ECAI 2010
H. Coelho et al. (Eds.)
IOS Press, 2010

doi:10.3233/978-1-60750-606-5-157

Non-intrusive Detection of Driver Distraction using Machine Learning Algorithms

Fabio Tango[1], Marco Botta[1], Luca Minin[2] and Roberto Montanari[2]

Abstract. Driver's distraction has become an important and growing safety concern with the increasing use of the so-called In-Vehicle Information Systems (IVIS), such as cell-phones, navigation systems, etc. A very promising way to overcome this problem is to detect driver's distraction and thus to adopt in-vehicle systems accordingly, in order to avoid or mitigate the negative effects. The purpose of this paper is to illustrate a method for the non-intrusive detection of visual distraction, based on the vehicle dynamic data; in particular, we present and compare two models, applying Artificial Neural Networks (ANN) and Support Vector Machines (SVM) which are well-known data-mining methods.

Despite of what already done in literature, our method does not use eye-tracker data in the final classifier. With respect to other similar works, we regard distraction identification as a classification problem and, moreover, we extend the datasets, both in terms of data-points and of scenarios.

Data for training the models were collected using a static driving simulator, with real human subjects performing a specific secondary task (SURT) while driving. Different training methods, model characteristics and features selection criteria have been compared.

Potential applications of this research include the design of adaptive IVIS and of "smarter" Partially Autonomous Driving Assistance Systems (PADAS), as well as the evaluation of driver's distraction.

1 INTRODUCTION

It is well-known that the majority of road accidents (> 80%) are due to human error [1], or anyway human (wrong) behavior. In particular, more recent data have identified inattention as the primary cause of accidents [2]. Therefore, distraction is key-factor for driving safety: between 13% and 50% of crashes are attributed to driver distraction, resulting in as many as 5000 fatalities and $40 billion in damages each year [3-5] (studies carried out in USA, but also European ones confirm such values, e.g. http://www.aide-eu.org).

Moreover, the increasing use of the so-called In-Vehicle Information Systems (IVIS) – e.g. cell-phones, navigation systems, etc. – can induce additional source of potential distraction.

In this context, allowing drivers to take benefits from the use of these IVIS without diminishing safety is a big and important challenge. One promising strategy to deal with such a problem involves the classification of driver's state – distracted driver, in this case – in real time and then using this classification for a twofold goal [6]:

- to adapt IVIS technologies, in order to mitigate the effects of distraction
- to adapt the strategies of the so-called Partially Autonomous Driving Assistance Systems (PADAS), in order to minimize the effects of distraction on the driving task

Machine Learning (ML) technology may be able to provide the right algorithms to cope with such a challenge. ML, or Data Mining (DM), is the technology of searching large volumes of data for unknown patterns. It has been successfully applied in business, health care and other domains [7-8]. In particular, this technology can be applied to build a discrimination model that captures the differences in behavior between when people drive normally and when distracted.

The main goal of this paper is to present a non-intrusive approach to detect and classify driver's distraction, applying ML algorithms and using vehicle dynamic data, without using the driver's eye-movements as input to the model.

2 MODEL CONSTRUCTION OF DISTRACTION CLASSIFIERS

This Section presents the definition of distraction concept and how the distraction classifiers have been built.

2.1 Definition of distraction

The American Automobile Association Foundation for Traffic Safety defines driver distraction as occurring "*when a driver is delayed in the recognition of information needed to safely accomplish the driving task because some event, activity, object or person within or outside the vehicle compelled or tended to induce the driver's shifting attention away from the driving task*" ([11], p. 21). In particular, there are 3 types of distraction: visual, manual and cognitive. The experiments described in the next Section are especially focused on visual (mostly) and cognitive, since the visual research task implies always a contemporary presence of these two types of distraction. We have considered these two types of distraction, since they regarded as the key-factors in accidents due to human errors ([2-3]. As stated in [12-13], there are four main categories (among others) of measures used to assess distraction: primary task performance, secondary task performance, subjective measures and physiological measures. In particular, eye movements and driving performance are the most suitable measures to estimate cognitive and visual distraction unobtrusively and in real-time.

