

Master of Science Thesis

Decision Support System for Image Analysts of the Royal Armed Forces of the Netherlands

by

Marien Spek

June 1, 2015



Dept. of Information and Computing Sciences
Utrecht University
Utrecht, the Netherlands

Supervisor:
prof. dr. J.-J.Ch. Meyer

Dept. of Perceptual and Cognitive Systems
Netherlands Organisation for Applied
Scientific Research (TNO)
Utrecht, the Netherlands

Supervisor:
dr. P.-P. van Maanen

Abstract

One of the tasks of a human image analyst of the Royal Armed Forces of the Netherlands is to assess the threat level of vehicles. Technical improvements provide means to cover larger areas in more detail resulting in higher workload for the human analysts. In order to circumvent cognitive lock up or overload, supporting the human analyst in their task is a necessity.

This study presents a reasoning system which can evaluate whether a vehicle poses a threat given a certain situation in order to support the human analyst.

Multiple models were trained, compared and reviewed in order to investigate which elements are useful for such a reasoning system. All models were trained using participant test data that were gathered from average Dutch civilians performing a simplified version of the image analyst task.

The results showed that the amount of predictors, the combination of predictors and training of the models are important elements for the reasoning system in order to properly assess threats. However, future research should indicate whether these models perform well enough in real world situations.

Contents

1	Introduction	3
2	Background	4
2.1	Human Machine Interaction	4
2.2	Situation Awareness	4
2.3	Decision Support Systems	6
3	Model	6
3.1	Bayesian Belief Networks	7
3.1.1	Structure and parameters of a Bayesian Network	7
4	Hypotheses	9
5	Method	9
5.1	Participants	9
5.2	Materials	10
5.2.1	Traffic Simulation	10
5.2.2	Training data	13
5.2.3	DSS models	13
5.3	Statistical analysis	17
6	Results	18
6.1	Differences in between the trained DSS models	18
6.2	Differences between the trained and fixed DSS models	20
7	Discussion	22
8	Conclusions	24
9	Future work	24
A	Participant Booklet	28

1 Introduction

The computer industry changed society over the last decades [9, 16]. Mobile phones wiped the phone booths off the streets, cameras have made most of the security personnel unnecessary and drones are taking over from pilots. All these technological changes made it possible to automate tasks and processes that used to be done by humans, leaving humans to the role of supervising the automated systems. As the technology improves, more complex processes are automated, resulting in a shift of focus on other problems in Human-Machine Interaction (HMI) [14].

An example is the security of a mall. In the past a team of security personnel was supervising the mall which were present on the shopping floors providing a safe shopping environment. Nowadays, malls are covered with cameras which are overseen by the security personnel. This has led to a reduction of personnel and more importantly represents a change in work activities for the personnel. Although the cameras provide better image coverage of the mall, the human supervisor can only focus on a small portion of them at any given time. Therefore, the supervisor could easily miss suspicious or criminal behavior when looking at feeds on which nothing of interest happens.

In small settings as a the security of a mall this might lead to minor problems. In a larger setting, for instance, monitoring a top down camera feed of ten square kilometers of a city in order to provide secure passage of a convoy to it's destination (see figure 2), missing information could have severe consequences.

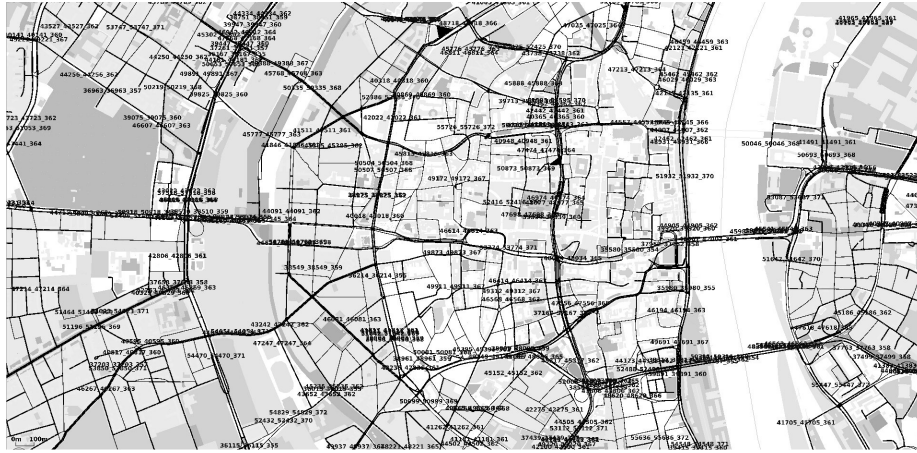


Fig. 2: City overview Cologne, Germany [17]

In order to prevent dangerous situations from occurring in large and dynamic environments, supporting the human supervisor with automated computer systems is a necessity [14]. These systems are also called Decision Support Systems (DSS). A DSS is an interactive computer based system that helps decision-making by using data and models to solve the provided ill-structured, unstructured or semi-structured problems [9].

In this study we will present a reasoning system to assess possible vehicle threats given a top down view of an urban area in order to support image analysts of the Royal Armed Forces of the Netherlands in their task. Although

this study is only concerned in creating a reasoning system for a DSS, several aspects of Human Machine Interaction are taken into consideration in order to prepare the reasoning system to be used in a DSS. The DSS models will be tested on participant test data, the results will be reviewed in order to answer the main question of this study; which elements are useful for a DSS to assess possible threats in traffic situations?

2 Background

2.1 Human Machine Interaction

Human Machine Interaction (HMI) is a field in computer science which focuses on the interaction between humans and computers. It especially concerns designing an intuitive interface for the human to interact with the underlying computer system [14]. HMI plays an important role in creating a DSS since HMI aims to present meaningful information to the human supervisor in a most intuitive and comprehensible manner. It also facilitates frameworks for the human supervisor to intuitively respond to the presented information, leading to the least possible overhead. [14]

For example, Arciszewski and colleagues described a system in which the processed information is displayed simultaneously with the current state of the supervisor [1]. The study of Hancock [6] pointed out that using an automated system in planes leads to fewer errors than only human controlled systems. It was shown that the full automation condition had the fewest errors overall however, the partial automation condition had the highest percentage of correct direct responses. Though both statements from the study of Hancock may seem contradictory, they are not since an incorrect direct response might not necessarily lead to an error and can be corrected with another response. It was also found that the partial automation condition had a faster reaction time, therefore it is recommended to always have a human supervisor present. These findings are consistent with findings of other studies in favor of HMI [14, 1, 6].

Although this study will only present a reasoning system, the DSS models that were created were designed with HMI in mind.

2.2 Situation Awareness

Situation Awareness (SA) is a crucial part of the human decision making process [3] and therefore also in Decision Support Systems (DSS), or also called Situation Awareness Support Systems (SASS) [3]. It is important to have a good understanding of how SA works in order to create a useful DSS as they are based on the same principles [3, 18]. SA is being aware of what is happening around you and understanding what that means, in the present and in the near future. This awareness is usually defined in terms of what information is important for a particular goal [18]. The concept of SA is most often applied to operational situations, where people must have SA for a specific reason such as driving a car, treating a patient or separating air traffic as an air traffic controller. Only those pieces of the situation relevant for the task at hand are important for SA. While the pilot of an aircraft must be aware of other planes, the weather and approaching terrain changes he does not need to know what the copilot had for breakfast [4, 13]. Each choice a human being makes is based on their currently

known information and their personal options. This also applies to a DSS. In order to provide useful decision support to a human supervisor, a DSS would have to assess the current state of the situation and how actions will effect this state in the future.

The formal definition of SA states that SA is more than the awareness of specific details in the environment. Therefore SA is divided into three levels [3].

The first level is the perception of elements in the environment. This concerns all available information which can be seen, heard, tasted, etc. Based on this perceived information, the following levels of SA will be passed.

The second level is the comprehension of the current situation. This involves interpreting and evaluating the incoming information, for example the detection that certain elements belong together. In this level it is important that the processor of the information has an understanding of the current context and a global comprehension of the world it is in.

The third level is the projection of future states. For this to work one should have knowledge about the (wanted) outcomes of possible actions to make the correct decision.

The higher levels of SA (levels two and three) are found to be critical for decision making in complex environments [3]. However, in all levels of SA errors do occur. Endsley [3] came up with a taxonomy of errors which could occur in each level of SA based on multiple studies, see figure 3. These studies show that errors most often occur when a human misses certain information or fails to correctly process the incoming data, which can have several causes [3]. Using the knowledge about SA in a DSS might prevent these errors, since a DSS also makes a decision based on incoming data and the reasoning system it is programmed with.

