A Supportive Situation Awareness Model for Human-Autonomy Teaming in Collaborative Driving

Rinta Kridalukmana [a], Hai Yan Lu [a] & Mohsen Naderpour [a]

[a]Centre for Artificial Intelligence (CAI), Faculty of Engineering and IT, University of Technology Sydney, Ultimo, NSW 2007, Australia

ARTICLE HISTORY
Compiled February 10, 2020

ABSTRACT
Driving has become a collaborative activity and a form of human-autonomy teaming (HAT) with the addition of autonomy to the advanced driver assistance system (ADAS), which makes situational decisions and sensible actions (e.g., autopilot and collision avoidance). However, it has been identified that in many fatal road accidents involving collaborative driving, over-reliance on the ADAS becomes the primary factor. To overcome this issue, the underlying situation awareness (SA) concept is investigated to identify an appropriate SA model for collaborative driving that could impact the intelligent agent's design in an HAT context. The formalization of existing SA model characteristics is defined and compared with those in collaborative driving. As a result, existing SA models are inadequate for explaining collaborative driving. Therefore, a new supportive SA (SSA) model is proposed. Based on the nature of this new model, applying transparency during SA development of the ADAS is suggested as a mechanism to comprehend ADAS behaviours. The proposed SA model is a significant expansion of multiple-agent SA models, and a transparent-based system can be a future direction of ADAS development to calibrate drivers' trust.

KEYWORDS
situation awareness; situation awareness model; supportive situation awareness model; collaborative driving; human-autonomy teaming

1. Introduction

Factors associated with drivers' situational awareness have been investigated by many researchers to provide useful features in advanced driver assistance systems (ADAS) with respect to safety problems. Situation awareness (SA) is defined as a state of knowledge needed to understand what is happening, which involves a human's cognitive ability to temporarily capture, update, integrate, prioritize and manipulate information from surrounding events that could require significant demands from a human's working memory (Johannsdottir and Herdman 2010). In the meantime, the dynamic states of SA held by human is reffered to as situational awareness in this study. The advancement of intelligent systems enables SA to be maintained not only by a human but also by a non-human agent (Stanton et al. 2017, 2006). The on-board ADAS in modern vehicles can be an example of such an intelligent system holding its own situ-

CONTACT Rinta Kridalukmana. Email: rinta.kridalukmana@student.uts.edu.au

ational awareness. Not only does the ADAS provide cognitive support for drivers, but it also has been granted more access to perform driving tasks, including taking over manual control as a backup in case of emergency. Thus, such an intelligent system in ADAS is transforming into an autonomous agent having a high level of autonomy (Endsley 2017). Using this perspective, driving becomes a multiple-agent-based activity involving two SA processes (from the driver and ADAS), which can be described as collaborative driving.

Previously, conventional driving has been a single-agent-based activity involving only one SA process from a driver to decide and control vehicle manoeuvres; therefore, this type of driving was modelled by Matthews et al. (2001) using an individual SA model. With such a model, the role of technological supports such as the ADAS in conjunction with drivers' situational awareness is limited to minimizing the secondary task effects and supporting drivers decision-making process by generating useful situational information or alerts, even though they might have to re-examine notifications delivered by the ADAS. Nevertheless, the individual SA model fails to explain that recent ADAS technologies have control abilities over vehicles based on its own artificial situation awareness (ASA). Therefore, it is necessary to use multiple-agent SA models to describe collaborative driving. In collaborative driving, human-autonomy teaming (HAT) is established, in which technologies can work together with humans as a teammate, not merely as automation tools (McNeese et al. 2018). Similar to a teammate, the ADAS has the ability to back the driver up, especially when the driver experiences SA development failures. For example, when high collision risk and inattentive driving are recognized, the ADAS can execute emergency actions to reduce speed or stop the vehicle as in a collision avoidance system. Additionally, the ADAS can reduce the driver's workload by temporarily replacing his/her driving tasks (as in autopilot features).