[1] University of Turin, Italy; e-mail: {tango, botta}@di.unito.it
[2] University of Modena and Reggio emilia, Italy; e-mail: {luca.minin, roberto.montanari}@unimore.it

2.2 Algorithms for the Classifier Construction

ML techniques seems to be very appropriated for this type of classification problem. From a more "philosophical" point of view, one of the most ambitious goal of automatic learning systems is to mimic the learning capability of humans and humans' capability of driving is widely based on experience, particularly on the possibility to learn from experience.

From a more technical point of view, data collected from vehicle dynamics and external environment are definitely non-linear. From literature, several studies have proved that in such situations machine learning approaches can outperform the traditional analytical methods. Moreover, also human's driver mental and physical behaviour is non-deterministic. So, since mental state of the drivers is not observable, no simple measure can index visual and cognitive distractions precisely [18-19].

Based on the results found in our previous work [20], we have selected two specific ML techniques: Support Vector Machines (SVM) and Artificial Neural Networks (ANN), as they represent a good trade-off between performances, implementation efforts and computational time.

2.2.1 Support Vector Machines

Support Vector Machines (SVM) are arguably one of the most important development in supervised classification of recent years. Proposed firstly by Vapnik in 1998, SVM are based on a statistical learning technique and can be used for pattern classification, as well as inference of non linear relationships between variables [14-15]. This method has been successfully applied to a wide variety of domains, such as: image processing (e.g. face recognition); text and speech recognition; bioinformatics (e.g. protein classification) [16]. SVM often achieve superior classification performance compared to other learning algorithms across most domains and tasks; they are fairly insensitive to the curse of dimensionality and are efficient enough to handle very large-scale classification in both sample and variables. The "classical" application of SVM concerns the binary classification tasks and this is the way in which they have been used in this research. The main idea of SVM is to implicitly map data to a higher dimensional space via a kernel function and then solve an optimization problem to identify the maximum-margin hyper-plane that separates training instances. The hyper-plane is based on a set of boundary training instances, called *support vectors*. New instances are classified according to the side of the hyper-plane they fall into. The optimization problem is most often formulated in a way that allows for non-separable data by penalizing misclassifications.

2.2.2 Feed-Forward Neural Networks

Artificial Neural Networks, or simply Neural Networks (ANN or NN), are an information processing system, which is inspired by biological nervous system (the brain) and that consists in a large number of highly interconnected processing elements, working together to solve specific problems [17]. In a neural network, signals are transmitted by connection links, characterized by an associated weight, which is multiplied along with the incoming signal (the input of the net) for any typical neural net. So the output signal is obtained by applying activations to the net input. One of the most important types of NN – used within our research – is the *Feed-forward Neural Networks* (FFNN). FFNN have a layered structure, where each layer consists of units receiving their input from units from a layer directly below and sending their output to units in a layer directly above the unit. There are no connections within a layer. The N_i inputs are fed into the first layer of $N_{h;1}$ hidden units. The input units are merely 'fan-out' units; no processing takes place in these units. The activation of a hidden unit is a function of the weighted inputs plus a bias. The output of the hidden units is distributed over the next layer of $N_{h;2}$ hidden units, until the last layer of hidden units, of which the outputs are fed into a layer of N_o output units. FFNN are considered *static networks*, since they have no feedback elements and contain no delays; the output is calculated directly from the input through feed-forward connections (despite of *dynamic networks*, where the output depends not only on the current input to the network, but also on the previous inputs, outputs, or states of the network).

2.3 Model Characteristics and Training Methods

By means of dedicated experiments (described in Section 3), using a static driving simulator, distraction and dynamic driving data have been collected.

Data of distraction constitute the target set, since we have adopted a supervised learning method. As detailed in next Section, distraction has been induced during driving, by means of a secondary visual research task reproduced on an in-vehicle display system. This task is called SURT (SUrrugate visual-Research Task – [9]). In particular, eyes-position has been extracted from videos with a video-processing laboratory software and transferred to a log file as Boolean values (1: eyes on the SURT; 0: eyes on the frontal screen). Then, the change of SURT status, from 0 to 1 and from 1 to 0, has been considered as the key factor to understand if the driver was distracted or not. In fact, from literature ([9] and [13]), if the drivers look away from the road for more than 2 seconds, they can be regarded as distracted. The switches of SURT status identify the period where drivers were engaged with secondary task completion. In such a way we get the target dataset for training the classifiers.