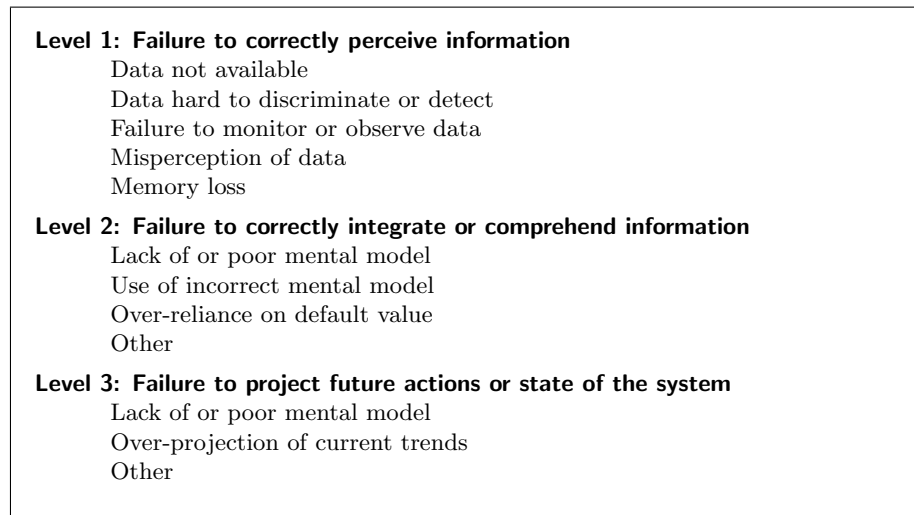


Fig. 3: SA Error Taxonomy [3]

2.3 Decision Support Systems

Decision Support Systems (DSS) are most often used when the incoming information must be evaluated to decide which action to take. They are seen in security settings and factories to support human supervisors. A DSS can function in several ways. The primary idea is that it processes incoming information and reports only necessary information to the supervisor [18]. Various techniques, such as auditory and/or visually cues, can be used to guide human attention to the desired part of the situation. To be able to aid the human in the decision making, the DSS needs to assess the task at hand. Therefore, a DSS is equipped with a reasoning system. Such a reasoning system can be as simple as a rule-based system, to a more complicated Bayesian Belief Network or even a complicated neural network. In previous studies multiple DSS techniques were already used, aiming for an optimal SA [1, 18, 2]. Some of these DSSs showed promising results.

Vachon et al. [18] have performed an elaborate study on DSSs in a complex dynamic situation which is comparable to the situation in this study. They programmed two different DSSs, one for level one SA and one for levels two and three SA. Their results showed that using a DSS resulted in an increase of the human supervisor's SA. However, the DSSs had a negative effect on performance in situations which involved high cognitive load for the supervisors, due to the overhead interpreting the DSS data. Therefore they recommended a more detailed DSS which is better aligned with a human supervisor, so they could work as a single unit. Cuchiarrá et al. [2] created a rule-based reasoning system with a module for level one SA and a module for levels two and three SA, which showed promising results in tracking vehicles from a camera feed. In a large study from Arziszewski and colleagues [1] it was tested whether supervisors would benefit from an adaptive DSS in which the supervisor could manually choose how much command the system was allowed and whether the system should engage in increased self-control when a situation becomes critical or stressful. In the reported workload of the human supervisors, no differences were found however, their performance was improved when the system intervened in critical situations.

3 Model

This study focuses on building a reasoning system for a DSS, which is used to detect threats in traffic situations by calculating the threat level for each vehicle. We created multiple models to determine which combination of state variables perform best.

Each model uses one or more state variables, which we call predictors, to calculate the threat level of a vehicle. Each predictor represents a certain attribute of the state of the vehicle. For example, the current speed of the vehicle. The more predictors a model uses the more information about the vehicle state it has to calculate the vehicle threat level.

A threat level ranges from one (none threatening) to ten (very threatening). Given the current state of a vehicle, each model calculates a prediction for each of the ten threat levels, the threat level of a vehicle is then calculated using

equation (1),

$$\text{Vehicle threat level} = \sum_{i=1}^{10} P_{\text{ThreatLevel}}(i) * i \quad (1)$$

where $P_{\text{ThreatLevel}}(i)$ denotes the model's prediction for threat level i . The prediction of each separate threat level is based on Bayesian belief networks [19].

3.1 Bayesian Belief Networks

A Bayesian Belief Network (BBN) is a graphical representation of a joint probability distribution on a set of statistical variables [19]. A Directed Acyclic Graph (DAG) is a set of nodes and links that specifies the conditional independence relationships that hold in the domain [15]. Since it is most often easy for a domain expert to decide which conditional independence relationships hold in the domain, the DAG presents a nicely structured knowledge representation of the desired domain.

Each node has an associated conditional probability distribution, which describes the influence of the nodes' parents on the probabilities of the node itself. Due to the conditional independence distributions, BBNs can be more compact than full joint probability distributions [15]. Therefore, BBNs can contain a large amount of variables without exponential growth of the conditional probability factors.

Although the DAG provides a nice knowledge representation of the domain, the real strength of BBNs is the influence on the nodes' probabilities when applying Bayes inference rules to propagate evidence through the network [15]. Using evidence propagation the network is able to answer queries and "what-if" questions about the variables within the network. Due to the relational nature of the DAG, these queries may include predictive reasoning (i.e., predicting the threat level of a vehicle, given its current state), diagnostic reasoning (i.e., given a certain threat level distribution, which predictor without evidence will contribute the most to this outcome) and inter-causal reasoning (e.g., given two mutually exclusive predictors, evidence on one of them will rule out the other) [15].

Due to the intuitive knowledge representation and the possibility to query this knowledge to answer several types of questions, a BBN fits well in the HMI paradigm. Since the conditional independence relationships are embedded in the DAG, the output of a BBN will be relatively easy to comprehend for a domain expert. Combined with the ability to query this knowledge, a BBN facilitates an interactive design which can easily implemented in a HMI system like a DSS.

3.1.1 Structure and parameters of a Bayesian Network

As mentioned previously, a DAG consists of variables, X_1, X_2, \dots, X_n and their parental relation to each other (see fig 4). Each node represents a variable and has a conditional probability distribution for that variable, given the status of the nodes' parents, So the joint probability of the entire DAG is:

$$p(X_1 = x_1, X_2 = x_2, \dots, X_n = x_n) = p(x_1, x_2, \dots, x_n) = \prod_{i=1}^n p(x_i | x_{pa(i)}) \quad (2)$$

where $p(x_1, x_2, \dots, x_n)$ denotes a specific combination of values x_1, x_2, \dots, x_n from the set of variables X_1, X_2, \dots, X_n , also known as a configuration, and $x_{pa(i)}$ represents the configuration of the parents of X_i , given the current DAG [9].

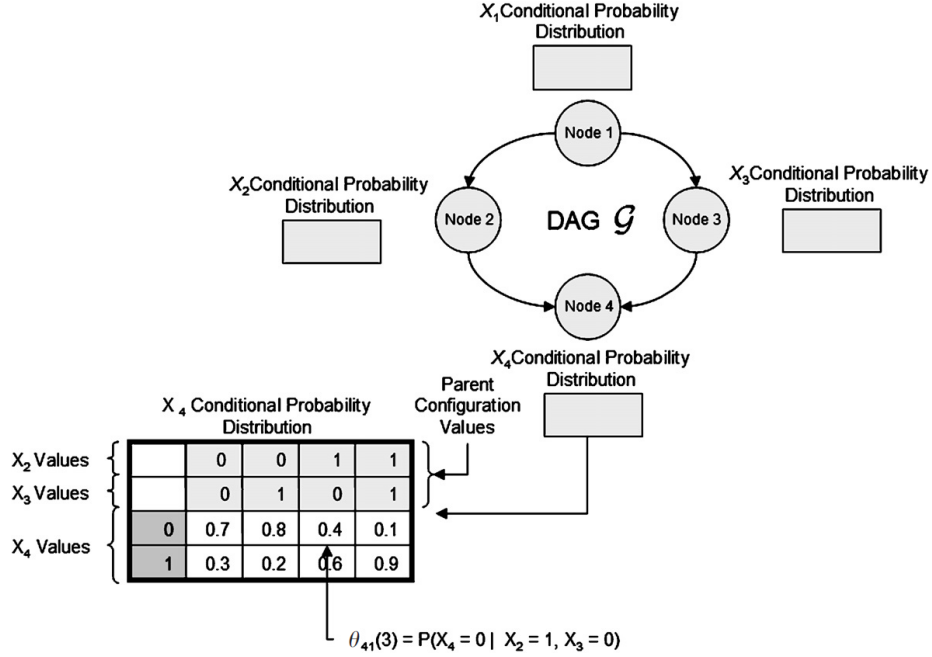


Fig. 4: A four-variable BBN

This figure depicts a four-variable BBN, DAG \mathcal{G} , where all variables, X_1, X_2, X_3, X_4 , are binary with 0 and 1 values. Additionally, the variables' conditional probability distributions as straightforward conditional probability distribution tables. Parameter $\theta_{41}(3) \equiv P(X_4 = 0 | X_2 = 1, X_3 = 0)$ represents the conditional probability, 0.4, that $X_4 = 0$, given the parents' values, $X_2 = 1, X_3 = 0$. [9]

The conditional probability distribution of a BBN with DAG \mathcal{G} can be represented as a set of parameters $\Theta = \{\Theta_1, \Theta_2, \dots, \Theta_n\} = \{[\theta_{ik}(j)]_{k=1}^{r_i}\}_{j=1}^{q_i}$, where $i = 1, \dots, n$ denotes each variable X_i ; $k = 1, \dots, r_i$ denotes all r_i variable values of X_i (e.g., $r_i = 2$ for binary variables); and $j = 1, \dots, q_i$ denotes the set of q_i valid parent configurations ($x_{pa(i)}$). Essentially Θ_i denotes the conditional probability distribution of X_i . Hence, for any configuration $\mathbf{c} = \{x_1, x_2, \dots, x_n\}$, equation (2) can be written as:

$$p(\mathbf{c} | \Theta, \mathcal{G}) = \prod_{i=1}^n p(x_i | x_{pa(i)}, \Theta_i, \mathcal{G}) = \prod_{i=1}^n \theta_{ik}(j) \quad (3)$$

where $\theta_{ik}(j)$ denotes the probability of X_i , given x_i is value k and parent configuration j [9]. A BBN with n binary nodes and an average of p parents per node, would require $n \times 2^p$ values to specify a full probability model, which is a lot smaller considering the 2^n values required for a full joint probability distribution.

