each node is the most informative one among

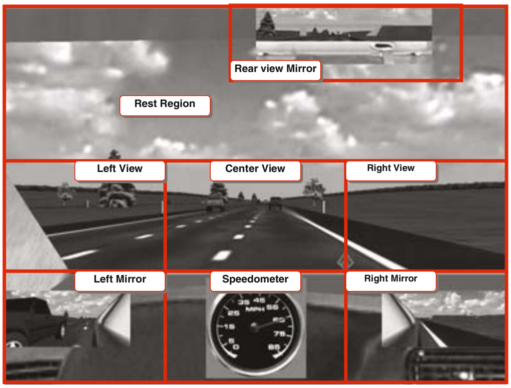


Fig. 7. The screen of the driving scene is divided into eight regions in order to count the gaze entries in each region. The region boundaries were not shown on the screen during the experiment

the attributes not yet considered in the path from the root. The significance of finding the most informative attribute is that making the decision on the most informative attribute can reduce the uncertainty about the ultimate output value to the highest extent.

In information theory, the uncertainty metric of a data set S is defined by entropy $H(S)$,

$$H(S) = -\Sigma_{i=1}^{c} P_i log_2(P_i),$$

where, S is a set of data, c is the number of categories in S, and P_i is the proportion of category i in S. The uncertainty about a data set S when the value of a particular attribute A is known is given by the conditional entropy $H(S|A)$,

$$H(S|A) = -\Sigma_{v \in Value(A)} P(A = v) H(S|A = v),$$

where $Value(A)$ is the set of all possible values for attribute A, and $P(A = v)$ is the proportion of data in S, whose attribute A has the value v. If we use S_v to represent the subset of S for which attribute A has the value v, the conditional entropy $H(S|A)$ can be rewritten as,

$$H(S|A) = -\Sigma_{v \in Value(A)} \frac{|S_v|}{|S|} H(S_v),$$

where $|\bullet|$ is the number of data points in the respective data set. As a result, the information gain of knowing the value of attribute A is defined as,

$$Gain(S, A) = H(S) - \Sigma_{v \in Value(A)} \frac{|S_v|}{|S|} H(S_v),$$

So, the most informative attribute A_{mi} is determined by,

$$A_{mi} = \arg \max_{for\ all\ As} Gain(S, A).$$

To improve the performance, a popular training algorithm called adaptive boosting or AdaBoost was used. The AdaBoost algorithm [37, 38] is an interactive learning process which combines the outputs of a set of N "weak" classifiers trained with different weightings of the data in order to create a "strong" composite classifier. A "weak" learning algorithm can generate a hypothesis that is slightly better than random for any data distribution. A “strong” learning algorithm can generate a hypothesis with an arbitrarily low error rate, given sufficient training data. In each successive round of weak classifier training, greater weight is placed on those data points that were mis-classified in the previous round. After completion of N rounds, the N weak classifiers are combined in a weighted sum to form the final strong classifier.

Freund and Shapire have proven that if each weak classifier performs slightly better than a random classifier, then the training error will decrease exponentially with N. In addition, they showed that the test set or generalization error is bounded with high probability by the training set error plus a term that is proportional to the square root of N/M, where M is the number of training data. These results show that, initially, AdaBoost will improve generalization performance as N is increased, due to the fact that the training set error decreases more than the increase of the N/M term. However, if N is increased too much, the N/M term increases faster than the decrease of the training set error. That is when overfitting occurs and the reduction in generalization performance will follow. The optimum value of N and the maximum performance can be increased by using more training data. AdaBoost has been validated in a large number of classification applications. See5 incorporates AdaBoost as a training option. We utilized boosting with $N = 10$ to obtain our DWE prediction results. Larger values of N did not improve performance significantly.

