Creating Cognitive Models from Activity Analysis: A Knowledge Engineering Approach to Car Driver Modeling

Olivier L. Georgeon, Matthias Henning, Thierry Bellet, Alain Mille

Creating Cognitive Models from Activity Analysis: A Knowledge Engineering Approach to Car Driver Modeling

Olivier Georgeon (olivier.georgeon@inrets.fr)
Institut National de Recherche sur les Transports et leur Sécurité,
25 avenue François Mitterrand, 69675 Bron, France.

Matthias J. Henning (matthias.henning@phil.tu-chemnitz.de)
Chemnitz University of Technology, Psychological Institute, 09107 Chemnitz, Germany.

Thierry Bellet (thiery.bellet@inrets.fr)
Institut National de Recherche sur les Transports et leur Sécurité.

Alain Mille (alain.mille@liris.cnrs.fr)
Université Claude Bernard Lyon 1 - LIRIS, 8, Boulevard Niels Bohr, 69622 Villeurbanne, France.

Abstract

A cognitive modeling approach coming from ergonomics is presented. It is based on an activity analysis that starts from a "trace of activity" made up of data recorded with sensors. Knowledge engineering software was developed to transform this trace and map it with explicative concepts. These concepts are defined by cognitive ergonomists to explain the activity by the mental states and processes of the operator. These concepts are produced on a pragmatic and evolutionist basis with the interactive help of the software, and are confronted with the operator's own assessment of his subjectivity. The approach is illustrated by its application for car driver cognitive modeling.

Introduction

The increase of professional activities having a "mental dimension" has encouraged the development of a cognitive ergonomics for several decades. Wilson and Corlett (2005) defined ergonomics as the theoretical and fundamental understanding of human behavior and performance in purposeful interacting systems, and its application to design interactions in the context of real settings. More specifically, cognitive ergonomics rests overall on the statement that this kind of activity cannot be understood without any reference to the operator's "subjectivity", including perceptions, intentions, feelings, expectations, knowledge, mental representation, etc. The main investigating method for completing this type of understanding is "activity analysis". It starts from an observation of the activity in a situation as natural as possible, then it consists of an analysis for creating models of activity, and eventually cognitive models.

Our work focuses on the first half of Wilson's definition: understanding for creating models that could explain and predict the operator's behavior. However, we take the term understanding in a pragmatic sense, which considers any meaning as grounded by a final purpose. We present here a methodology and a software tool to help ergonomists build cognitive models of an operator involved in an activity. We illustrate it by the creation of models of the car driver, within the global purpose of assessing or designing intelligent driving assistance systems.

Cognitive Modeling of the Car Driver

Most of the cognitive user models come from Human Computer Interaction studies. As Salvucci and Lee (2003) introduced it, they have typically addressed behavior on one of two levels of abstraction. The higher level modeling frameworks, e.g. GOMS, represents behavior as basic user actions, such as moving a mouse or pressing a key. They mostly represent the user's knowledge and representations under complex data structures such as "scripts" or "frames" (Minsky, 1975). The lower level frameworks, e.g. ACT-R or EPIC, describe "atomic components" of behavior with rules triggering over cognitive steps of roughly 50 milliseconds. Here the user's knowledge is represented under declarative "chunks" and procedural "production rules".

There have been many cognitive models of the car driver at both of these levels. At the higher level, Bellet and Tattegrain-Veste (1999) implemented Cosmodrive, a specific modeling framework of the car driver. At the lower level, ACT-R was used by Salvucci, Boer, and Liu (2001) to predict behavior during lane changes on a motorway, or by Ritter, Van Rooy, St. Amant, and Simpson (2006) to model the driver of a driving simulator.

The modeling task is facilitated by these modeling frameworks that increasingly embed general scientific knowledge about human cognition. The existence of these frameworks thus raises the question of how to feed them with a realistic coding of the specific cognitive activity which is modeled. The specification of these "frames", "scripts", "chunks" and "production rules", for creating a specific model is indeed more than a simple programming task. It is a task of modeling the activity from a cognitive point of view. It involves an expertise about the activity itself, and also about how the operator builds and processes the information needed to complete it. This expertise is acquired by ergonomists through their professional practice. What interests us is to support the development of this expertise, formalize it and capitalize it in a knowledge engineering tool for activity analysis and modeling.