However, such collaborative driving raises other SA problems, leading to a mental model where a driver over-relies on the ADAS (Banks et al. 2018). With such a mental model, drivers tend to spend less time paying attention to the road, indicating that they are willing to take more risks while driving (i.e., they are more engaged in secondary tasks) (Körber et al. 2018). In other words, drivers become more vulnerable to distractions. Such over-reliance problems can also be viewed as fundamental coordination ability issues in HAT because humans fail to comprehend their non-human teammate behaviours (Freiman et al. 2018). As a result, many fatal road accidents involving collaborative driving occur because of this mental model. To address this problem, this study aims to examine the underlying SA concept so that a suitable SA model can be identified for collaborative driving. It is believed that the abilities of a non-human agent in the HAT context are designed based on the SA models representing the nature of the relationship among team members' goals, the environment, and SA requirements to form their SA.

The existence of shared and non-shared goals can be an example factor affecting an intelligent system's abilities for collaborative works. For instance, Demir et al. (2017) noticed that in HAT, where each member has non-shared goals, providing push notification/information abilities in a non-human agent will have a better impact on coordination performance than providing information crawling mechanisms for its human teammate. When there are both shared and non-shared goals, sharing knowledge ability is developed for the overlapping responsibilities that distributes information to each member as a point of coordination (Endsley 1995). Additionally, a simplified form of graphical display on the shared knowledge can also be helpful to enhance coordination (Chen et al. 2018). In the HAT context, such shared knowledge can be propagated as an input into the non-human agent.

The SA concept involving multiple agents in current studies can be grouped into three models including the team SA model Endsley (1995), mutual SA model (Shu and Furuta 2005), and shared SA model (Endsley 1995; Saner et al. 2009). This study formalizes the relationship characteristics of the goals, environment, and SA requirements of each existing SA model and compares them with those under collaborative driving. As a result, current SA models are insufficient to cater to team members' relation characteristics in collaborative driving mainly because they all have non-shared goals, which means that these current models reflect the horizontal coordination hierarchy.

In collaborative driving, the ADAS has a subset of driver goals reflecting the vertical coordination hierarchy and, therefore, a supportive SA (SSA) model is proposed. For this new model, providing transparency features of the non-human agent's decision-making process is recommended to improve HAT coordination (Chen et al. 2018). Transparency in this context refers to mechanisms used to provide understanding and supervision of the agent's behaviours, including the predictability of immediate future actions (Endsley 2017). Furthermore, the SSA methodology is also presented as an effort to enhance collaborative performance in SSA-based HAT by answering three critical questions as follows:

(1) How are SA requirements identified for the SA development of involved team members?
(2) What if the team members experience SA development problems?
(3) How is the information necessary to monitor the SA development of a non-human agent determined during its decision-making process?

The rest of this paper is structured as follows: Section 2 presents background and related works, which is followed by an introduction of the proposed SSA model in Section 3. In Sections 4 and 5, the application of the SSA model and conclusions with some future works are presented, respectively.

2. Background and Related Works

2.1. *Collaborative Driving, Automation, and Artificial SA*

The advanced technologies of modern vehicles currently allow different driving mode options: manual driving with situational information support, partial automation, or full automation (driverless). Based on the United States National Highway Traffic Safety Administration (NHTSA), the degree of automation in driving has been formally categorized into five levels, where level 0 represents manual driving and level 4 represents a fully driverless car that requires nothing from its occupants except in case of emergency and setting up a destination (Casner et al. 2016). NHTSA's level 1 refers to a function-specific automation operation in a car, such as a cruise-control system, to keep the vehicle running at a target speed. Furthermore, the vehicle's automated systems at level 2 and above allow non-involvement from the driver during loop control of driving tasks.

Generally, there are four aspects that can be automated from a task: monitoring or acquiring information, generating options to achieve goals, choosing an option to execute, and implementing actions with or without operator consent (Endsley 1999; Endsley and Jones 2011). Therefore, Stanton et al. (2006) argued that SA can be held by either human or non-human agents. Regarding a human agent, SA is about understanding what is happening (Jones 2014) which is described by Endsley (1995) using three-level stages of perception (SA level 1), comprehension (SA level 2), and projection (SA level 3). Regarding a non-human agent, the ability to provide mechanisms to project future actions based on prior knowledge, perception and projection is referred to as ASA (McAree and Chen 2013). The availability of driving modes at the NHTSA level 2 or above indicates that the ADAS in modern vehicles has some degree of SA.