6 Experimental Results

The researchers from the University of Illinois at Urbana-Champaign reported the effect of secondary tasks on driver’s behavior in terms of the following features:

- Portion of gaze time in different regions of driver’s front view
- Mean pupil size
- Mean and standard deviation of lane position
- Mean and standard deviation of the vehicle speed

The significance of the effect was based on the analysis of variance (ANOVA) [39] with respect to each of these features individually. The general conclusion was that the effect of secondary tasks on some features was significant, such as speed deviation and lane position. However, there was an interaction effect of driving environments and tasks on these features, which means the significance of the task effect was not consistent over different driving environments [40].

It should be noted that ANOVA assumes Gaussian statistics and does not take into account the possible multi-modal nature of the feature probability distributions. Even for those features showing significant difference with respect to secondary tasks, a serious drawback of ANOVA is that this analysis only tells the significance on average. We can not tell from this analysis how robust an estimator can be if we use the features on a moment-by-moment basis.

In the study presented in this chapter, we followed two strategies when conducting the learning process, namely driver-independent and driver-dependent. In the first strategy, we built models over all of the available data. Depending on how the data were allocated to the training and testing sets, we performed training experiments for two different objectives: subject-level and segment-level training, the details of which are presented in the following subsection. In the second training strategy, we treated individual subjects’ data separately. That is, we used part of one subject’s data for training and tested the learned estimator on the rest data of the same subject. This is a driver-dependent case. The consideration here is that since the workload level is driver-sensitive, individual difference makes a difference in the estimation performance.

The performance of the estimator was assessed with the cross-validation scheme, which is widely adopted by the machine learning community. Specifically, we divided all the data into subsets, called *folds*, of equal sizes. All the folds except one were used to train the estimator while the left-out fold was used for performance evaluation. Given a data from the left-out fold, if the estimation of the learned decision tree was the same as the label of the data, we counted it as a success. Otherwise, it was an error. The correct estimation rate, r_c, was given by,

$$r_c = \frac{n_c}{n_{tot}},$$

where n_c was the total number of successes and n_{tot} was the total number of data entries. This process rotated through each fold and the average performance on the left-out folds was recorded. A cross validation process involves ten folds (ten subsets) is called a *tenfold cross validation*. Since the estimator is always evaluated on the data disjoint from the training data, the performance evaluated through the cross validation scheme correctly reflects the actual generalization capability of the derived estimator.

6.1 Driver-Independent Training

The structure of the dataset is illustrated schematically in the upper part of Fig. 8. The data can be organized into a hierarchy where individual subjects are at the top. Each subject experienced eight critical periods with single or dual tasks under urban, highway, and rural driving scenarios. Each critical period can be divided into short time windows or segments to compute the vectors of features. For clarity we do not show all subjects, scenarios, and critical periods in Fig. 8.

In the subject-level training, all of the segments for one subject were allocated to either the training or testing set. The data for any individual subject did not appear in both the training and testing sets. The subject-level training was used for estimating the workload of a subject never seen before by the system. It is the most challenging workload estimation problem. In the segment-level training, segments from each critical period were allocated disjointly to both the training and testing sets. This learning problem corresponds to estimating workload for individuals who are available to train the system beforehand. It is an easier problem than the subject-level training.

The test set confusion table for the subject-level training with 30-s time window is shown in Table 3. We were able to achieve an overall correct estimation rate of 67% for new subjects that were not in the training set using segments equal in length to the critical periods (30 s). The rules converted from the learned decision tree are shown in Fig. 9. Reducing the segment length increased the number of feature vectors in the training