Explaining Behavior by Mental Activity

A car passenger who sees something dangerous and that the driver does not react to could obviously infer that the driver is not aware of the danger and warn him. Inferring others' mental state is something we are always doing in our everyday life. This fact of granting other people with having a subjectivity can be taken in our approach either as an ethic injunction, or as a pragmatic viewpoint: the intentional stance (Dennett, 1987). Our purpose is to make this assessment of the subjectivity more rational, from the observation of behavior and situation as it can be recorded by sensors.

We rely on concepts of cognitive psychology such as "mental representation", or "Situation Awareness" (Endsley, 1995). The latter simultaneously covers the perception, the comprehension and the anticipation for describing how the elements of the situation are "mentally" taken into account by the operators. However, if these concepts seem clear for explaining behavior which results from a symbolic reasoning, it becomes less clear when considering an activity in which the subject is physically involved. Here, their subjective experience "accompanies" their actions and can be seen as much as a cause than as a consequence of their activity. In this case, behavior can be explained by "operational schemas", "implicit knowledge" or automatic processing (Schneider & Shiffrin, 1977). Finally the relation between different levels of control can be studied: skills, rules, and knowledge: (Rasmussen, 1983), or implicit versus explicit control.

The Objectivity of Mental Activity

There could be a vicious circle in explaining the operators' behavior by their mental activity that is itself inferred from their behavior. This raises the question of how to take other's subjectivity as an objective reality, which can be scientifically studied. Psychologists have been discussing this issue since the beginning of psychology. In this study we retain three elements of response:

(a) A pragmatic epistemology should be adopted, where the concepts created for explaining activity make sense in virtue of their usage (Wittgenstein, 1953). The approach should support a short testing of the usability of the concepts and an interactive negotiation of their meaning.

(b) The approach should also conform to an evolutionist epistemology (Popper, 1972). An explanation cannot be proved to be true in an absolute sense, but can be retained as long as it has not been falsified. The approach should facilitate the production of explanations and their possible falsification in a short loop.

(c) The explanations should be confronted with the subjective assessment made by the subjects themselves. These assessments should be reasonably trusted and analyzed as formal input data (Ericsson & Simon, 1993).

To address these requirements, we need some facilities to easily define and manipulate explicative concepts. The data collected from a recording of the activity should be easily linked to these concepts.

Methodology and Tool

This issue of making sense of collected data addresses the research area of "knowledge discovery" as defined by Fayyad (1996): the overall process of discovering potentially useful and previously unknown information or knowledge from a database. In our case, the data is the recording of the activity collected by sensors, and the expected knowledge consists of models explaining the activity in terms of the cognition of the subject. To Fayyad, this problem is one of mapping low-level data into other forms that might be more compact, more abstract, or more useful. A knowledge discovery system should enable a user to drive a cyclical process of abstracting and understanding data. The knowledge is finally built "in the analyst's mind" with the interactive use of the system. Progressively, with this interaction, the knowledge is however capitalized in the software under the form of a step by step improvement of the abstraction and of the visualization facilities.

To Miles and Huberman (1994), analysis involves a classification of datum that could be theory-driven, data-driven (evolving a "grounded theory"), or a compromise between both, which he calls "ontological coding". The latter is what we are proposing here, to define people-centered categories such as context, intention, state, or patterns of relationship between elements.

A Trace-Based System

The specificity of our data is that it is sequential, i.e. every item of data is associated to a time-stamp and related to a particular moment of the activity. Sanderson and Fisher (1994) have set the bases of what they called Exploratory Sequential Data Analysis (ESDA) in this founding paper. They define it as an exploratory process that aims at "Looking at data to see what it seems to say". Hilbert and Redmiles (2000) propose a survey of such tools dedicated to the analysis of human computer interaction. Namely, MacShapa (Sanderson, McNeese & Zaff, 1994) can be highlighted for its covering facilities of visualization, searching, filtering, abstracting and computing statistics. Hawk (Gusdial, Santos & Badre, 1994) is also noticeable for its facilities of abstraction based on a specific programming language. But these tools are dedicated to usage modeling and not to cognitive modeling. The need remains for facilities which could help model the mapping between low level events and higher level explicative concepts, and help investigate the meaning of these concepts.