In order to create a functional BBN, one has to estimate the values for Θ . In this study we used supervised learning with Leave-One-Out cross-validation to estimate the values for Θ [15].

4 Hypotheses

Based on earlier mentioned research on DSSs and the models used in this study, two hypotheses were formed. These hypotheses will address the main question of this study; which elements are useful for a DSS to assess possible threats in traffic situations?

The DSS models were trained using data from human test participants and use the vehicle state to calculate the vehicle threat level. The amount of predictors varies between the DSS models. Therefore, it is to be expected that a model with more predictors will perform better than a model with fewer predictors, since they have more information about the vehicle state. The amount of predictors is expected to be an important element for the assessment of possible threats in traffic situations.

Hypothesis 1. The more complex a model, i.e. the more predictors it has, the better the performance will be. Meaning that a more complex model yields better predictions.

Although it is to be expected that a more complex model will perform better than a less complex model, it could still be the case that a more complex model performs bad overall. To investigate whether trained models using the vehicle state perform well overall, these models will be compared to untrained models returning a fixed vehicle threat level, regardless the vehicle state. This led to the following hypothesis.

Hypothesis 2. A trained DSS model using the vehicle state to calculate the vehicle threat level, will perform better than all models which return a fixed vehicle threat level regardless the vehicle state.

5 Method

5.1 Participants

For this study average civilians of the Dutch population were asked to volunteer on a computer task, which took 45 to 60 minutes. Each participant received a test booklet, see Appendix A, containing an informed consent, short questionnaire, test explanation, answer sheets and a wellness gift card form. Between all participants who filled in the wellness gift card form a wellness gift card was awarded at random. The test explanation was included in the test booklet such that each participant would have the same explanations, without interference or suggestions from the test leader. When a participant had questions there was always a test leader present to answer them.

At the beginning of each test, two short test simulations were run. During these test simulations the test leader checked whether the test environment worked correctly and was available to address any uncertainties regarding the given task. After these short test simulations the actual test started. When

the test was completed, each participant handed in their test booklet and was thanked for their participation.

In total 56 participants volunteered, 37 men and 19 women. One volunteer was excluded from participation for not being able to complete the task.

5.2 Materials

For this study the following materials were created.

5.2.1 Traffic Simulation

To generate training data that was similar to real world situations image analysts of the Royal Armed Forces of the Netherlands would encounter, a simulation environment was created. In this simulation participants were presented with a partial overview of The Hague, Netherlands [5]. On this map different vehicles were present, each with their own number (see figures 5 and 6). Multiple situations were created which all took place at the same map, therefore participants would get familiar with the context reducing test time. To compensate for this learning effect, each participant was presented with the situations in a different order.

There were a total of eight situations divided equally into two categories. Category one contains situations with a restricted area in which no vehicles are allowed and a semi-restricted area in which only destination traffic is allowed, see figure 5. Though there is no visual distinction between traffic that is or is not allowed in the semi-restricted area. Category two contains situations with a marked route on the map, along which a convoy will travel to its final destination in the center of the map, see figure 6.

Each simulation lasted 60 seconds, divided in six parts of ten seconds each. At the beginning of each new situation the simulation was paused in order for the participants to orient them about the situation. At the end of each part, the simulation was paused and the participants were asked to rate all vehicles present on the screen.



Fig. 5: Overview of situation in Traffic Simulation category 1

The dotted zone indicates a restricted, no-go, area. The dashed zone indicates a semi-restricted area where only destination traffic is allowed.



Fig. 6: Overview of situation in Traffic Simulation category 2

The dashed line indicates the route of the convoy (vehicles A, B, C). The convoy will always travel to the center of the map.

The test environment was based on SUMO [8]. SUMO is a simulation program which is able to simulate traffic. SUMO facilitates a top-down view of the situation (see figures 5, 6). The benefit of such simulation like SUMO is that one can control the entire environment, in order to provide equal test scenarios for all participants. Another benefit of SUMO is the fact that roadmaps from OpenStreetMaps [20] can be converted to be used within SUMO.

Traffic Simulation was created with SUMO, provided with a roadmap from OpenStreetMaps and an underlying image from Google Maps [5] to provide the participants with an environmental context.

5.2.2 Training data

Since there were a total of eight situations, six stops per situation and twelve items per stop, the entire data set consist of 576 data points per participant. However, due to the length of the test each participant was only presented with a total of four different situations, resulting in at least 25 participants per situation.

The training data consists of all participant data of 575 data points, participant data of one item is left out which is used to measure the performance of the model. This was repeated so all data points are left out once, resulting in a total of 576 performance data points per model. Each model was trained using frequency counting of the training data.

This approach however, did not produce accurate results since each participant threat level was weighted equally, resulting in less accurate detection of higher threat levels. The reason for this was the high spread of participant answers for each vehicle. In order to counter this effect while still using frequency counting, the training data was altered to a linear weighted dataset. Instead of adding each participant threat level just once to the dataset, higher threat levels were added as many times as the given threat level. This resulted in a dataset, where a participant threat level of ten weights ten times more than a participant threat level of one.

5.2.3 DSS models

Multiple extensive and less extensive DSS models were programmed to judge the situations of Traffic Simulation. These models were created with Matlab [10] and the Bayes Net Toolbox for Matlab [12]. The aim for each DSS was to perform the exact same task as the human participant's. In order to assess a vehicle, the model used the current state of the vehicle. As seen in the study of Cucchiara et al [2] commonly used predictors are position, speed and shape. In this study the vehicle states were based on the following five predictors: speed, zone, distance, heading and junction.

A total of 31 models were created, based on the five different predictors (see table 1). Each model consists of one or more predictors in order to calculate the threat level of a given vehicle, see table 2 for the distribution of predictors among the different models. A more graphical representation of model 31 is shown in figure 7.

Speed The speed predictor is based on the maximum allowed speed of a given road.

Normal Indicates a speed of 10% to 110% of the allowed speed

Stop Indicates that a vehicle is completely stopped

Slow Indicates a speed $< 10\%$ of the allowed speed

Fast Indicates a vehicle speed of $> 110\%$

Zone The zone predictor indicates in what kind of zone the vehicle currently is in.

Yellow Indicates a semi-restricted area where only destination traffic is allowed (see figure 5).

Red Indicates a restricted no-go area where no traffic is allowed (see figure 5).

Route Indicates whether a vehicle is present somewhere on the route of the convoy (see figure 6).

None Indicates that a vehicle is not in any of the above mentioned zones.

Distance The distance predictor indicates the distance between the vehicle and the closest edge of the red zone (see figure 5), or the distance between the vehicle and the central car of the convoy (see figure 6).

Heading The Heading predictor is based on the distance difference of the previous time step and the current time step.

Junction The junction predictor indicates whether a vehicle is currently placed on a junction or not.