For what concerns the data about the vehicle dynamic of the Host-Vehicle (HV, that is the vehicle driven by the human user), the following variables have been collected:

- Speed [m/s]
- Time To Collision [s]
- Time To Lane Crossing [s]
- Steering Angle [deg]
- Lateral Shift [m]
- Position of the accelerator pedal [%]
- Position of the brake pedal [%]

The frequency of data collection was 20 Hz (1 data-point each 0.05s).

It is worth to note here that these variables constitute the inputs of the classifiers, so the eye-movements do not appear, but they have been used only to construct the target set. As discussed later on, this is one of the most relevant differences of this research with respect to the literature.

For each parameter in the list, the mean on different mobile windows has been computed, as method to group (summarize) the data. In fact, the punctual measures (speed, steering angle, etc.) may contain each minimum variation of variables, which would be used as valid data, while they represent rather a noise and a disturbance to the measure. So, such a noise can be minimized by using the mobile window average. Window size denotes the period over which eye movement and driving data were averaged. The comparisons window size serves to identify the appropriate length of data that can be summarized to reduce the noise of the input data without losing useful information. We chose four window sizes: 1 (raw data), 2, 3, and 5 steps. All the results achieved with these different summarizing parameters method have been compared (see next Section).

Following the ordinary procedure for supervised learning, each data set has been split in three different subsets:

- *Training data* (around 60% of the whole dataset) ⇒ These are presented to the network during training and the network is adjusted according to its error.
- *Checking data* (around 15% of the whole dataset) ⇒ These are used to measure network generalization and to halt training when generalization stops improving
- *Testing data* (around 25% of the whole dataset) ⇒ These have no effect on training and so provide an independent measure of network performance during and after training

3 DESCRIPTION OF THE EXPERIMENTS

A driving experiment has been conducted on a ScanerII (www.scaner2.com) car simulator, a fixed based system that comprises a mock-up of a car with real driving controls (i.e. seat, steering wheel, pedals, gear, handbrake), a digital simulated dashboard displaying a traditional instrumental panel and a frontal projection screen where the simulated environment is displayed to the driver.

Twenty participants with a previous experience on the driving simulator have been selected and divided into two groups: ten drivers in the age between 20 and 25 and ten between 30 and 45. Gender has been controlled (3 female and 7 male each group): a minimum amount of driver experience was required, in particular:

- At least 2 years of driving licence;
- At least 6000 km driven per year.

Participants were asked to drive for appreciatively 50 minutes on a simulated three lanes highway with a total length of 60 km: the driving task consisted in keeping the lane and driving at an average speed of 100 km/h at safety distance to the vehicles drivers encountered ahead. For the moment, we have considered a motorway scenario for a couple of reasons: first, it represents a more structured and controlled environment; second, it is more suitable for the integration with the ADAS application under investigation, the Adaptive Cruise Control (ACC).

As mentioned in the previous Section, distraction has been induced by means of a secondary visual research task, called SURT, reproduced on an in-vehicle display system (7'' TFT touch screen installed on the right-hand side of the car cabin). Figure 1 shows the situation.

SURT was chosen with the aim at evaluating the interferences caused by a generic visual search task rather than a specific IVIS (In Vehicle Information System). Like most commercial In-Vehicle Information Systems, it requires visual perception and manual response: such activities, according to Wickens' multiple resources model [11], requires the same mental resources of the driving task and is therefore more likely to interfere, possibly causing a degradation of driving task performances.

Each participant was asked to complete 16 secondary task sessions, each one lasting three minutes. When SURT is activated the display shows a black screen with 30 symbols (each 1.4 cm high), specifically: 14 blue circles, 15 red squares and 1 red circle. The screen is equally divided into two vertical sides and each time the SURT is presented the driver is asked to touch the side where the red circle is located. The time interval between two consecutive screens was pseudo-randomized between 3 and 9 seconds.

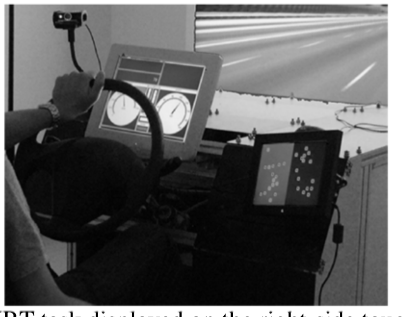

Figure 1 SURT task displayed on the right-side touch screen

The number of correct answers together with drivers' reaction time on the SURT (i.e. the difference between the instant the task is presented and the touch of the driver) have been recorded.