From a driving perspective, SA is composed of multiple clusters of knowledge (see Figure 1), which consist of goal-driven knowledge, global driving environment and object interaction knowledge, spatial orientation knowledge, and vehicle status knowledge (Ma and Kaber 2005; Matthews et al. 2001; Pew 1994). In collaborative driving, knowledge about the ADAS as a vehicle's intelligent system is also required to avoid poor mental models when using such technology. Mental models are useful for guiding drivers' constant adaptations, which are required to maintain and regulate the driving situation status within the boundaries of tolerable and safe changes during the dynamic process of driving activities Bellet et al. (2009). Mental models provide human mechanisms to produce prototypical situations or the status of aided-system models during the perception process Endsley et al. (1997) to develop SA, which includes the descriptions of the purpose and function of aided systems and the way that the system knowledge is retrieved from long-term memory during interaction when necessary Endsley (2000).

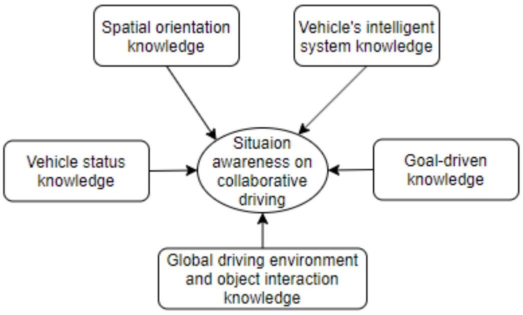

Spatial orientation knowledge
Vehicle's intelligent system knowledge
Vehicle status knowledge
Situaion awareness on collaborative driving
Goal-driven knowledge
Global driving environment and object interaction knowledge

Figure 1. SA-related knowledge on driving

2.2. *SA Model: Single Agent vs Multiple Agents*

To form SA, an agent (either a human or non-human agent) must first acquire SA elements (see Figure 2). SA elements are referred to as necessary, observable variables of objects used for inferring the current status and predicting the immediate future status of situations within an agent's goal-related environment. For example, a driver needs to pay attention to road signs when operating a vehicle. In this case, road signs can be an example of SA elements in a driving environment. Sometimes, agents need to combine several SA elements to comprehend the status of situations within the intended environment to project future actions. SA elements and their combinations, which are needed at every level of an agent's SA development mechanism, can then be referred to as SA requirements. The SA model is then used to explain the relationship of the agent's goals, goal-related environment and SA requirements.

As only one agent is involved to achieve goals in the problem domain, Figure 2 is referred to as a single agent or individual SA model. Conventional driving can be an example of an activity that can be described with an individual SA model, in which a

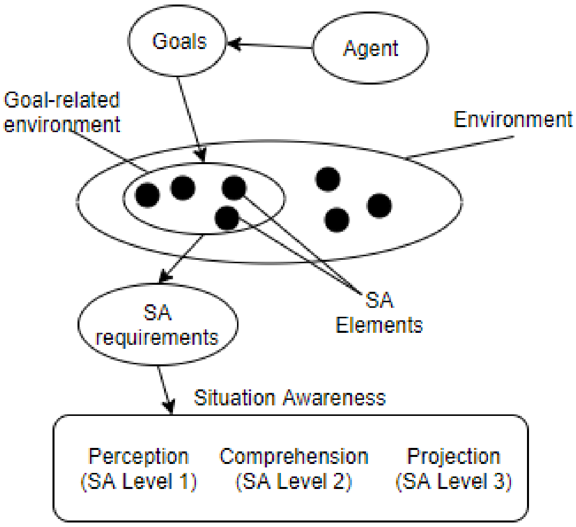


Figure 2. SA model: Relationship among goals, the environment, and SA requirements to form SA

driver is the agent. Fastenmeier and Gstalter (2007) described that, basically, driving has three main goals: navigating, controlling, and continuous monitoring. In this case, the driver is responsible for the entire driving goal and, therefore, the model produces the driver's responsibilities in the goal-related environment. As a consequence, all SA requirements for driving environments are intended for the driver. Moreover, all decisions that respond to given situations within the driving environment are only made by the driver. In other cases, it is possible that multiple agents work together with various relationships among their goals, environment, and SA requirements. However, the single agent SA model is the basis that forms SA for the involved agents.