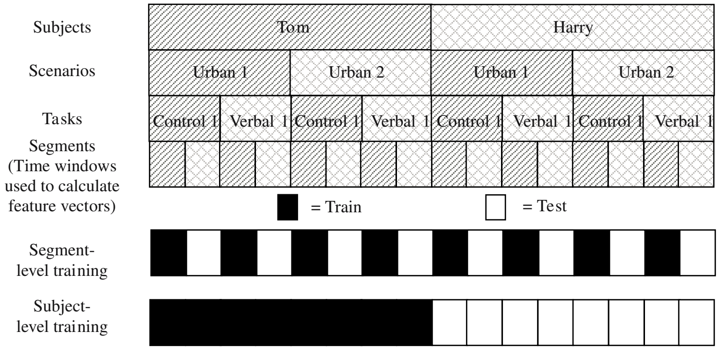


Fig. 8. Allocation of time windows or segments to training and testing sets for two training objectives

Table 3. The test set confusion table for the *subject-level* driver-independent training with time window size of 30 s

Workload estimation	Dual-task	Single-task	Total
High workload	186	93	–
Low workload	84	172	–
Correct estimation rate (%)	69	65	67

In the table, dual-task refers to the cases when the subjects were engaged in both primary and secondary tasks and, therefore, bore high workload. Similarly, single-task refers to the cases when the subjects were engaged only in primary driving task and bore low workload

```
Rule 1/1: (41.4/4.6, lift 2.2)
        F10 > 3.526
        F21 <= 0.0635
        -> class High  [0.871]

Rule 1/2: (13.9/1.5, lift 2.1)
        F01 <= 0.3084
        F21 <= 0.0635
        F23 <= 0.0023
        -> class High  [0.841]

Rule 1/3: (3.8, lift 2.1)
        F01 <= 2.4056
        F21 <= 0.0635
        F30 > 50.5714
        -> class High  [0.828]

Rule 1/4: (3.8, lift 2.1)
        F08 <= 0.0243
        F15 <= 0.0056
        F26 > 0.3819
        -> class High  [0.828]

Rule 1/5: (11.5/1.5, lift 2.1)
        F01 > 2.4056
        F21 <= 0.0635
        -> class High  [0.812]

Rule 1/6: (21.4/3.8, lift 2.0)
        F09 <= 21.2443
        -> class High  [0.794]

Rule 1/7: (9.9/1.5, lift 2.0)
        F01 <= -1.563
        F15 <= 0.0174
        -> class High  [0.788]

Rule 1/8: (11.5/3.1, lift 1.8)
        F15 > 0.0174
        F21 > 0.0635
        F28 <= 1.25
        -> class High  [0.699]

Rule 1/9: (45.2, lift 1.2)
        F09 > 21.2443
        F15 > 0.0056
        F15 <= 0.0174
        F21 > 0.0635
        -> class Low  [0.979]

Rule 1/10: (92.1/1.5, lift 1.2)
        F01 > -1.563
        F08 > 0.0243
        F09 > 21.2443
        F15 <= 0.0174
        F21 > 0.0635
        -> class Low  [0.973]

Rule 1/11: (31.6, lift 1.2)
        F01 > 0.3084
        F01 <= 2.4056
        F10 <= 3.526
        F30 <= 50.5714
        -> class Low  [0.970]

Rule 1/12: (28.7, lift 1.2)
        F01 <= 2.4056
        F10 <= 3.526
        F23 > 0.0023
        F30 <= 50.5714
        -> class Low  [0.967]
```

Fig. 9. The rules converted from the driver-independent decision tree. Each rule is characterized by the statistics (N/E, lift L), where N is the number of training cases covered by the rule, E (if shown) is the number of them that do not belong to the rule's class, and L is the estimated accuracy of the rule (the number in square brackets, e.g., [0.871]) divided by the prior probability of the rule's class. The reason that N and E may be non-integral numbers is that when the value of an attribute in the tree is not known, See5 splits the case and sends a fraction down each branch. Please refer to Table 2 for the attribute indices. For example, F01 refers to feature 1 in Table 2

Table 4. The correct estimation rates in the *subject-level* driver-independent training with different time window sizes

Time window size (s)	1.9	7.5	30
Correct estimation rate (%)	61	63	67

Table 5. The test set confusion table for the *segment-level* driver-independent training with time window size of 30 s

Workload estimation	Dual-task	Single-task	Total
High workload	219	50	–
Low workload	51	215	–
Correct estimation rate (%)	81	81	81

In the table, dual-task refers to the cases when the subjects were engaged in both primary and secondary tasks and, therefore, bore high workload. Similarly, single-task refers to the cases when the subjects were engaged only in primary driving task and bore low workload

set but the variance was also increased due to the shorter averaging period. The overall effect of reducing the segment length on subject-level training was a degradation in performance (see Table 4).