Drivers' visual attention allocation to the SURT has been assessed by means of two infrared cameras installed in front of the participants, capturing his/her eyes fixations. Two types of fixations have been triggered:

- Drivers' eyes on the SURT;
- Drivers' eyes on the frontal screen where the driving environment was simulated.

Eyes position has been extracted from videos with a video-processing laboratory software and transferred to a log file as Boolean values (1: eyes on the SURT; 0: eyes on the frontal screen): these data have been then aligned to the recording frequency of the driving simulator logger.

4 DATA ANALYSIS AND RESULTS

After the creation of the datasets, as described in the previous Section, different models have been trained and compared. First, classifiers were trained for each participant with different characteristics and parameters for each algorithm, including different combinations of input variables and different values of mobile windows. This procedure has been adopted both for SVM and FFNN.

Second, we have carried out different comparisons. First of all, the performances of each model have been compared one other inside the data regarding the same subject ("within-subject" analysis) in order to assess how a specific model can fit a specific subject (very interesting for the personalisation issue). Then, we have considered a "between-subject" analysis:

- the model of a subject (training) has been used to classify the distraction (testing) of all other subjects, following a "leave-one-out" approach through the whole number of subjects available.

- the data from all subjects (randomly sampled) have been merged together and then a model has been trained on the whole dataset available.

This procedure allows checking how much a single Distraction model can be generalised to different users and to different scenarios or conditions.

4.1 Results of Distraction Classification for each Subject

For the first type of analysis we have compared two ML techniques: FFNN and SVM. We have considered the first one for the pattern classification of distraction. Different networks configurations and topologies have been analysed for each subject, with different window values of the data and different inputs. The chosen network has the following characteristic:

- training method = Scaled Conjugate Gradient Back propagation
- number of layers = 2 layers topology has been chosen, with one Hidden Layer (HL – very rare the case in which more than 2 are needed; in our case, two did not provide appreciable improvement of results) and one Output Layer (OL)
- transfer function = a Sigmoid transfer function has been used for both the HL and OL.

In the HL, different numbers of Hidden Neurons (HN) have been tested; in particular, the following table shows the results obtained for each subject:

Table 1. Performance of the FFNN Classifier for each subject.

Subject	Win	CR	HN	MSE
S1	3	84.5	50	0.113460
S2	2	79.8	50	0.149205
S3	2	79.8	10	0.137261
S4	5	82.7	50	0.119556
S5	3	84.4	20	0.121516
S6	2	89.7	50	0.079988

Mean Square Error (MSE) has been used to evaluate the performances and so as stop-criterion; in fact, training automatically stops when generalization stops improving, as indicated by an increase in the MSE of the validation samples (that, we remember, is the 15% of the dataset).

The Correct Rate (CR) of classification has been regarded as one meaningful parameter to assess the different models (both for FFNN and for SVM). As Table 1 shows, the best performance has been obtained for S6, with a CR equal to 89.7%. In this case, the training time was 151s. Hereafter, the ROC plot is showed:

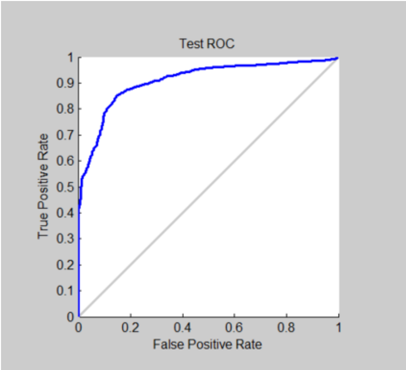

Figure 2 ROC curve of testing set for S6, with HN = 50 and Win = 2

With reference to the plot, the number of true positives (TP), of false positives (FP), of false negatives (FN) and the number of true negatives (TN) is the following:

- TP = 45.1%
- FP = 3.8%
- FN = 6.5%
- TN = 44.5%

It is worth to note here that mobile windows have given the best results with respect to raw measures, especially the window values of 2 steps. In most cases, HN = 50 has given the best performances (more HN caused a much longer training time, without a significant improvement in performances). Each configuration has been trained multiple times, since this procedure generates different results due to different initial conditions and sampling.