We suggest addressing this need by modeling the sequential data as a "trace", as defined by Laflaquière, Sofiane-Settouti, Prié, and Mille (2006). For them, a trace is a graph where nodes are facts or events and arcs are relations between them. A trace is associated with a "trace model” that consists of an ontology where the semantic of facts, events, and relations is defined. On this basis, transformation and mapping of traces can be modeled and performed by what they call a “Trace-Based System”, which is a Knowledge-Based System dedicated to traces.

The ABSTRACT Software Tool

Our implementation of a Trace-Based System for cognitive activity analysis is named ABSTRACT (Analysis of Behavior and Situation for menTal Representation Assessment and Cognitive acTivity modeling). It is described from a computer science point of view in (Georgeon, Mille & Bellet, 2006). ABSTRACT relies on several basic standards and tools to represent and manipulate the traces. The traces themselves are encoded under the RDF format (Resource Description Framework), a standard for encoding graphs. The trace models are encoded under the RDFS format (Resource Description Framework Schema), a standard for encoding ontologies. The inference rules for abstracting the traces are written in SPARQL, a graph query language for RDF. The graphical visualization is made in the SVG format (Scalable Vector Graphic). It is a vector graphic standard which enables interaction.

The most challenging issue concerns interactivity, since the whole process rests on the identification, the understanding and the definition of patterns of interest by the ergonomist. We address this issue by providing facilities to easily browse and transform the visual plots, define new explicative symbols and specify inference rules for producing new instances of these symbols. So far, the specification of these inference rules requires the ergonomist to have a basic understanding of the SPARQL langage. He creates them in a semi-graphic way, i.e. he interactively produces skeletons of queries, and then he manually completes them. We are still working on improving the functionality of creating these queries towards a full graphical interface.

The Modeling Process with ABSTRACT

The Experimental Data

In our example, the data is collected with an instrumented car during driving experimentations on an open road. Figure 1 shows an example of the video recording. The top left image is the video output of the obstacle detection system, the top right is the front view, the lower left - upper part is the rear view, the lower left - bottom part is the lane view, and the lower right is the video output of the eye tracker. In addition, data is collected from sensors: speed, steering angle, pedal use, distance ahead, obstacle detection from telemeters, and GPS positioning. Situations of interest are marked by the experimenter in the car by pressing a button. Subjective evaluation is obtained from the driver during a post-experiment “self-confrontation” interview. The driver reviews the video with the ergonomist and is asked to assess specific situations by placing cursors on linear scales for each criteria: difficult, critical, dangerous, stressful, responsibility, surprise, fear, and performance.

The experimentation collects a set of time-stamped data. Its choice and its form is not neutral; it is based on the initial knowledge of the activity, and on the research goals. This choice constitutes a form of “anticipatory data reduction”. This data is also possibly pre-processed with algorithms of noise reduction or of sensor calibration.

11055	108	3.28	Eye_Left_Mirror	duration	0.184009	
11056	108	3.28	Brake	subtype	Min	valu
11057	108	1.04	Brake	subtype	Threshold_Down	valu
11058	108	0.88	Eye_Ahead	duration	0.284013	
11059	108	-0.48	Speed	subtype	Max_Variation	valu
11060	108.28	-1.44	Speed	subtype	Min_Variation	valu
11061	108.56	-2.96	Eye_Left_Mirror	duration	0.117006	
11062	108.73	-4.48	Indicator	subtype	Off	valu
11063	108.94	-4.48	Eye_Ahead	duration	0.334016	
11064	109.36	-4.96	Steering	subtype	Threshold_Up	valu

Figure 1: Confronting the collected trace to the video.

The Collected Trace

The collected trace is the first level of trace in ABSTRACT. It is obtained by converting the collected data into a succession of low level events precisely defined in an ontology. For instance, Figure 2 shows how points of interest of the analogical curves are extracted and merged into the trace: Thresholds, local Minimums and Maximums, inflection points, etc.