The test set confusion table for segment-level training is shown in Table 5 for segments that were equal to the critical period (30 s). In this case, the data from all subjects was disjointly distributed in either the training or testing sets. Compared to the results for subject-level training, the correct estimation rate was

Table 6. The correct estimation rates in the *segment-level* driver-independent training with different time window sizes

Time window size (s)	1.9	7.5	30
Correct estimation rate (%)	71	78	81

Table 7. The correct estimation rates of driver-dependent training with 0.5 s time window under tenfold cross validation

Subjects	1	2	3	4	5	6	7	8	9	10	11	12	Average
Correct estimation rate (%)	85.7	80.6	86.4	81.8	77.8	82.4	88.0	87.9	81.3	93.4	90.8	88.4	85.4
Standard deviation (%)	0.7	0.8	0.6	0.8	0.9	0.9	0.7	0.7	0.8	0.4	0.5	0.5	0.69

Table 8. The average correct estimation rates of driver-dependent training with different time window sizes under tenfold cross-validation

Time window size (s)	0.5	2	5
Correct estimation rate (%)	85.4	80.1	78.2

improved from 67 to 81%. Similar to the case of subject-level training, the overall effect of reducing the segment length on segment-level training was a degradation in performance (see Table 6).

Note that, considering the significant reduction of segment length, the degradation of estimation performance shown in both Tables 4 and 6 seems to be quite acceptable.

6.2 Driver-Dependent Training

The cognitive capacity required to perform the same tasks varies from person to person [2]. So does the workload level for different drivers. This may violate one of the assumptions for any learning-based estimators, i.e., the distribution of the training data is the same as the distribution of the testing data. As a result, the workload estimator obtained from the data of some drivers may not yield good estimation performance when applied to other drivers' data. The driver-dependent training strategy is adopted to evaluate and address this issue.

In driver-dependent training strategy, the data from a single subject was divided into disjoint training and test sets for tenfold cross validation. The ten correct estimation rates for each subject was averaged to represent the estimation performance over that subject's data. The standard deviation of the ten correct estimation rates was also calculated to evaluate the robustness of the estimation performance. Since there was only limited data from each subject, we used relatively short time-window size in order to make sure there was enough data to train the decision tree. Table 7 shows the performance of the learned estimator for each of the twelve subjects when the size of the time window was 0.5 s. The highest correct estimation rate reached 93.4% and the average performance was 85.4%. Because of the limited amount of data, we were only able to label the data with a temporal resolution of 30 s (the duration of a critical period). This labeling strategy failed to reflect the possible workload fluctuation within each critical period. As a result, it introduced errors to both training and evaluation. In addition, the level of fluctuation differed for different subjects, which is probably the major reason for the performance variance among subjects. However, given that the standard deviation of the performance for each subject is under one percentage point, the estimator design is highly robust.

We tried different window sizes for the feature vector computation and the average correct estimation rates are listed in Table 8. As the windows size got larger, the amount of training data was reduced, which contributed to the degraded performance of the estimator.

6.3 Feature Combination

To understand the contribution of different subsets of the features to the estimation performance, we trained See5 decision trees using various combinations of the features with the segment-level training. The time window size was set to be 30 s. The performance for various groupings of features is shown in Table 9. The best performance we were able to obtain was 81% using all of features. The eyegaze-related features were more predictive than the driving-performance features since removing all other features (the driving-performance features) from the training set reduced performance by only 1 percentage point from 81 to 80%. Removing the eyegaze-related features (feature 9–34), on the other hand, reduced performance from 81 to 60%. All of the driving-performance and eyegaze-related features, however, contributed to workload detection accuracy, albeit in varying degrees.