Second, we have considered the SVM, also in this case with different configurations, different windows and different inputs. In particular, several kernels have been tested, different values of parameters:

- Linear (LIN)
- Quadratic (QUAD)
- Polynomial (POLY)
- Radial Basis Function (RBF)
- Multi-Layer Perceptron (MLP).

The results are illustrated in the following table:

Table 2. Performance of the SVM Classifier for each subject.

Subject	Win	Kernel	Parameters	CR
S1	1	RBF	Sigma = 0.5	95.6
S2	2	POLY	Order = 3	95.2
S3	2	RBF	Sigma = 0.5	96.4
S4	2	RBF	Sigma = 0.5	94.2
S5	5	RBF	Sigma = 0.1	94.5
S6	5	RBF	Sigma = 0.5	93.4

As Table 2 shows, the RBF has proven to be the best Kernel function in 5 cases out 6; whose expression is:

$$K(x_i, x_j) = e^{-\sigma|x_i - x_j|^2} \qquad (1)$$

Where x_i and x_j represent the data points and σ is a predefined positive scaling factor parameter.

The RBF is a very robust kernel function, using which, it is possible to implement both nonlinear and linear mapping by manipulating the values of its parameters. Moreover, the RBF can reduce numerical difficulties and ends to obtain more robust results than other kernels, such as polynomial and linear ([18] and also confirmed by our results).

In this case, the best results were for S3, having the best CR, equal to 96.4%; the RBF Kernel has a σ parameter of 0.5, with a Win value of 2. In particular, the *sensitivity* (that is, Correctly Classified Positive Samples or True Positive Samples) is 98.9%, while the specificity (Correctly Classified Negative Samples or True Negative Samples) is 93.9%.

A comparison between the two models of Distraction (SVM and FFNN) gives the results that SVM outperforms FFNN. In fact, from table 1 and 2, the CR parameter is definitely better for SVM classifier than for FFNN classifier.

Finally, we have used MATLAB Neural Networks Toolbox for the FFNN model and the MATLAB Bio-informatics Toolbox for the SVM model.

4.2 Results of Distraction Classification for a "generic" Subject

Another research question we want to answer was about the generalization capability of the distraction classifier.

First of all, the model of a subject has been used to classify the distraction of the other subjects, following leave-one-out method.

Unfortunately, such an approach did not give the expected results with an unsatisfactory value of the CR parameter, both for FFNN (no more than 62%, on average) and for SVM (no more than 72% on average). If such algorithms are implemented for a real-time application on-board a vehicle, these results are strongly too poor, especially because the rate of TP was much lower than TN, implying that the model was not good enough to classify if the driver was actually distracted. One possible reason is that every subject responds in a very specific way to the secondary task (used to induce distraction during the test) and thus different subjects present different patterns on the data.

For this reason, we preferred to focus our attention on the second method we have used, in order to test the generalization capability of these models. Thereby, a new dataset has been created (always with the same percentage of training, checking and testing sets), taking randomly the data from all subjects, in order to create a more homogenous dataset; also in this case, the same different window values have been taken into account.

For the FFNN model, the following performances have been achieved:

Table 3. Performance of the FFNN Classifier for the whole dataset.

Win	CR	HN	MSE
1	74.9	50	0.165221
2	71.2	20	0.176795
3	73.1	10	0.169885
5	72.8	10	0.170707
5	72.8	50	0.172920

FFNN model has performed worst than in the previous analysis, focused on each subject. In fact, now the best result has been obtained with a CR = 74.9% and it is interesting to note without using mobile windows on the data.

For what concerns the results achieved using SVM model, the situation is the following:

Table 4. Performance of the SVM Classifier for the whole dataset.

Win	Kernel	Parameters	CR
1	RBF	Sigma = 0.5	84.3
2	RBF	Sigma = 0.3	81.8
3	RBF	Sigma = 0.5	80.5
5	RBF	Sigma = 0.3	82.1

Also in this case we obtain the best value with an RBF Kernel, using a σ parameter equal to 0.5. The CR of 84.3% is much better than 74.9% and thus SVM model outperforms FFNN model even in this case; however, SVM does not obtain the same results of each single subject. In particular, the sensitivity is 84.5% and the specificity is 84.1%.