Tab. 1: Prediction priors

Speed	Zone	Distance	Heading	Junction
Normal	None	≤ 50	Closing in	True
Stop	Route	≤ 100	Moving away	False
Slow	Yellow	≤ 200	Parallel	
Fast	Red	≤ 300		
		≤ 400		
		≤ 500		
		> 500		

Tab. 2: Used predictors per model

Model number	Speed	Zone	Distance	Heading	Junction
1	x				
2		x			
3			x		
4				x	
5					x
6	x	x			
7	x		x		
8	x			x	
9	x				x
10		x	x		
11		x		x	
12		x			x
13			x	x	
14			x		x
15				x	x
16	x	x	x		
17	x		x	x	
18	x	x		x	
19	x	x			x
20	x		x		x
21	x			x	x
22		x	x	x	
23		x	x		x
24		x		x	x
25			x	x	x
26	x		x	x	x
27	x	x	x	x	
28	x	x		x	x
29	x	x	x		x
30		x	x	x	x
31	x	x	x	x	x

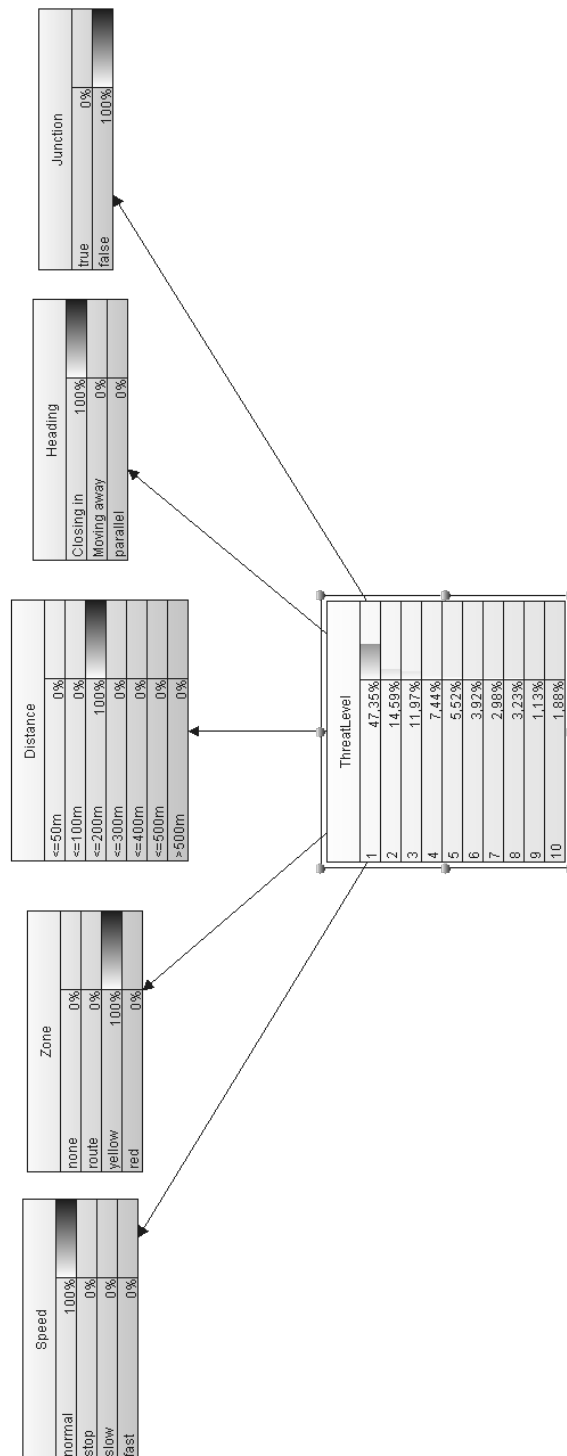


Fig. 7: Graphical representation of model 31 with entered vehicle state.

In order to rule out sampling errors and to test hypothesis two, ten extra models were created. These models return a fixed vehicle threat level independent of the vehicle state. Model I always returns one, model II always returns two, model III always returns three, until model X which always returns a vehicle threat level of ten.

The fixed models represent undesired behavior of a DSS model and form a control group for the trained models. Since a fixed model always returns a certain vehicle threat level independent of the vehicle state, these models would be unable to properly support the human domain experts. Therefore, in order for the trained models to be considered performing well, they have to perform significantly better than all fixed models. Otherwise the performance of the trained models could be due to sampling errors.

5.3 Statistical analysis

For all tests, a significance level of $p < 0.05$ was maintained [11]. Before the DSS models were trained with the test data of the participants the data was screened for outliers. The mean score of each data point was calculated. When a participant score differed more than two standard deviations from the mean, the participant score for that data point was removed. When two or more data points from the same participant were removed in the same fragment, all data points of that participant were removed from the entire situation. The remaining dataset was used for training the DSS models.

Using the participant data, all Bayesian models were trained using frequency counting and leave-one-out cross-validation [15]. This means that each time a model trains a situation one item score is left blank which the model must evaluate. By repeating this, a model will have judged all items once. These 576 results are then used to calculate the performance of the model. In order to calculate the performance of a model, the participant scores of the left out item are compared to the judgment of the model for that item, using equation 4:

$$\sum_{n=1}^{576} \sum_{p=1}^{p_n} |s_{pn} - s_{mn}| \times s_{pn} \quad (4)$$

where p_n is the number of participant scores for item n , s_{pn} is the participant score of participant p for item n and s_{mn} is the model score for item n . Meaning that a model will be punished more when it deviates from a higher participant threat level and the model with the lowest performance score performs best.

To test the first hypothesis a one-way ANOVA was used to investigate whether there were significant differences between the DSS models [11]. When a significant result was found, a Tukey HSD post hoc analysis was performed to evaluate which models differed significantly. The reason to perform Tukey HSD post hoc analysis instead of multiple t-tests is due to the fact that Tukeys HSD corrects for type 1 errors [11].

For the second hypothesis multiple one-way ANOVAs were used to compare all trained models with each fixed model. Since we were only interested in the possible differences between the trained models and each of the fixed models and not in the differences between the fixed models, a one-way ANOVA between the trained models and each of the fixed models I to X was performed in order to study whether there were significant differences. Using a one-way ANOVA per

fixed model reduces data overlap in the results [11]. When significant results were found Tukey HSD post hoc analyses were performed.

6 Results

In order to determine which elements are best suited for a DSS to assess vehicle threats, the 31 DSS models were evaluated.

6.1 Differences in between the trained DSS models

The first hypothesis states that the more complex a model, the better it will perform. To test whether more complex models indeed have better results, a one-way ANOVA was performed to indicate whether there were significant differences between the model performances.

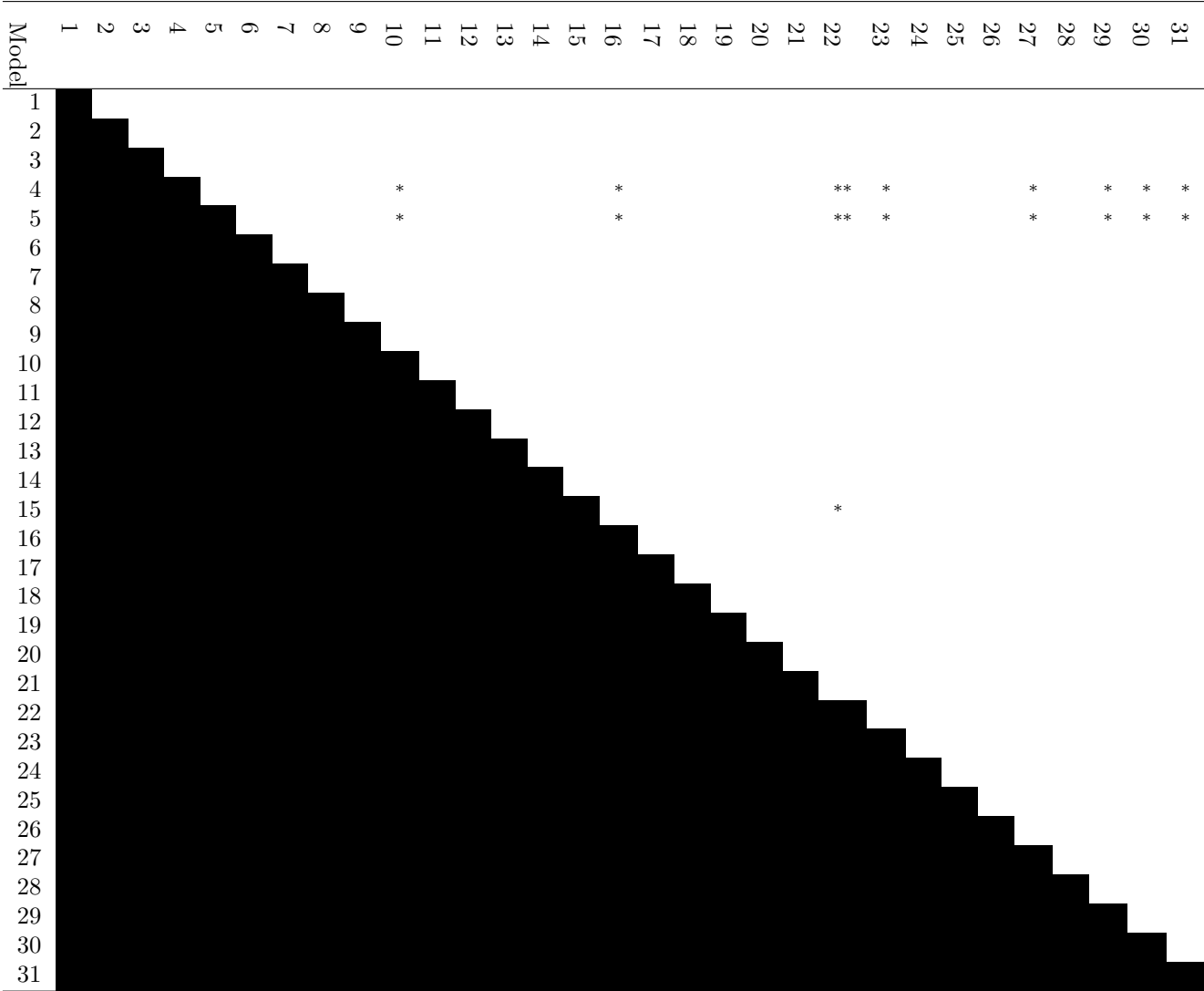
A significant effect was found: $p < 0.05$, ($F(30, 17825) = 2, 58, p = 0.0000$). To indicate between which specific models these significant differences occurred, a Tuckey HSD post hoc analysis was performed. The Tuckey HSD post hoc analysis showed seventeen significant differences, see table 3. Both models 4 and 5 differed significantly from models 10, 16, 22, 23, 27, 29, 30 and 31 while models 15 and 22 differed significantly as well.