From past experience we have found that there is a high correlation between feature frequency in a decision tree and the predictive power of the feature. We did an analysis on the learned See5 trees by counting how many times each of the feature appeared. A histogram of feature usage in the learned rule sets is shown in Fig. 10. The feature with the highest frequency is the standard deviation of pupil diameter, which is consistent with the known correlation between pupil changes and workload [41], especially in the controlled illumination conditions of the driving simulator.

Among the driving-performance features, vehicle speed (F1) and speed deviation (F2) have the highest frequency of occurrence. The selection of F2 by the learning algorithm is consistent with the ANOVA results done by the University of Illinois at Urbana-Champaign. The ANOVA analysis did not include vehicle speed since the vehicle speed was largely determined by the driving scenarios such as the speed limit of the road and was not considered a good predictor of workload. However, it is understandable that a driver under high workload would tend to have higher speed deviation when the vehicle speed is high, compared to the case when the vehicle speed is low. In other words, under high workload, speed deviation correlates with vehicle

Table 9. The correct estimation rates in the 30-s segment-level driver independent training with various feature combinations

Feature combination	Number of features	Correct estimation rate (%)
All features (F01-F34)	34	81
Eyegaze-related features (F09-F34)	26	80
All but pupil-diameter features (All but F09-F10)	32	70
Pupil-diameter features only (F09-F10)	2	61
Driving-performance features (F01-F08)	8	60

Please refer to Table 2 for the feature indices

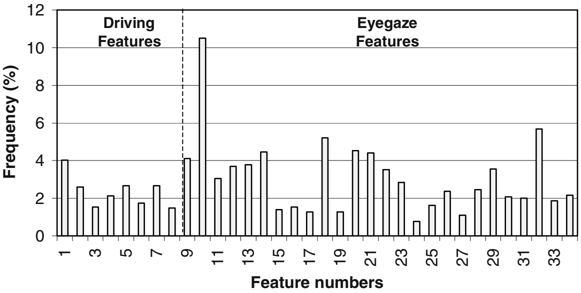


Fig. 10. Histogram of feature frequency in the workload estimator trained by See5. Please refer to Table 2 for the features

speed. This correlation together with the correlation between speed deviation and workload may have made the decision-tree algorithm to pick vehicle speed relatively frequently in order to determine the workload.

It is noteworthy, however, that all of the features are used at least to some extent by the workload estimator, which indicates that all of the feature have some predictive power. The best estimation performance should be obtained when all of the features are used together.

7 Conclusions and Future Work

Compared to the existing practice, the proposed learning-based DWE does not require sophisticated domain knowledge or labor-intensive manual analysis and modeling efforts. With the help of mature machine learning techniques, the proposed method provides an efficient way to derive models over a large number of sensory inputs, which is critical for a problem as complicated as cognitive workload estimation.

The preliminary experimental results of learning-based DWE show that a driver's cognitive status could be estimated with an average correct rate of more than 85% for driver-dependent training. Although this performance is still a long way from a practical DWE system, it is very encouraging especially considering that the estimation was conducted at a rate of twice a second. As we anticipated, the estimation performance was not as good in the driver-independent case as in the driver-dependent case, which restates the importance that a DWE system should capture individual difference.

Recall that our labeling strategy was that all the sensor inputs in the dual-task critical periods had high workload label and those in the control critical periods were labeled with low workload. Within each critical period, the driver might switch their attention between the driving and secondary tasks and, thus, change their actual workload level, which may have contributed to the error. In addition, in the conducted experiments, we used a general machine learning package. Usually a customized algorithm has a better performance than a general-purpose one. It will be interesting to see how much improvement we can achieve with a fine tuned learning algorithm dedicated to our specific problem.