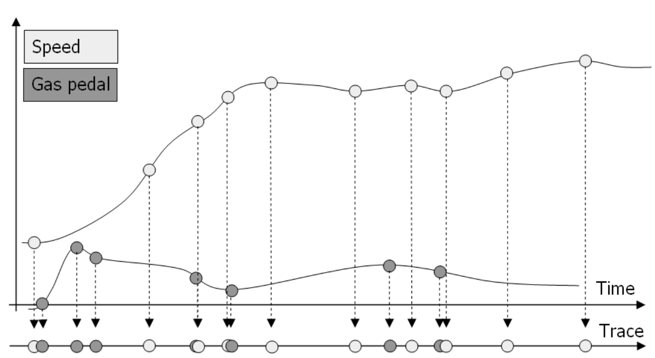

Figure 2: Discretization of the analogical data.

The line at the bottom of the figure represents the collected trace. The circles represent the events from different sources. Numerical or textual properties are attached to these events as needed: their time-code, their source, their type, their value, their duration, the variation rate of their value at this point, etc. This collected trace requires a validation by the ergonomist to ensure that these events can respond to his scientific issues. The adjustment of the various parameters used for producing these events is a delicate but important step. This validation is made with the help of a facility provided by ABSTRACT for playing the video in synchronization with the trace (Figure 1). The collected trace is represented in a spreadsheet with a line for each event and their properties in columns. A color code makes it easier to read. The spreadsheet is automatically scrolled up or down to follow the video when it is played forwards or backwards. The first event of Figure 1 represents a triggering event of the left mirror area of the eye tracker. It lasts 184 ms and happened at a speed of 108 km/h with a steering wheel angle of 3.28°. The last line of the figure is a threshold crossing event upwards of the steering wheel angle. The threshold value of 5° was chosen after different trials and checks because it proved to be a meaningful/useful threshold for studying lane changes on motorways. The video is not entirely encoded into symbols and remains exploited by the ergonomist in parallel with the symbolic traces.

The Analyzed Traces

The collected trace is then enriched by more abstract symbols to produce traces as shown in Figures 4 and 5. In these displays, the time goes from left to right and the "x" position of the events is given by their time-code. The ergonomist is free to define the shapes of symbols, their colors and their "y" position in the display. He can click on symbols to see their properties. These figures use a display model where the lower level symbols are the dots at the bottom. The middle level symbols are placed around the middle horizontal axis. What concerns the left of the vehicle is placed above this axis and what concerns the right is placed below it. Triangles oriented to the right concern something "frontward" (e.g. "Look ahead"), triangles oriented to the left concern something "backward" (e.g. "Left mirror glance"). The lines are the relations of inference from lower level symbols to higher level symbols. The upper part of the display is used for the highest level symbols which are not concerned by this left/right rule (e.g. "Decision"). The colors in the real display are used to indicate the type of data (i.e., blinker in orange), it makes it much easier to read than in print. When applied, an inference rule adds a new instance of a symbol in the trace everywhere the specified pattern is found. Filters can mask the undesired symbols.

The Ontology

Parallel to the creation of the inference rules, the ergonomist defines the class of the symbols in an ontology. ABSTRACT provides them with an access to the "Protégé" ontology editor to let them define the semantic and visual properties of each class of symbol. This ontology is exploited by the SPARQL inference engine and by the display functions.

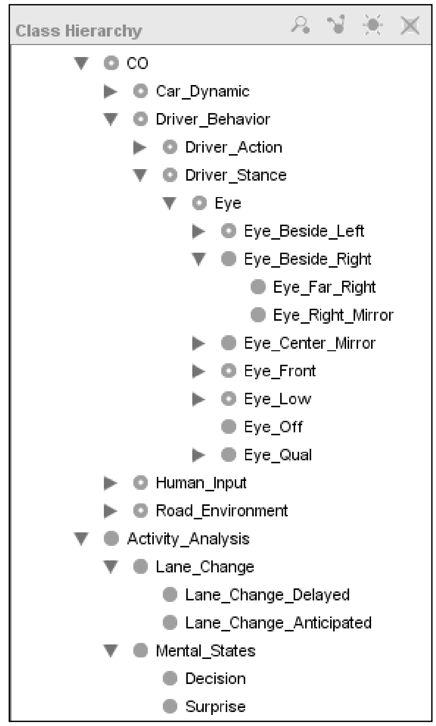

Figure 3: The ontology edited with Protégé.