When closer examining these models it showed that model 4 and 5 had less predictors than models 10, 16, 22, 23, 27, 29, 30 and 31 (see table 2). However, we were still unable to conclude which group of models (4 and 5 versus 10, 16, 22, 23, 27, 29, 30 and 31) performed better. Therefore the performance scores of the models had to be compared (see table 4).

Table 4 showed that models 10, 16, 22, 23, 27, 29, 30 and 31 have a lower performance score than models 4 and 5. According to equation 4, this indicates that models 10, 16, 22, 23, 27, 29, 30 and 31 perform significantly better than models 4 and 5. The same holds for models 22 and 15. Model 22 has more predictors and performs significantly better than model 15 according to tables 2, 3 and 4

The significant differences between the trained models showed that models with more predictors perform better therefore, hypothesis one was accepted.

Tab. 3: Tukey HSD post hoc analysis



*, $p < 0.05$, **, $p < 0.01$

Tab. 4: Performance scores of the trained models

model	performance	model	performance	model	performance
1	157,7056999	12	151,568891	23	143,5992329
2	150,8646652	13	153,9409896	24	150,278639
3	155,2331885	14	153,8449198	25	151,9031742
4	168,9228728	15	166,7971214	26	148,4818452
5	169,1754054	16	143,7997717	27	143,5469104
6	150,1860152	17	147,3490919	28	150,0566011
7	148,4451604	18	149,4314595	29	144,6466635
8	156,9941808	19	150,6031461	30	143,2847355
9	157,7246022	20	149,1822632	31	144,5375811
10	143,1331061	21	157,6077212		
11	149,7324792	22	142,1066702		

6.2 Differences between the trained and fixed DSS models

The second hypothesis states that each DSS model will perform better than a fixed model. It was first tested whether there were significant differences between the fixed models and the trained models using one-way ANOVAs. All ANOVAs showed significant results (see Table 5), meaning that in each set significant differences were found. The Tukey HSD post hoc analyses showed multiple significant results which are displayed in Table 6.

Tab. 5: ANOVA analysis between trained and fixed models

Model	DF	F	p
I	31	16.27	0.0000
	18400		
II	31	7.18	0.0000
	18400		
III	31	3.58	0.0000
	18400		
IV	31	2.92	0.0000
	18400		
V	31	3.13	0.0000
	18400		
VI	31	4.87	0.0000
	18400		
VII	31	9.79	0.0000
	18400		
VIII	31	20.65	0.0000
	18400		
IX	31	42.9	0.0000
	18400		
X	31	77.45	0.0000
	18400		

Tab. 6: Tukey HSD post hoc analyses between trained and fixed models

Model	I	II	III	IV	V	VI	VII	VIII	IX	X
1	***	***				***	***	***	***	***
2	***	***	**			***	***	***	***	***
3	***	***	*			***	***	***	***	***
4	***	***					***	***	***	***
5	***	***					***	***	***	***
6	***	***	***			***	***	***	***	***
7	***	***	***			***	***	***	***	***
8	***	***				***	***	***	***	***
9	***	***				***	***	***	***	***
10	***	***	***	**	**	***	***	***	***	***
11	***	***	***			***	***	***	***	***
12	***	***	**			***	***	***	***	***
13	***	***	**			***	***	***	***	***
14	***	***	**			***	***	***	***	***
15	***	***					***	***	***	***
16	***	***	***	*	**	***	***	***	***	***
17	***	***	***		*	***	***	***	***	***
18	***	***	***			***	***	***	***	***
19	***	***	**			***	***	***	***	***
20	***	***	***			***	***	***	***	***
21	***	***				***	***	***	***	***
22	***	***	***	**	**	***	***	***	***	***
23	***	***	***	*	**	***	***	***	***	***
24	***	***	***			***	***	***	***	***
25	***	***	***			***	***	***	***	***
26	***	***	***			***	***	***	***	***
27	***	***	***	*	**	***	***	***	***	***
28	***	***	***			***	***	***	***	***
29	***	***	***	*	**	***	***	***	***	***
30	***	***	***	*	**	***	***	***	***	***
31	***	***	***	*	**	***	***	***	***	***

*: $p < 0.05$, **: $p < 0.01$, ***: $p < 0.001$

Table 6 showed that all trained models differed significantly from the fixed models which always score one, two, seven, eight, nine or ten. Less significant differences were found between all trained models and the fixed models which always score four or five. In table 7 the performance scores of all models are shown.

According to table 7, models IV and V provided the best performance of the fixed models, yet they still performed significantly worse than models 10, 16, 22, 23, 27, 29, 30 and 31.

These results showed that trained models 10, 16, 22, 23, 27, 29, 30 and 31 perform significantly better than all fixed models, so their performance is not due to sampling errors. Therefore, hypothesis two was accepted.

Tab. 7: Performance scores of the trained models and the fixed models

model	performance	model	performance	model	performance	model	performance
1	157,7056999	12	151,568891	23	143,5992329	I	258,5833333
2	150,8646652	13	153,9409896	24	150,278639	II	213,3541667
3	155,2331885	14	153,8449198	25	151,9031742	III	182,2013889
4	168,9228728	15	166,7971214	26	148,4818452	IV	170,8715278
5	169,1754054	16	143,7997717	27	143,5469104	V	172,4722222
6	150,1860152	17	147,3490919	28	150,0566011	VI	190,375
7	148,4451604	18	149,4314595	29	144,6466635	VII	219,3611111
8	156,9941808	19	150,6031461	30	143,2847355	VIII	258,6284722
9	157,7246022	20	149,1822632	31	144,5375811	IX	311,7569444
10	143,1331061	21	157,6077212			X	370,6666667
11	149,7324792	22	142,1066702				

7 Discussion

The aim of this study was to investigate which elements are best suited for a DSS in order to assess vehicle threats in certain traffic situations. Multiple DSS models were created and trained to see which amount of predictors performs best and to test whether their performance is not due to sampling error. Test results of participants were used to train the models and a vast test environment was created to reduce error. The results indicate which predictors are necessary for a well performing DSS.

The first hypothesis stated that the more complex a model the better it would perform. For traffic situations the following five predictors were relevant, as were based upon the predictors from the study of Cucchiara et al [2], Speed, Zone, Distance, Heading and Junction. In order to investigate whether adding more predictors yields better performance, 31 DSS models were created covering all possible combinations of the five predictors. To test the first hypothesis the performance scores of all trained models were compared. It was expected that there would be significant differences in performance between the models and that the models with more predictors would perform better. The results showed that there were indeed significant differences between the models, meaning that certain models are better than others. On closer examination of the results it was found that models 10, 16, 22, 23, 27, 29, 30 and 31 perform significantly better than models 4 and 5. Also model 22 performed significantly better than model 15. Even so, it is important to note that models 4 and 5 both had one predictor, while models 10, 16, 22, 23, 27, 29, 30 and 31 all had multiple predictors. This indicates that models with more predictors indeed perform better and therefore the first hypothesis was accepted.

The second hypothesis states that a trained model always performs better than a model which generates a fixed score. To test this, ten extra models were created which always generated a fixed score, i.e. from one to ten. The results showed that most trained models performed significantly better than all fixed models. When taking into account the performance scores of the other trained models, it became apparent that even though not significantly, all trained models performed better than all fixed models, which is in line with the findings of Vachon and colleagues [18]. Therefore, the second hypothesis was accepted as

well

When taking a closer look at the results, there were several interesting findings. All models that performed significantly better than models 4 and 5 were all based on both the Zone and Distance predictor while models 4 and 5 were only based on one predictor, Heading or Junction. This implies that each predictor adds a certain weight to the performance of the model, which is also in line with the findings of Vachon and colleagues [18]. Looking at all single predictor models and taking into account the above implication, one can order the predictors based on their performance score as follows, from best to worst performance: Zone, Distance, Speed, Heading, Junction.

The above ordering suggests that any model with a combination of the Zone, Distance and Speed predictors would always outperform a model in which the Speed predictor is replaced with either the Heading or Junction predictor. However, this is not the case. When comparing the performance of all models which include both Zone and Distance, another interesting discovery was made. Adding the Speed and/or Junction predictor to the model consisting of only Zone and Distance results in an overall performance decrease. While this supports the implication that each predictor adds a certain weight to the model's performance, adding poor performing predictors might also hurt a model's performance.

Although the above would imply that adding a poor performing predictor to the model containing the Zone and Distance predictors will hurt the model's performance, the opposite holds when adding the Heading predictor. When comparing the models with only a single predictor, the model based on the Heading predictor has the second worst performance of all five predictors. However, only when adding the Heading predictor to the model containing the Zone and Distance predictor an increase in overall performance was seen. Even though the Heading predictor alone might yield a poor performance, it was the only predictor that showed a vast increase in performance when combined with either the Zone or Distance predictor. Implying that a poor performing predictor might vastly increase performance when combined with the right predictors. Although these are interesting findings, none of them showed a significant difference in performance and therefore might due to sampling errors.