The significance of learning-based DWE is not limited to quickly getting good estimation results. With the manual analysis and modeling methodology, a DWE system works in a static mode. That is, the rules for workload estimation and the thresholds specified in the rules are usually determined through certain human factors studies before the vehicles are sold to the customers. Whether the induced rules represent the customer population very much depends on how the sample subjects are selected and the sample size of the studies. It is also questionable whether one set of rules fit customers with different genders, different ages, different driving experiences, different education background, and etc. Ideally, the rules for DWE should be tailored to each individual driver in order to capture individual difference.

On the other hand, the learning-based method provides a possibility to automate the adaptation process. Figure 11 illustrates one possible implementation of adaptive DWE. Instead of collecting the training data in a pre-designed experiments, adaptive DWE collects data when the end customer is driving the vehicle. Compared to Fig. 4, Fig. 11 replaces the Subjective/Secondary Measure module with a Driver Workload Assessment module. The Driver Workload Assessment module allows the driver, while driving, to occasionally submit the subjective assessment of workload level (e.g., 1, 2,...10, or high, medium, low) through an driver–vehicle interaction mechanism, such as a push button or voice user interface. Such action triggers the collection of a running cache of streaming sensor data. With both the sensory readings and the subjective assessment (the label of workload level) in place, the machine-learning algorithm module can update the DWE module without manual interference, using pre-specified learning mechanisms, e.g., decision tree learning with reducing entropy as the learning criterion.

Having stated the advantages that the learning-based method may deliver, it is important to understand that not all problems are solved. For example, there is still a bottleneck in the learning-based DWE design process, i.e., the subject/secondary task measures module that provides the training data with the labels. We need to address the automatic label generation issue in order to reach a full automatic DWE design process.

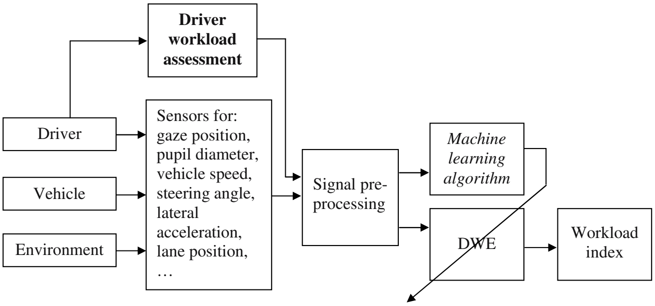

Driver workload assessment
Driver
Vehicle
Environment
Sensors for: gaze position, pupil diameter, vehicle speed, steering angle, lateral acceleration, lane position, …
Signal pre-processing
Machine learning algorithm
DWE
Workload index

Fig. 11. A possible implementation of adaptive DWE

Acknowledgments

The authors would like to thank X.S. Zheng, G. McConkie, and Y.-C. Tai for collecting the data under a contract with GM, which made this research possible.