Generally, the multiple-agent SA model is based on the perspective that the hierarchy of goals in a problem domain is distributed to more than one agent. Goals produce tasks and the environment of the responsibilities for each agent. For multiple agents, there are three variants of the SA model in the literature, which can be a team SA model, mutual SA model, or shared SA model. In a team SA model, each team member has his/her own SA requirements that are independent from each other because they have to deal with completely different environments within their responsibilities, as do their SA requirements (Endsley 1995). In a mutual SA model, the shared goals are described as a collaborative goal so that each member shares a common environment but produces different SA requirements for the team members, and each member needs to share what they know about the environment in a cooperative manner to achieve an understanding of the common environment (Shu and Furuta 2005). In a shared SA model, it is possible that at some points in time, team members share an environment because their goals overlap with each other and generate task redundancy. As a consequence, this model generates the same SA for the shared SA requirements (Endsley 1995; Saner et al. 2009). Such a shared understanding of a situation can also be referred to as point of coordination (Salas et al. 1995).

The multiple-agent SA model might appear in many problem domains, such as military domains. For example, 4 soldiers in an army's special force are sent to a district's battlefield. These soldiers are split up, in which one soldier goes to the west (soldier A), east (soldier B), south (soldier C), and north (soldier D), and they are responsible for analysing the enemy's strengths and weaknesses in those areas. In this case, each agent carries his/her own goals and addresses a different environment because they have different responsibilities. As they have completely different environments, the SA requirements for each agent are also independent of each other. This example, then, represents a team SA model.

In a different scenario, in which there is a hill located northwest of the district, soldiers A and D are responsible for finding water sources and identifying places that are suitable for the basecamp on that hill, respectively. In this case, preparing the basecamp is the overlapping goal between agents A and D, and the hill becomes the shared environment between these two agents. However, on that hill, soldiers A and D have different sub-goals that are independent from each other, which produces different SA requirements even though they have a shared environment from a shared goal. Then, soldiers A and D collaboratively share their knowledge to obtain a good understanding of the situation on the hill and determine a good spot for the basecamp. Such a relationship among members is referred to as mutual SA. In contrast, when both soldiers A and D are assigned to identify water resources and a suitable spot for the basecamp together, this can be identified as shared SA model. In shared SA situations, redundancy exists due to the shared goals and, therefore, it typically generates the same SA.

In the HAT context, the mutual SA model can exist in the aviation field. An aircraft feature called Emergency Autoland System (EAS) can be considered as a non-human agent. When both the pilot and co-pilot are disabled, a passenger can activate the EAS, and the EAS will contact the nearest airport listed in its database. After obtaining a response, at this point, the mutual SA model is established between EAS and the air traffic controller officer with the same goal, which is to land the aircraft safely. In this regard, the officer's task is to manage the air traffic for the landing process performed by EAS. Then, the teaming between an automated unmanned aerial vehicle (UAV) and an army troop can be an example of a shared SA model, in which both have the intersection goal to collect information from a battlefield. Nevertheless, the troop also has a different goal to obtain, i.e., custody, while the UAV must serve another troop. In the meantime, an example of a team SA model is demonstrated by an autonomous robot squad that is assigned to send supplies for a troop. In this regard, both the troop and the robot squad have different goals.

The existence of non-shared goals indicates that an agent has a horizontal coordination hierarchy with other agents. In the meantime, in a vertical coordination hierarchy, non-shared goals that belong to subordinate team members do not exist because this hierarchy is used to represent the relationship between a parent goal and the subset goals within the parent goal. While parent goals are usually owned by a superior agent, subset goals belong to subordinates of the superior agent. Subordinates do not have other goals and environments outside their superior's responsibilities. Based on this analogy, existing multi-agent SA models only cover the horizontal coordination hierarchy and cannot be used for vertical coordination hierarchies. Therefore, in the following sections, the new SA model is proposed and referred to as the SSA model.

3. The Supportive Situation Awareness Model

3.1. *Underlying Goal-driven SA Concept in Collaborative Driving*

Collaborative driving in this study holds the assumption that the ADAS has driving abilities for partially automated driving, in which a driver's involvement is still necessary in automatic mode (NHTSA level 3). In this section, the underlying SA concept in collaborative driving is explained by first reviewing the goals of human drivers and the ADAS as team members in the HAT context. By understanding team members goals, their goal-related environment and SA requirements can be identified. Therefore, this section presents two goal hierarchies associated with driving that consist of a hierarchy for controlling manoeuvres and one for monitoring situations. It is believed that these two goals could represent almost all aspects of the driving abilities of the ADAS.