Figure 3 shows an extract of the top classes of the current ontology. The CO class defines all the Collected Objects existing in the collected trace. One branch of the eye tracker data is expanded to show how the area of interest can be organized. Heritage is supported so that inference rules and visualization settings can be defined at any level of the hierarchy.

The Activity_Analysis class concerns concepts that are inferred from CO. For instance, two types of lane changes could be modeled and can be recognized from specific patterns in the data: lane-change-anticipated and lane-change-delayed, published in (Henning, Georgeon & Krems, 2007). All these concepts are defined on a pragmatic and evolutionist basis as said above. They come from a trade-off between, on one hand, the possibility of inferring them from the data and on the other hand from the ergonomist's quest for explanation. Moreover they can be searched in collaboration between the ergonomist and the subject himself during the self-confrontation interview.

Examples of Modeling

Figure 4 shows a typical example of a lane-change-delayed schema. In this figure, the "Start-thinking" symbol corresponds to the moment when the driver declares that he starts considering changing lane in the self-confrontation interview. The "Button" is a signal from the experimenter recorded during the course for indexing every lane change, and the "Lane-crossing" is the moment where the left front wheel crosses the lane, manually encoded from the video. All other symbols are automatically inferred from the sensor data.

In this kind of situation the driver is impeded by a vehicle slower than his desired speed. He may check his left mirror several times, then, when he decides to overtake, he accelerates while checking his mirror; he switches his blinker on, starts steering, and crosses the line. In this situation, the decision to perform the lane change can be inferred from the conjunction of both the acceleration and the left mirror glance within a certain lap of time ("Decision" symbol). This symbol can be useful as a predictor of the lane change in this situation, we call it "Decision" within the explicative framework of "operational schemas". The inference is made by a query expressing: "add a new node of type "decision", linked through edges of type "inferred", to pairs of nodes matching the condition: one is of type "accelerate", other is of type "Left mirror glance" and their "time-stamp" properties are comprised within one second. Once defined, this query can be applied to the whole trace for statistical assessment of true detection and false alarm. This driver made 11 lane changes, 5 could be categorized as "delayed" and 2 as "anticipated". The decision is detected before the blinker in the 5 lane-change-delayed, and there is one false detection.

Another example of analysis is shown in figure 5. It is a case study where the driver attempts to change lane but did not see a car arriving from the rear on the left lane. He initiated the maneuver but aborted it at the last moment. The figure shows that the driver did not look at his rear mirror: he was looking at his speedometer instead. He stabilized his speed at a higher value than the car ahead. The "Distance ahead" event at a relative speed of 3.5km/h comes from the front telemeter. From these two pieces of information we can already make the diagnosis that he was intending to overtake. Then he looked at his left mirror almost at the same time as switching his indicator on. At this moment he saw the car coming on the left lane from behind and he aborted the lane change. He made an abrupt steering back to the right, switched off his indicator, and pressed the brake pedal to avoid bumping into the car he was intended to overtake. Finally he started looking again at his left mirror and accelerated again to perform the lane change when the other car had passed him.

The subjective evaluation from the interview indicates that he was very surprised: (90% of the scale), stressed (75%) but the difficulty was not so high: 30%.