References

1. R.D. O'Donnel and F.T. Eggemeier, "Workload assessment methodology," in *Handbook of Perception and Human Performance, Vol II, Cognitive Processes and Performance*, K.R. Boff, L. Kaufman, and J.P. Thomas, Eds. Wiley, New York, 1986.
2. C.D. Wickens and J.G. Hollands, *Engineering Psychology and Human Performance*, Prentice-Hall, Upper Saddle River, NJ, 3rd edition, 2000.
3. S.G. Hart and L.E. Seaveland, "Development of NASA-TLS (task load index): results of empirical and theoretical research," in *Human Mental Workload*, P.A. Hancock and N. Meshkati, Eds., North Holland, Amsterdam, 1988.
4. G.B. Reid and T.E. Nygren, "The subjective workload assessment technique: a scaling procedure for measuring mental workload," in *Human Mental Workload*, P.A. Hancock and N. Meshkati, Eds., North Holland, Amsterdam, 1988.
5. L.S. Angell, R.A. Young, J.M. Hankey, and T.A. Dingus, "An evaluation of alternative methods for assessing driver workload in the early development of in-vehicle information systems," in *Proceedings of SAE Government/Industry Meeting*, Washington, DC, May 2002.
6. J. Hurwitz and D.J. Wheatley, "Using driver performance measures to estimate workload," in *Proceedings of the Human Factors and Ergonomics Society 46th Annual Meeting*, Santa Monica, CA, pp. 1804–1808, 2002.
7. C. Mayser, W. Piechulla, K.-E. Weiss, and W. König, "Driver workload monitoring," in *Proceedings of the Internationale Ergonomie-Konferenz der GfA, ISOES und FEES*, Mnchen, Germany, May 7–9, 2003.
8. O. Nakayama, T. Futami, T. Nakamura, and E.R. Boer, "Development of a steering entropy method for evaluating driver workload," in *Proceedings of SAE International Congress and Exposition Detroit*, Detroit, MI, March 1999.
9. C.D. Wickens, J. Helleberg, J. Goh, X. Xu, and B. Horrey, "Pilot task management: testing an attentional expected value model of visual scanning," Tech. Rep. ARL-01-14/NASA-01-7, University of Illinois, Aviation Research Lab, Savoy, IL, 2001.
10. S.K. Card, T.P. Moran, and A. Newell, "The model human processor: an engineering model of human performance," in *Handbook of Perception and Human Performance*, Wiley, New York, pp. 1–35, 1986.
11. D.E. Kieras and D.E. Meyer, "An overview of the epic architecture for cognition and performance with application to human-computer interaction," *Human-Computer Interaction*, 12, 391–438, 1997.
12. J.R. Anderson and C. Lebiere, *The Atomic Components of Thought*, Erlbaum, Mahwah, NJ, 1998.
13. M.A. Recarte and L.M. Nunes, "Effects of verbal and spatial-imagery task on eye fixations while driving," *Journal of Experimental Psychology: Applied*, 6(1), 31–43, 2000.
14. J.L. Harbluk and Y.I. Noy, "The impact of cognitive distraction on driver visual behaviour and vehicle control," Technical Report, Transport Canada, http://www.tc.gc.ca/roadsafety/tp/tp13889/en/menu.htm, February 2002.