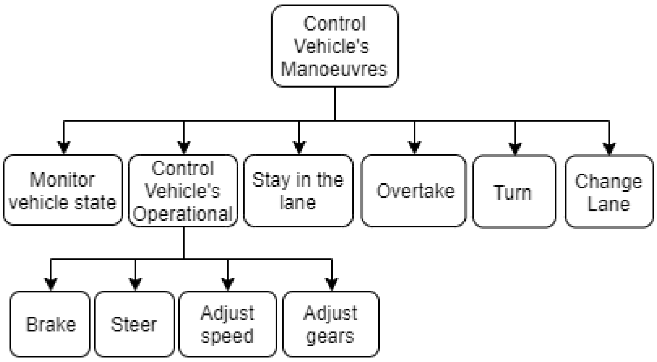


Figure 3. Goal hierarchy to control vehicle manoeuvres

In collaborative driving, only some driver's goals in conjunction with controlling vehicle's manoeuvres (see Figure 3) are delegated to the ADAS so that it only has limited control in performing driving manoeuvres. For example, the ADAS has no overtake manoeuvre abilities. Such overtake manoeuvres require other manoeuvres, such as lane changing. Even though the latest technologies of the ADAS enable lane changing, this task must be performed after obtaining permission from the driver to ensure the driver's acknowledgement of this manoeuvre. With such procedures, the self-performed overtaking of other vehicles by the ADAS would not be possible. The limited capability of a vehicle's sensory tools to sense what happens within environment is one factor that has caused ADAS to lack SA. Hence, the goal hierarchy to control vehicle manoeuvres is only partially delegated to ADAS. Thus, it can be inferred that when controlling a vehicle's manoeuvres, the ADAS has a subset of the drivers' goals.

Another explanation showing that the ADAS possesses the subset of driver's goals can be seen in the goal hierarchy to monitor driving situations (see Figure 4). In a collaborative driving context, the ADAS is embedded with the ability to detect object movement at all surrounding sides of the vehicle and assess the risks of situations, but it comes with a limited scope. For example, not all road signs and rules can be scanned directly from roads or acquired from traffic information providers. Moreover, not all objects' states within the environment can be inferred by the ADAS, mainly when predicting other road users' movements. In addition to observing situations on the road, collaborative driving also produces an additional goal to monitor one who is currently active in controlling the vehicle, which can be either the driver or the ADAS depending on the activated mode (i.e., manual or automatic driving mode). This goal

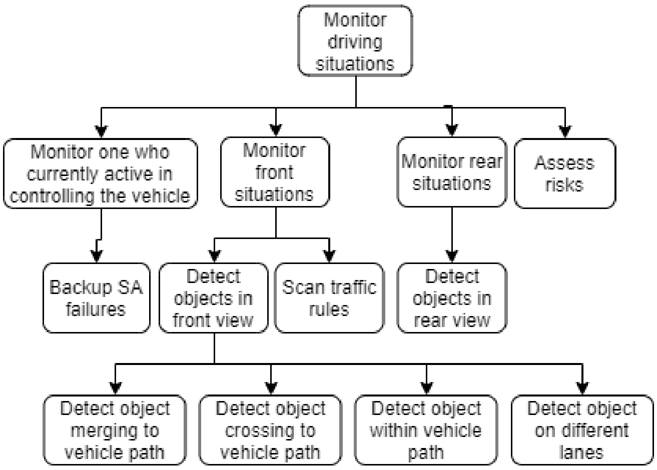

Monitor driving situations
Monitor one who currently active in controlling the vehicle
Monitor front situations
Monitor rear situations
Assess risks
Backup SA failures
Detect objects in front view
Scan traffic rules
Detect objects in rear view
Detect object merging to vehicle path
Detect object crossing to vehicle path
Detect object within vehicle path
Detect object on different lanes

Figure 4. Goal hierarchy to monitor driving situations

is applied for both the driver and ADAS mainly to back up their SA development failures due to their weaknesses and limitations. Therefore, in all driving modes, both agents are always actively involved. In this regard, both can take over control from each other as a backup in case of such SA failures.