In summary, the significant results showed that models with more predictors performed better. However, there are certain nuances between models 10, 16, 22, 23, 27, 29, 30 and 31 that might suggest otherwise. The combination of predictors a model is based upon is an important element to its performance. Next to performance, the DSS models were only labeled as being useful when their results mirrored those of the participants, meaning a model highlighted the same possible threats.

While created carefully with domain experts, the simulations presented to the test participants were very simplified versions of reality. Therefore, it is hard to predict whether the best performing model of this study would still perform well enough in real world scenarios. Though, in real world scenarios, a human supervisor would have to oversee thousands of vehicles and point out possible threats in a matter of seconds [2, 9, 3]. In such scenarios a human supervisor will suffer from either cognitive lock up or cognitive overload and will need to be supported to be able to perform his tasks.

8 Conclusions

This study investigated which elements are useful for a DSS to assess possible threats in traffic situations. Multiple DSS models were created based upon five predictors; Zone, Junction, Heading, Speed and Distance. The models were trained with average participants' test data to evaluate their performance. In order to study which models would perform best and whether trained models would perform better than fixed models, ten fixed models were created. The trained models were based on the five different predictors while the fixed models always returned a fixed value.

The results showed that adding more predictors yields a better performance. It appeared that the model based on the Zone, Distance and Heading predictors performed best. All other models that also performed significantly better than the worst performing models were based on Zone and Distance as well, which is in line with the performance scores of the single predictor models. Furthermore, the results showed that each trained model performed better than all fixed models. Therefore, it can be concluded that in order for a DSS to correctly identify vehicle threats in traffic situations the following elements are of importance: the amount of predictors, the combination of predictors and training of the models.

9 Future work

In this study we showed that the presented training method is able to train models in such a way that they assess possible threats in traffic situations. However, there are opportunities for further research based on this study.

First of all, since the models in this study are fit on the gathered training data from average test participants, it could be the case that the presented models will not perform best when supporting domain experts. Therefore, we recommend to use domain expert training data in order to fit the models as best as possible for the given task.

Secondly, this study only presented a reasoning system. This system should still be embedded in a DSS in order to investigate how to present the data from the reasoning system to domain experts and whether the DSS is able to provide enough support to domain experts in order to prevent a cognitive lock up.

Thirdly, although all tested situations were carefully created with the aid of domain experts, they represent a very simplified version of reality. Further study should indicate whether these models perform well enough in real world situations where hundreds of vehicles need to be processed instead of twelve.

Lastly, for the presented reasoning system there are several promising opportunities to research how well it would perform in combination with adaptive autonomy [1, 3, 18, 7]. Since the presented reasoning system is based on a Bayesian belief network using conditional probability distributions, other learning strategies to estimate Θ may be considered. It could, for example, be interesting to combine supervised learning, to determine the conditional probability distribution priors, with online learning, to update the conditional probability distribution during a real world scenario. Another possibility would be to research how long it would take for the presented reasoning system to reach adequate performance using only online learning or, due to the modular nature of the reasoning system, how long it would take for the reasoning system to

adapt when new predictors are added.

References

- [1] Henryk FR Arciszewski, Tjerk E De Greef, and Jan H Van Delft. Adaptive automation in a naval combat management system. *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on*, 39(6): 1188–1199, 2009.
- [2] Rita Cucchiara, Massimo Piccardi, and Paola Mello. Image analysis and rule-based reasoning for a traffic monitoring system. *Intelligent Transportation Systems, IEEE Transactions on*, 1(2):119–130, 2000.
- [3] Mica R Endsley. Situation awareness and human error: Designing to support human performance. In *Proceedings of the high consequence systems surety conference*. Lawrence Erlbaum Associates, 1999.
- [4] Mica R Endsley. *Designing for situation awareness: An approach to user-centered design*. Taylor & Francis US, 2003.
- [5] Google. Google maps, 2014. URL <https://www.google.nl/maps/@52.0861132,4.2949674,1417m/data=!3m1!1e3>.
- [6] P.A. Hancock. In search of vigilance: The problem of iatrogenically created psychological phenomena, 2013. tandf.
- [7] PA Hancock. Task partitioning effects in semi-automated human-machine system performance. *Ergonomics*, 56(9):1387–1399, 2013.
- [8] Daniel Krajzewicz, Jakob Erdmann, Michael Behrisch, and Laura Bieker. Recent development and applications of SUMO - Simulation of Urban MObility. *International Journal On Advances in Systems and Measurements*, 5(3&4):128–138, December 2012. URL <http://elib.dlr.de/80483/>.
- [9] Eitel J.M. Laura and Peter J. Duchessi. A bayesian belief network for it implementation decision support. *Decision Support Systems*, 42(3):1573 – 1588, 2006. ISSN 0167-9236. doi: <http://dx.doi.org/10.1016/j.dss.2006.01.003>. URL <http://www.sciencedirect.com/science/article/pii/S0167923606000078>.
- [10] Mathworks. Matlab 2014b, 2014. URL <http://nl.mathworks.com/products/matlab/>.
- [11] Lawrence S. Meyers, Glenn Gamst, and A.J. Guarino. *Applied multivariate research: Design and interpretation*. SAGE Publications, 2006.
- [12] Kevin Murphy. Bayes net toolbox for matlab, 2014. URL <https://code.google.com/p/bnt/>.
- [13] Mohsen Naderpour, Jie Lu, and Guangquan Zhang. An intelligent situation awareness support system for safety-critical environments. *Decision Support Systems*, 59:325–340, 2014.
- [14] Raja Parasuraman, Thomas B Sheridan, and Christopher D Wickens. A model for types and levels of human interaction with automation. *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on*, 30(3):286–297, 2000.

- [15] Stuart Russell and Peter Norvig. *Artificial Intelligence A Modern Approach*. Prentice Hall, 2nd edition, 2003.
- [16] Andrzej MJ Skulimowski. Future trends of intelligent decision support systems and models. In *Future Information Technology*, pages 11–20. Springer, 2011.
- [17] Sandesh Uppoor and Marco Fiore. Mobicom 2011 poster: vehicular mobility in large-scale urban environments? *ACM SIGMOBILE Mobile Computing and Communications Review*, 15(4):55–57, 2012.
- [18] François Vachon, Daniel Lafond, Benoît R Vallières, Robert Rousseau, and Sébastien Tremblay. Supporting situation awareness: A tradeoff between benefits and overhead. In *Cognitive Methods in Situation Awareness and Decision Support (CogSIMA), 2011 IEEE First International Multi-Disciplinary Conference on*, pages 284–291. IEEE, 2011.
- [19] Linda van der Gaag and Silja Renooij. *Probabilistic Reasoning*. Utrecht University, 2014.
- [20] OpenStreetMap Wiki. Main page — openstreetmap wiki, 2014. URL <http://wiki.openstreetmap.org/>.

A Participant Booklet

Onderzoeks-handleiding

- Onderzoek naar de beoordeling van de veiligheid van situaties -

Uitgevoerd als onderdeel van het afstudeeronderzoek voor de opleiding AI Informatica van:

Marien Spek BSc

Onder leiding van:

Prof. DR. John-Jules HC. Meyer (Universiteit Utrecht),

en

Dr. Peter-Paul van Maanen (TNO Soesterberg)

Doel van dit onderzoek

Steeds vaker nemen computers het werk over van mensen. Dit gebeurt bijvoorbeeld in fabrieken en bij banken, maar ook bij instanties die de veiligheid moeten bewaken en waarborgen. In winkelcentra hangen bijvoorbeeld een heel aantal camera's die door één beveiligiger wordt gemonitord. De werkdruk van deze persoon kan soms te hoog worden, met als gevolg dat hij dingen over het hoofd gaat zien. Het is daarbij dus van belang dat we weten wanneer en hoe mensen ondersteund kunnen worden door computers om dit te voorkomen. Dit onderzoek dient er toe om hier meer over te weten te komen, zodat in de toekomst nog beter op deze situaties kan worden ingespeeld.

Het experiment

Het onderzoek bestaat uit het invullen van een korte vragenlijst en 4 computertaken op de computer. Het hele onderzoek duurt ongeveer 60 minuten.

Deelname aan dit onderzoek is geheel vrijwillig en vrijblijvend. Alle informatie die in het kader van dit onderzoek wordt verzameld wordt strikt vertrouwelijk en anoniem behandeld. Er zal voor worden gezorgd dat derden geen inzage krijgen in uw gegevens en ook dat de gegevens niet tot personen terug te leiden zijn.

Voor meer informatie over dit onderzoek, kunt u mailen naar: Marien Spek BSc:

afstudeeronderzoek_marien@outlook.com.

Vergoeding

Door deelname aan dit onderzoek kunt u een saunabon winnen t.w.v. 30 euro. Zie voor meer informatie de laatste pagina.