15. M. Rahimi, R.P. Briggs, and D.R. Thom, "A field evaluation of driver eye and head movement strategies toward environmental targets and distracters," *Applied Ergonomics*, 21(4), 267–274, 1990.
16. P.R. Chapman and G. Underwood, "Visual search of dynamic scenes: event types and the role of experience in viewing driving situations," in *Eye Guidance in Reading and Science Perception*, G. Underwood, Ed., Elsevier, Amsterdam, pp. 369–393, 1998.
17. Y. Zhang, Y. Owechko, and J. Zhang, "Machine learning-based driver workload estimation," in *Proceedings of the 7th International Symposium on Advanced Vehicle Control*, Arnhem, Netherlands, August 23–27, 2004.
18. Y. Zhang, Y. Owechko, and J. Zhang, "Driver cognitive workload estimation: a data-driven perspective," in *Proceedings of 7th International IEEE Conference on Intelligent Transportation Systems*, Washington D.C., October 3–6, 2004.
19. T.M. Mitchell, *Machine Learning*, WCB/McGraw-Hill, Boston, MA, 1997.
20. P. Bhagat, *Pattern Recognition in Industry*, Elsevier Science, Oxford, UK, 2005.
21. K.-F. Kraiss, "Implementation of user-adaptive assistants with neural operator models," *Control Engineering Practice*, 3(2), 249–256, 1995.
22. J. Forbes, T. Huang, K. Kanazawa, and S. Russell, "Batmobile: Towards a bayesian automated taxi," in *Proceedings of International Joint Conference on Artificial Intelligence*, Montreal, Canada, pp. 1878–1885, 1995.
23. A.P. Pentland and A. Liu, "Modeling and prediction of human," *Behavior Neural Computation*, 11(2), 1999.
24. N. Oliver and A.P. Pentland, "A graphical models for driver behavior recognition in a smartcar," in *Proceedings of Intelligent Vehicles*, Detroit, Michigan, October 2000.
25. D. Mitrovic, "Reliable method for driving events recognition," *IEEE Transactions on Intelligent Transportation Systems*, 6(2), 198–205, 2005.
26. R. Simmons, B. Browning, Y. Zhang, and V. Sadekar, "Learning to predict driver route and destination intent," in *Proceedings of the 9th International IEEE Conference on Intelligent Transportation Systems*, Toronto, Canada, September 17–20, 2006.
27. A. Rakotonirainy and R. Tay, "Driver inattention detection through intelligent analysis of readily available sensors," in *Proceedings of The 7th International IEEE Conference on Intelligent Transportation Systems*, Washingtong DC, October 3–6, 2004.
28. A. Rakotonirainy and R. Tay, "In-vehicle ambient intelligent transport systems (i-vaits): towards an integrated research," in *Proceedings of The 7th International IEEE Conference on Intelligent Transportation Systems*, Washingtong DC, October 3–6, 2004.
29. C. Brooks, A. Rakotonirainy1, and F. Maire, "Reducing driver distraction through software," in *Proceedings Australasian Road Safety Research Policing Education Conference*, Wellington, New Zealand, November 2005.
30. L.M. Bergasa, J. Nuevo, M.A. Sotelo, R. Barea, and E. Lopez, "Visual monitoring of driver inattention". Chapter 2 in this volume.
31. Y. Liang, M.L. Reyes, and J.D. Lee, "Real-time detection of driver cognitive distraction using support vector machines," *IEEE Transactions on Intelligent Transportation Systems*, 8(2), 340–350, 2007.
32. Applied Science Lab Inc., *Eye Tracking System Instruction Manual – Model 504*, Applied Science Group, Bedford, MA, 2001.
33. C.D. Wickens, S.E. Gordon, and Y. Liu, *An Introduction to Human Factors Engineering*, Addison–Wesley, New York, 1998.
34. M.A. Recarte and L.M. Nunes, "Mental workload while driving: effects on visual search, discrimination, and decision making," *Journal of Experimental Psychology: Applied*, 9(2), 119–137, 2003.
35. J. Quinlan, *C4.5: Programs for Machine Learning*, Morgan Kaufmann, San Mateo, CA, 1993.
36. D. Chapman and L.P. Kaelbling, "Input generalization in delayed reinforcement learning: An algorithm and performance comparisons," in *Proceedings of the Twelfth International Joint Conference on Artificial Intelligence (IJCAI-91)*, Sydney, Australia, pp. 726–731, August 1991.
37. Y. Freund and R.E. Shapire, "A decision-theoretic generalization of on-line learning and an application to boosting," *Journal of Computer and System Sciences*, 55(1), 119–139, 1997.
38. Y. Freund and R.E. Shapire, "A short introduction to boosting," *Journal of Japanese Society for Artificial Intelligence*, 14(5), 771–780, 1999.
39. D.C. Montgomery and G.C. Runger, *Applied Statistics and Probability for Engineers*, Wiley, New York, 4th edition, 2006.
40. X.S. Zheng, G.W. McConkie, and Y.-C. Tai, "The effect of secondary tasks on drivers' scanning behavior," in *Proceedings of the 47nd Annual Meeting of the Human Factors and Ergonomics Society*, pp. 1900–1903, October 2003.
41. D. Kahneman, *Attention and Effort*, Prentice-Hall, Englewood Cliffs, NJ, 1973.