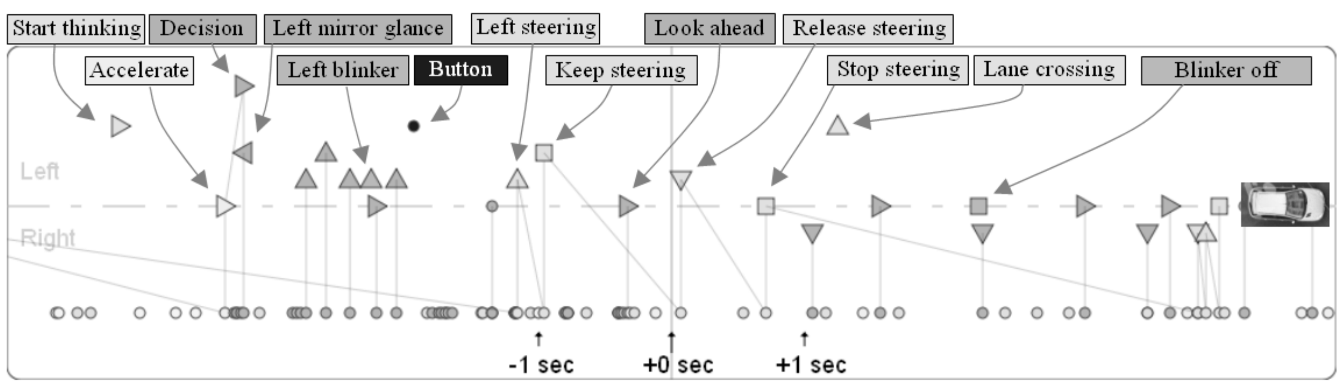

Figure 4: Prototype of a "lane-change-delayed" schema.

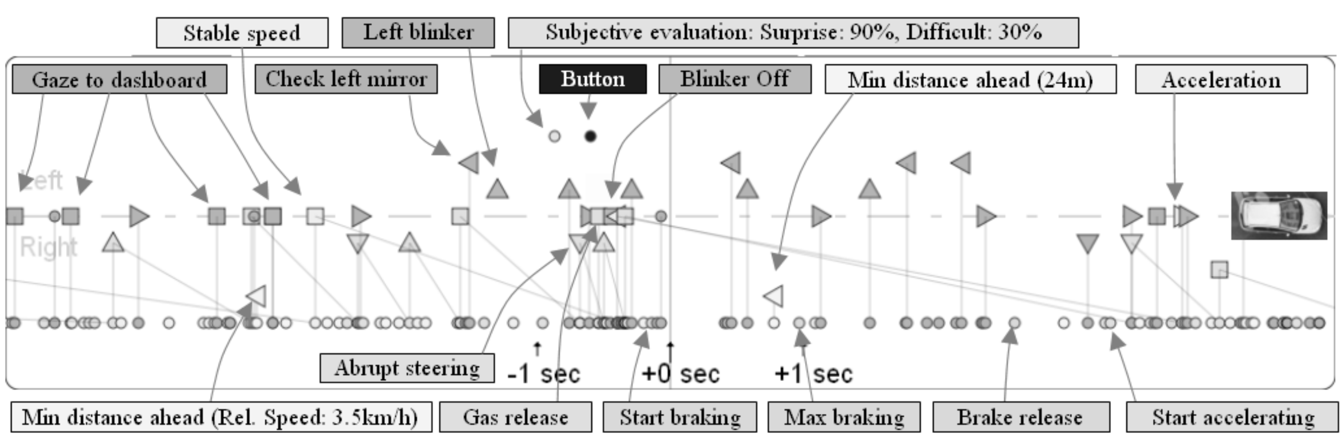

Figure 5: Lane change attempt with mistake of situation awareness.

This example shows how we can explain a mistake by an error of "situation awareness". It also let us propose a threshold value of the variation rate of the steering wheel that can indicate the subjective surprise of the driver. Finally, it shows that more than one second before this mistake, we could automatically make the diagnosis that the driver was intending to make an error from the objective data. Based on this diagnosis, a driving assistance system could warn him.

Conclusion

A methodology and tool were presented to support the activity analysis and the cognitive modeling of an operator. They are intended to formalize the process of explaining activity by the mental states and processes of the operator. The tool can capitalize on the expertise of the ergonomist through inference rules, ontologies, and visualization settings. The resulting models consist of explicative concepts defined in the ontology and of abstract traces describing patterns of behavior with these concepts. The modeling process lets categories emerge from situations in parallel with their description and provides means to validate them by statistics. The analysis is made from data collected in the field of the studied activity. It consists of a step-by-step process of abstraction driven by the ergonomist in interaction with a software tool. It is shown how this analysis can help design assistance systems for dealing with specific situations.