Deelnamecriteria en ondertekening

Door ondertekening van dit formulier geeft u aan bovenstaande te hebben gelezen en begrepen en dat u:

- tussen de 18 en 65 jaar oud bent
- Nederlands sprekend bent
- goed zichtvermogen heeft
- een opleidingsniveau heeft vergelijkbaar aan MBO+ of hoger

Datum:

Plaats:

Handtekening:

Vragenlijst achtergrondinformatie

Probeer alle vragen naar waarheid te beantwoorden en lees indien van toepassing de cursief gedrukte toelichting aandachtig door. Kruis het juiste alternatief aan of vermeld uw antwoord in een open ruimte.

Algemene vragen:

Wat is uw geboortedatum? ----/----/----

Wat is uw geslacht? M/V (omcirkel a.u.b. het juiste antwoord)

Welke nationaliteit heeft u?

Wat is de nationaliteit van uw vader?

Wat is de nationaliteit van uw moeder?

In welk land bent u geboren?

Wat is de hoogst genoten opleiding? (*u hoeft deze opleiding (nog) niet te hebben afgerond*).....

Bent u : 0 Linkshandig *Kruis aan wat van toepassing is*
 0 Rechtshandig

Draagt u:

<input type="checkbox"/> een bril	<i>Kruis aan wat op dit moment van</i>
<input type="checkbox"/> lenzen	<i>toepassing is</i>
<input type="checkbox"/> Geen van beide	

Bent u kleurenblind? Ja/ Nee (*Omcirkel a.u.b. het juiste antwoord*)

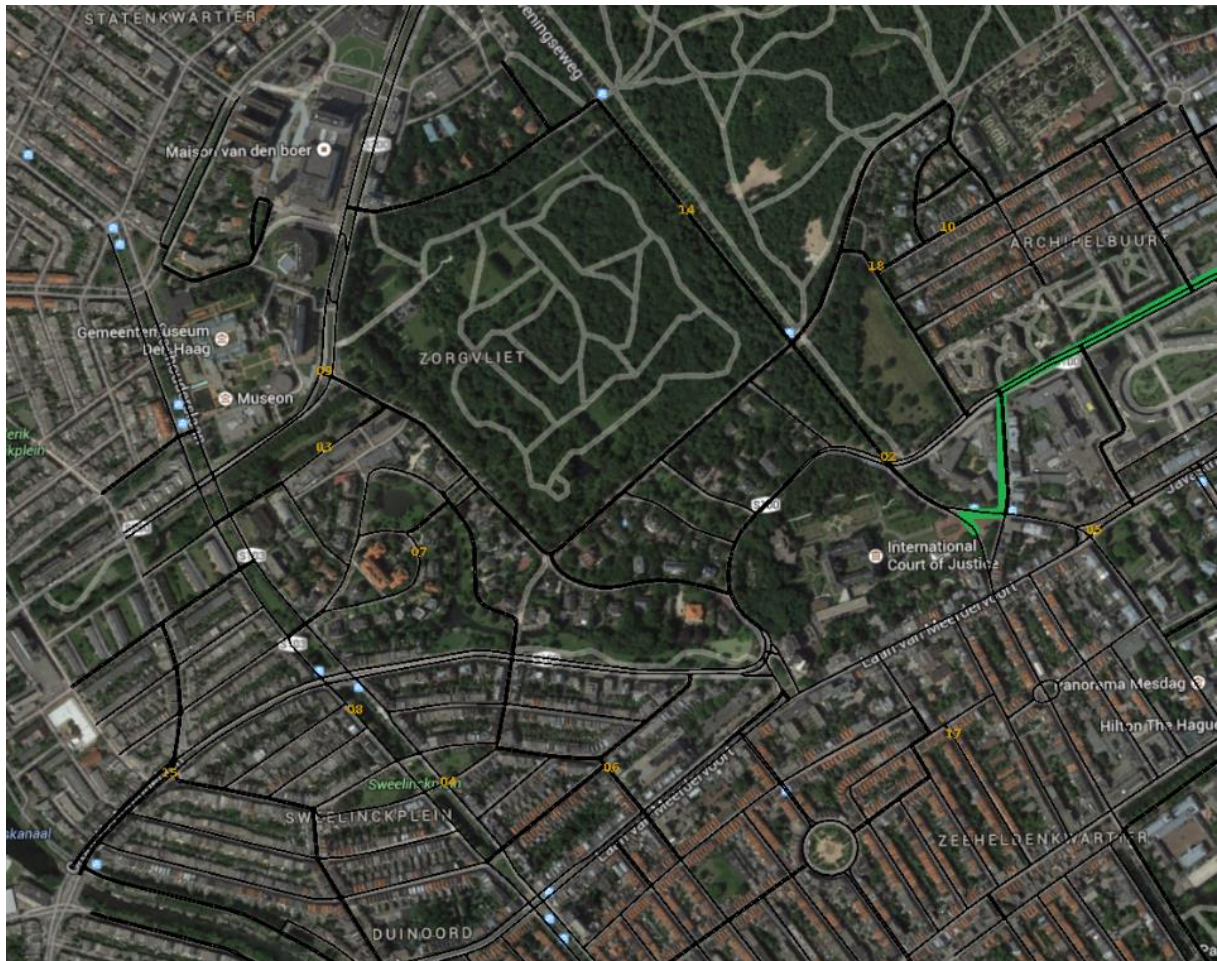
Gebruikt u medicijnen? Ja/ Nee (*Omcirkel a.u.b. het juiste antwoord*)

Zo ja, welke? en hoe vaak?

Handleiding computertest

Bedankt dat u mee wilt doen aan dit afstudeeronderzoek naar de beoordeling van situatie veiligheid. Uw deelname hieraan is voor het onderzoek dan ook erg belangrijk! Lees onderstaande omschrijving aandachtig door. Als u vragen heeft kunt u deze stellen aan de testleider.

U gaat zo dadelijk 4 tests op een PC maken, voorafgaand krijgt u eerst nog een oefenmoment. Het doel van de test is het vergaren van uw beoordeling over de situatie. Tijdens iedere test krijgt u een plattegrond te zien, zoals hieronder in de afbeelding.



De test bestaat uit het monitoren van verschillende objecten op het scherm. U kunt het zien als het in de gaten houden van een camera op een situatie van bovenaf. Op het scherm zult u verschillende voertuigen aanwezig zien, andere objecten zijn buiten beschouwing gelaten. De voertuigen kunnen stilstaan of zich met verschillende snelheden verplaatsen. Ieder voertuig kunt u herkennen aan een eigen nummer.

Het is aan u om ieder voertuig te gaan beoordelen om mate van dreiging. Dit doet u door ieder voertuig een cijfer te geven variërend van 1 tot 10. Waarbij 1 helemaal niet dreigend, 5 mogelijk dreigend en 10 zeker dreigend. Het gaat hierbij om uw eigen inzicht en er is geen goed of fout. Probeert u steeds alle voertuigen een cijfer te geven.

Iedere testsituatie duurt ongeveer 2 minuten. U krijgt steeds 6 maal een kort fragment te zien (van 10 seconden) waarna u gevraagd wordt alle genummerde objecten op het scherm te beoordelen aan de hand van bovenstaande opties. Na ieder kort fragment is een pauze waarop u gevraagd wordt het antwoordformulier voor dat moment in te vullen. Tijdens de pauze blijft het laatste moment zichtbaar en ziet u links de snelheden waarmee de voertuigen rijden. Uw beoordeling kunt u invullen op het antwoordformulier. De 6 fragmenten per testsituatie zijn opvolgend, net als een filmpje met pauzes erin. Daardoor kan het zijn dat uw mening over een bepaald object veranderd. Dit kunt u dan in de volgende pauze aangeven. Probeer achteraf niets meer te wijzigen het gaat erom wat u op het moment zelf dacht. Ieder fragment stopt vanzelf. Als u op de “Enter” toets op het toetsenbord klikt (nadat u het formulier heeft ingevuld) gaat de situatie weer verder. Er zijn steeds 2 soorten situaties die elkaar afwisselen, daardoor weet u wanneer een nieuwe situatie begint, dit wordt zo uitgelegd tijdens de testsituatie.

Voorbeeld:

U ziet 12 objecten op de camera. Geeft u hieronder uw mening in hoeverre u deze objecten helemaal niet dreigend vindt (cijfer 1) tot heel erg dreigend (cijfer 10).

<i>object</i>	<i>beoordeling</i>	<i>object</i>	<i>beoordeling</i>	<i>object</i>	<i>beoordeling</i>
01	1	06	3	11	4
02	2	07	7	12	2
03	1	08	8		
04	2	09	2		
05	2	10	5		

In het voorbeeld heeft ieder object een cijfer gegeven. De objecten die gewoon rijden krijgen lage cijfers 1 tot 3 (afhankelijk van hoe hard ze rijden en waar ze rijden). Object 08 rijdt met een snelheid van 70 waar 50 is toegestaan (dit zie je omdat alle andere voertuigen op die weg ook 50 rijden) daarom krijgt deze score 8. Object 07 begint vaart te maken en komt vanuit dezelfde richting als object 08 daarom krijgt deze score 7. *Hoe u de objecten beoordeeld is naar uw eigen inzicht. Op het antwoordformulier krijgt u altijd ruimte om een korte toelichting te geven.*

Er zullen nu eerst twee oefensituaties gegeven worden. Deze twee soorten situaties zullen elkaar in de echte test steeds afwisselen. De testleider blijft er even bij om te zien of alles goed werkt. Als u nog vragen heeft kunt u deze nu stellen.