Since the second best performance both for FFNN and SVM models is given by win = 5, we can guess that a deeper investigation, using wider windows, should be explored (see last Section for more details).

5 DISCUSSION

This strategy of developing methodology to detect driver's distraction, based on ML algorithms, is not completely new (see, for example, [12], [19] and [20]). In particular, the first two works develop real-time methods for distraction classification using Bayesian Networks and Support Vector Machines, respectively. The results obtained are comparable with ours (related to the whole dataset) with a rate of correct classification of around 80.1%. Our best case was 84.5, so – considering also the differences in the experiments, even if both carried out in a driving simulator – absolutely comparable. Our innovation with respect to these works regards mainly these two aspects:

- Different use of input features for the classifiers
- Personalization aspects

For the first point, all the aforementioned works used eye-tracker information, which is not so easy to obtain. In fact, when using the simulator, it is relatively easy to have it, but in a real-time application on the car, this is extremely difficult. Several limitations can be met. The first concerns the problem of integration: a dedicated camera and related ECU is needed and has to be integrated into the cockpit of the vehicle. Second, although the information provided by eye-tracker device are absolutely useful, nonetheless they require – for example – that the drivers do not wear sunglasses and glasses, or eye make-up, because these conditions may affect negatively tracking accuracy [12]. Moreover, there is the problem to obtain consistent and reliable sensor data. Eye trackers may lose tracking accuracy when vehicles are traveling on rough roads or when the lighting conditions are variable. Of course, the use of other physiological measures (such as heart rate or respiration rate, skin conductance, etc.) can provide other excellent indicators, but they are even more intrusive and difficult to use in real-time in the ordinary cars. In this context, our challenge was to provide a data-mining based method, which does not require the mandatory use of eye-tracker information (or other physiological measures) but it is based only on vehicle dynamic data. Despite the fact that for the whole dataset we did not reach the performance level of a single subject, we can say that such a goal has been achieved.

Secondly, this research has proved an excellent method to personalize the model; from one side, a "generic" distraction classifier is easier to be extensively applied and trained; however, on the other side, the performances obtained with the application of specific model for each driver are definitely good. Perhaps, this is a direction to take into account in the distraction classification field, since different drivers respond to external or internal stimuli – which are responsible of distraction – in very different manner, as our data proved..

Finally, with respect to our previous research [20], we have considered an actual classification problem and not a reaction time profile to reconstruct (which implies the use of "unknown" thresholds in post-processing, in order to understand if the driver was really distracted or not). In addition, the number of data points collected in this experiment, as well as the number of conditions and scenarios where the experiments have been performed, is much larger and thus more representative of ordinary use for such a system.

6 CONCLUSIONS

This paper developed a data mining based model, to detect driver distraction from driving dynamic data. We have explored two algorithms: FFNN and SVM, which has proved to constitute a viable means of detecting driver's distraction. We have considered a personalization aspect, with one specific model for each subject and one generalization aspect, where SVM has obtained results comparable with those present in literature, but without using eye movements as inputs to the classifier.

All in all, some limitations of our work are discussed hereafter. First, we need to improve the generalization capability of the classifiers. So, future researches imply the assessment of new ML techniques, like Adaptive Network-based Fuzzy Inference systems (ANFIS), Relevance Vector Machines (RVM) or Hidden Markov Models (HMM). Moreover, different types of Kernels can be explored, such as Gaussian Processes, a relatively recent kernel-based method.

Second, we have to consider the training time and the delay of distraction detection system, to evaluate whether it is appropriate for the application. The training time for the best performances of FFNN model took around 151s on the whole dataset and 35.6s on the dataset of subject 6 (the one with the best results), while the training time of SVM models took 20s and 5s (for subject 3), respectively. Although SVM are faster, nonetheless also response time is crucial and under this viewpoint, FFNN are usually better. So, the delay coming from data-reduction and response time of the models, have to be evaluated. Concerning the delay from summarizing data across windows, one of the possible future activities is to consider larger values (within the upper limit of 2s, anyway, since it is the limit for detecting the distraction, see Section 2). However, larger windows may cause longer delay and thus such lags shall be quantified precisely, with specific and dedicated procedure to do it. The consequence of these lags will depend on the particular distraction mitigation strategy they support. So, the development of real-time classification system for distraction has to balance the cost of time lags with the precision of distraction estimates for a particular mitigation strategies and this represents an important research issue.