The obtained models could also be exploited for programming simulations in cognitive simulation frameworks, which would confront them to theoretical constraints on human cognition.

Acknowledgments

This project was supported by the Humanist Network of Excellence funded by the European Commission. We appreciate the assistance of B. Mathern for software implementation and of F. Ritter for his useful comments on this paper.

References

Bellet, T. & Tattegrain-Veste, H. (1999). A framework for representing driving knowledge. *International journal of cognitive ergonomics*, 3(1), 37-49.

Dennett, D. C. (1987). *The intentional stance*. Cambridge, MA: MIT Press.

Endsley, M. R. (1995). Toward a theory of situation awareness in dynamic systems. *Human Factors*, 37(1), 32-64.

Ericsson, K. A. & Simon, H. A. (1993). *Protocol analysis: Verbal reports as data*. Cambridge, MA: MIT Press.

Fayyad, U. (1996). From Data Mining to Knowledge Discovery in Database. *AI Magazine*, 17(3), 37-54.

Georgeon, O., Mille, A. & Bellet, T. (2006). Analyzing behavioral data for refining cognitive models of operator. *Philosophies and Methodologies for Knowledge Discovery, Seventeenth international Workshop on Database and Expert Systems Applications* (pp. 588-592). Krakow, Poland: IEEE Computer Society Press.

Gusdial, M., Santos, P. & Badre, A. (1994). Analyzing and visualizing log files: A computational science of usability. *HCI Consortium Workshop*.

Henning, M. J., Georgeon, O. & Krems, J. (2007). The quality of behavioral and environmental indicators used to infer the intention to change lanes. *4th International Driving Symposium on Human Factors in Driver Assessment*. Stevenson, Washington USA.

Hilbert, D. M. & Redmiles, D. F. (2000). Extracting Usability Information from User Interface Events. *ACM Computing Surveys (CSUR)*, 32(4), 384–421.

Laflaquière, J., Sofiane Settouti, L., Prié, Y. & Mille, A. (2006). A trace-based System Framework for Experience Management and Engineering. *Second International Workshop on Experience Management and Engineering (EME 2006)*. Bournemouth, UK.

Miles, M. B. & Huberman, A. M. (1994). *Qualitative data analysis: An expanded sourcebook*. London: Sage Publications.

Minsky, M. (1975). A Framework for Representing Knowledge. The Psychology of Computer Vision. P. H. Winston. New York: Mc Graw-Hill.

Popper, K. (1972). *Objective Knowledge*: Oxford University Press.

Rasmussen, J. (1983). Skills, Rules and knowledge; Signals, Signs and Symbols and other Distinctions in Human Performance Models. *IEEE Transaction on Systems, Man and Cybernetics*, 13(3), 257-266.

Ritter, F. E., Van Rooy, D., Saint Amant, R. & Simpson, K. (2006). Providing user models direct access to interfaces: An exploratory study of a simple interface with implications for HRI and HCI. *IEEE transactions on systems, man, and cybernetics, part a: Systems and Humans*, 36(3), 592-601.

Salvucci, D. D., Boer, E. R. & Liu, A. (2001). Toward an integrated model of driver behavior in a cognitive architecture. *Transportation Research Record*, 1779, 9-16.

Salvucci, D. D. & Lee, F. J. (2003). Simple Cognitive Modeling in a Complex Cognitive Architecture. *Human Factors in Computing Systems*. New York: ACM Press.

Sanderson, P. M. & Fisher, C. A. (1994). Exploratory sequential data analysis: Foundations. *Human -Computer Interaction*, 9, 251-317.

Sanderson, P. M., McNeese, M. D. & Zaff, B. S. (1994). Handling complex real-word data with two cognitive engineering tools: COGENT and MacSHAPA. *Behavior Research Methods, Instruments, and Computers*, 26, 117-124.

Schneider, W. & Shiffrin, R. M. (1977). Controlled and automatic human information processing: I. Detection, search and attention. *Psychological Review*, 84, pp 1-66.

Wilson, J. R. & Corlett, E. N. (2005). *Evaluation of Human Work*: CRC Press.

Wittgenstein, L. (1953). *Philosophical Investigations*.