Oefensituatie:

U ziet 12 objecten op de camera. Geeft u hieronder uw mening in hoeverre u deze objecten helemaal niet dreigend vindt (cijfer 1) tot heel erg dreigend (cijfer 10).

<i>Test 1 object</i>	<i>beoordeling</i>	<i>object</i>	<i>beoordeling</i>	<i>object</i>	<i>beoordeling</i>
01		06		11	
02		07		12	
03		08			
04		09			
05		10			
<i>Test 2 object</i>	<i>beoordeling</i>	<i>object</i>	<i>beoordeling</i>	<i>object</i>	<i>beoordeling</i>
01		06		11	
02		07		12	
03		08			
04		09			
05		10			

Saunabon

De saunabon is geldig tot 6 februari 2015 (m.u.v. 20 december 2014 t/m 4 januari 2015). Dit betreft een bon voor 2 personen bij één van de deelnemende sauna's van de Thermen & Beautygroup Nederland. Voor meer informatie kunt u contact opnemen met de betreffende sauna's. Vooraf reserveren is gewenst.

Als u kans wilt maken op de saunabon met uw deelname aan dit onderzoek, vul dan hieronder uw naam en e-mailadres in. Dit formulier kunt u losmaken van het boekje en inleveren in de blanco envelop.

Na afloop van het onderzoek wordt er blind een deelnemer getrokken en via het opgegeven emailadres benaderd. Overige proefpersonen worden niet meer benaderd en deze gegevens zullen verwijderd worden. Deelname is dus NIET gekoppeld aan uw overige testresultaten.

Naam: _____

Emailadres: _____

Antwoordformulier

- Onderzoek naar de beoordeling van de veiligheid van situaties –



In te vullen door testleider:

Proefpersoon nummer :

man/ vrouw :

Informed consent ondertekend :

Gelieve voor alle situaties alle objecten te beoordelen en een cijfer(1 t/m 10) toe te kennen. Anders kan uw deelname niet meegenomen worden in het onderzoek.

Testsituatie 1

Fragment 1:

<i>Object</i>	<i>Beoordeling</i>	<i>Object</i>	<i>Beoordeling</i>
01		11	
02		12	
03			
04			
05			
06			
07			
08			
09			
10			

Fragment 2:

<i>Object</i>	<i>Beoordeling</i>	<i>Object</i>	<i>Beoordeling</i>
01		11	
02		12	
03			
04			
05			
06			
07			
08			
09			
10			

Fragment 3:

<i>Object</i>	<i>Beoordeling</i>	<i>Object</i>	<i>Beoordeling</i>
01		11	
02		12	
03			
04			
05			
06			
07			
08			
09			
10			

Fragment 4:

<i>Object</i>	<i>Beoordeling</i>	<i>Object</i>	<i>Beoordeling</i>
01		11	
02		12	
03			
04			
05			
06			
07			
08			
09			
10			

Fragment 5:

<i>Object</i>	<i>Beoordeling</i>	<i>Object</i>	<i>Beoordeling</i>
01		11	
02		12	
03			
04			
05			
06			
07			
08			
09			
10			

Fragment 6:

<i>Object</i>	<i>Beoordeling</i>	<i>Object</i>	<i>Beoordeling</i>
01		11	
02		12	
03			
04			
05			
06			
07			
08			
09			
10			

Kunt u aangeven hoe lastig u deze testsituatie vond op een schaal van 1 (gemakkelijk) tot 10 (zeer moeilijk)?

Gemakkelijk 1 2 3 4 5 6 7 8 9 10 Zeer Moeilijk

Wat vond u gemakkelijk/ lastig aan deze situatie? _____

Opmerkingen _____

Testsituatie 2

Fragment 1:

<i>Object</i>	<i>Beoordeling</i>	<i>Object</i>	<i>Beoordeling</i>
01		11	
02		12	
03			
04			
05			
06			
07			
08			
09			
10			

Fragment 2:

<i>Object</i>	<i>Beoordeling</i>	<i>Object</i>	<i>Beoordeling</i>
01		11	
02		12	
03			
04			
05			
06			
07			
08			
09			
10			

Fragment 3:

<i>Object</i>	<i>Beoordeling</i>	<i>Object</i>	<i>Beoordeling</i>
01		11	
02		12	
03			
04			
05			
06			
07			
08			
09			
10			

Fragment 4:

<i>Object</i>	<i>Beoordeling</i>	<i>Object</i>	<i>Beoordeling</i>
01		11	
02		12	
03			
04			
05			
06			
07			
08			
09			
10			

Fragment 5:

<i>Object</i>	<i>Beoordeling</i>	<i>Object</i>	<i>Beoordeling</i>
01		11	
02		12	
03			
04			
05			
06			
07			
08			
09			
10			

Fragment 6:

<i>Object</i>	<i>Beoordeling</i>	<i>Object</i>	<i>Beoordeling</i>
01		11	
02		12	
03			
04			
05			
06			
07			
08			
09			
10			

Kunt u aangeven hoe lastig u deze testsituatie vond op een schaal van 1 (gemakkelijk) tot 10 (zeer moeilijk)?

Gemakkelijk 1 2 3 4 5 6 7 8 9 10 Zeer Moeilijk

Wat vond u gemakkelijk/ lastig aan deze situatie? _____

Opmerkingen _____

Testsituatie 3

Fragment 1:

<i>Object</i>	<i>Beoordeling</i>	<i>Object</i>	<i>Beoordeling</i>
01		11	
02		12	
03			
04			
05			
06			
07			
08			
09			
10			

Fragment 2:

<i>Object</i>	<i>Beoordeling</i>	<i>Object</i>	<i>Beoordeling</i>
01		11	
02		12	
03			
04			
05			
06			
07			
08			
09			
10			

Fragment 3:

<i>Object</i>	<i>Beoordeling</i>	<i>Object</i>	<i>Beoordeling</i>
01		11	
02		12	
03			
04			
05			
06			
07			
08			
09			
10			

Fragment 4:

<i>Object</i>	<i>Beoordeling</i>	<i>Object</i>	<i>Beoordeling</i>
01		11	
02		12	
03			
04			
05			
06			
07			
08			
09			
10			

Fragment 5:

<i>Object</i>	<i>Beoordeling</i>	<i>Object</i>	<i>Beoordeling</i>
01		11	
02		12	
03			
04			
05			
06			
07			
08			
09			
10			

Fragment 6:

<i>Object</i>	<i>Beoordeling</i>	<i>Object</i>	<i>Beoordeling</i>
01		11	
02		12	
03			
04			
05			
06			
07			
08			
09			
10			

Kunt u aangeven hoe lastig u deze testsituatie vond op een schaal van 1 (gemakkelijk) tot 10 (zeer moeilijk)?

Gemakkelijk 1 2 3 4 5 6 7 8 9 10 Zeer Moeilijk

Wat vond u gemakkelijk/ lastig aan deze situatie? _____

Opmerkingen _____

Testsituatie 4

Fragment 1:

<i>Object</i>	<i>Beoordeling</i>	<i>Object</i>	<i>Beoordeling</i>
01		11	
02		12	
03			
04			
05			
06			
07			
08			
09			
10			

Fragment 2:

<i>Object</i>	<i>Beoordeling</i>	<i>Object</i>	<i>Beoordeling</i>
01		11	
02		12	
03			
04			
05			
06			
07			
08			
09			
10			

Fragment 3:

<i>Object</i>	<i>Beoordeling</i>	<i>Object</i>	<i>Beoordeling</i>
01		11	
02		12	
03			
04			
05			
06			
07			
08			
09			
10			

Fragment 4:

<i>Object</i>	<i>Beoordeling</i>	<i>Object</i>	<i>Beoordeling</i>
01		11	
02		12	
03			
04			
05			
06			
07			
08			
09			
10			

Fragment 5:

<i>Object</i>	<i>Beoordeling</i>	<i>Object</i>	<i>Beoordeling</i>
01		11	
02		12	
03			
04			
05			
06			
07			
08			
09			
10			

Fragment 6:

<i>Object</i>	<i>Beoordeling</i>	<i>Object</i>	<i>Beoordeling</i>
01		11	
02		12	
03			
04			
05			
06			
07			
08			
09			
10			

Kunt u aangeven hoe lastig u deze testsituatie vond op een schaal van 1 (gemakkelijk) tot 10 (zeer moeilijk)?

Gemakkelijk 1 2 3 4 5 6 7 8 9 10 Zeer Moeilijk

Wat vond u gemakkelijk/ lastig aan deze situatie? _____

Opmerkingen _____

Bedankt voor uw deelname!

U kunt de formulieren inleveren bij de testleider.