Finally, we need to perform tests directly on road data in a more diverse set of conditions and scenarios; this is a fundamental step in order to assess the generality of the results.

This research has been carried out and it is going on within the European co-funded project ISI-PADAS (http://www.isi-padas.eu). Authors would like to thank the whole consortium.

7 REFERENCES

[1] Treat J., et al.; Tri-level Study of the Causes of Traffic Accidents: Final Report, volume 1. Technical Report Federal Highway Administration, US DOT (1979)

[2] Beirness D.J. et al.: The Road Safety Monitor: Driver Distraction. Traffic Injury Research Foundation. Ontario, Canada (2002)

[3] Sussman E. D., Bishop H., Madnick B., and Walter R., "Driver inattention and highway safety," *Transp. Res. Rec.*, no. 1047, pp. 40–48, 1985

[4] Wang J., Knipling R. R., and M. J. Goodman, "The role of driver inattention in crashes; New statistics from the 1995 crashworthiness data system (CDS)," in *Proc. 40th Annu.: Assoc. Advancement Automotive Med.*, 1996, pp. 377–392

[5] Stutts J. C., Reinfurt D. W., Staplin L., and Rodgman E. A., "The role of driver distraction in traffic crashes," *AAA Foundation of Traffic Safety*, 2001

[6] Haigney D. and Westerman S. J., "Mobile (cellular) phone use and driving: A critical review of research methodology," *Ergonom.*, vol. 44, no. 2, pp. 132–143, Feb. 2001

[7] Tan, P.-N. *Introduction to Data Mining*. Pearson Addison Wesley, Boston, 2005

[8] Baldi, P. and Brunak S.. *Bioinformatics: The Machine Learning Approach*, 2nd edition ed. MIT Press, 2001

[9] Mattes, S. (2003) The lane change task as a tool for driver distraction evaluation. In *H.* Strasser, H. Rausch & H. Bubb (Eds.), *Quality of work and products in enterprises of the future* (pp. 57-60). Stuttgart: Ergonomia Verlag.

[10] Wickens, C.D. (2002) Multiple Resources and performance prediction. Theoretical Issues In Ergonomics Science, 3(2), 159-177

[11] Treat J.R.: A study of the Pre-crash Factors involved in Traffic Accidents. In: The HSRI Review, *10*(1), 1–35 (1980)

[12] Liang et al.; "Non-intrusive Detection of driver Cognitive Distraction in real-time using Bayesian Networks; TRB 2007 Annual Meeting

[13] Young K. Regan M. and Hammer M. Driver Distractionç a review of the Literature. Technical Report No 206, Monash University, accident Research Center, November 2003

[14] Vapnik V. N., *The Nature of Statistical Learning Theory*. New York: Springer-Verlag, 1995

[15] Cristianini N. and Taylor J. S., *An Introduction to Support Vector Machines and Other Kernel-Based Learning Methods*. Cambridge, U.K.: Cambridge Univ. Press, 2000

[16] Byun H. and Lee S. W., "Applications of Support Vector Machines for pattern recognition: A survey," in *Proc. 1st Int. Workshop, SVM Pattern Recog. With Support Vector Mach.*, Niagara Falls, ON, Canada, 2002, pp. 213–236

[17] Haykin S.: Neural Networks: a comprehensive Foundation. Prentice Hall (1999)

[18] Zhang Y., Owechko Y., and Zhang J. Driver cognitive workload estimation: A data-driven perspective. In *Proc. IEEE Intell. Transp. Syst. Conf.*, Washington, DC, 2004, pp. 642–647

[19] Liang Y., Reyes M.L. and Lee J.D.: Real-time Detection of Driver Cognitive Distraction Using Support Vector Machines. In *Proc. IEEE Transactions on Intelligent Transportation Systems*, Vol. 8, No. 2 – June 2007

[20] Tango F., Botta M. Evaluation of Distraction in a Driver-Vehicle-Environment Framework: an application of different Data-mining techniques. In Proc. 9th Industrial Conference on Data Mining (ICDM09). Leipzig, Germany, 2009. Springer-Verlag Eds.