By having the driver's goals subset, the ADAS also has a subset of the driver's driving environments. Within this shared subset environment, the same concern is placed on the SA elements, which then generates the same SA requirements for each agent. In other words, the SA requirements of the system also follow a subset relationship. However, as the ADAS is a non-human agent, it might have different mechanisms or sub-goals compared to the driver in achieving its goals. For example, a driver may monitor the ADAS behaviour from its manoeuvre outcomes or be alerted. In the meantime, the ADAS uses different methods (i.e., observing the driver's initiative), which can be achieved through acquiring samples of eye gaze or hand-on-steering wheel recognition. However, when assessing frontal collision risks, these methods do the same thing by estimating the distance to the vehicle ahead even though they have different ways to acquire the distance value due to their differences in sensing mechanisms. Therefore, the sub-goals of the driver and ADAS within the delegated goals can overlap each other.

3.2. *The Concept of SSA Model*

This section introduces the SSA model within the HAT context by generalizing the underlying SA concept in collaborative driving. Therefore, the driver and ADAS are described as a human agent (HA) and autonomy agent (AA), respectively. To better understand the proposed SSA model, this study uses a two-agent-based relation that consists of one HA and AA, in which the AA is inferior to the HA. Furthermore, several related definitions are defined as follows:

(1) **Agent**. An agent is a party that has a mechanism to form SA based on determined goals, and it can be a human or non-human agent.
(2) **Human Agent (HA)**. A human agent is the term used to represent an individual as an agent.
(3) **Autonomy Agent (AA)**. An autonomy agent is the term used to represent

an intelligent system or technology as a non-human agent with a high level of autonomy, which is authorized to make decisions and perform actions based on its artificial SA.

(4) **Goal**. A goal is something to achieve within a domain of problems.

(5) **Tasks**. Tasks are a set of activities to achieve determined goals.

(6) **Environment**. Starting in this section, a goal-related environment is referred to as the environment only, which has goal-related objects that are either physical or abstract, by which human or non-human agents are surrounded depending on their responsibilities in a domain of problems.

(7) **Supreme agent (SPR)**. A supreme agent is an HA with goals to be achieved and some of these goals are distributed as a subset goal to his/her subordinate agent.

(8) **Subordinate agent (SBA)**. The subordinate agent is an AA that acts as the subordinate of the SPR and has a subset goal of the SPR.

(9) **Situation**. A situation can be described as a state of the environment, which is a set of state values of the involved SA elements.

An agent's goals in a domain of problems are critical for identifying the SA elements in an environment so that the status of the surrounding situation's associated goals can be measured or predicted. To achieve these goals, an agent might receive support from another agent, as reflected in the relation between the SPR and SBA, which have vertical coordination hierarchies. As a subordinate and a non-human agent, the SBA owns some hierarchal goals of the SPR even though the SBA could have different sub-goals due to different mechanisms in, i.e., acquiring information from the environment or monitoring of teammates. However, these sub-goals are still under the SPR's goal hierarchy as the parent goal. Therefore, the SBA views the subset environment of the SPR. Within this subset environment, the SBA has the same SA requirements as the SPR for the same sub-goals, but it is not the same if different sub-goals exist. Such a relationship among goals, the environment, and SA requirements is referred to as the SSA model.

3.3. *Formalization of the SSA Models*

To highlight the distinction between existing SA models and the proposed SSA model, formalizing the relation among the two agents goals, environments, and SA requirements is presented in Table 1, in which the involved agents are HA and AA, respectively. In the SSA model, AA represents the SBA and is a subordinate of HA, which represents the SPR. Regarding the goals relations, in the team SA model, HA and AA have independent goals from each other and, therefore, the intersection of goals between HA (GHA) and AA (GAA) results in an empty set. In contrast, in the mutual and shared SA model, GHA intersecting GAA produces a non-empty set, but GHA and GAA are not equal to GHA intersecting GAA, which reflects the existence of non-shared goals in GHA and GAA. In the meantime, GHA intersecting GAA in SSA generates GAA, reflecting GAA as the subset form of GHA. Moreover, the environments of HA (EHA) and AA (EAA) also have the same intersection characteristics as their intersecting goals.

Within intersecting goals (i.e., shared goals), sub-goals may exist for HA (SGHA) and AA (SGAA), except in the team SA model, which has no intersecting goal. In mutual SA, the joint function between SGHA and SGAA produces overall shared goals, but their intersection results in an empty set. This result means that SGHA and SGAA are different sets of sub-goals. However, in the shared SA, SGHA and SGAA are equal to each other and to shared goals. In the meantime, SGHA and SGAA in SSA may be partially different as AA can have additional sub-goals (ASGAA) in addition to having a delegated goal hierarchy from SPR. ASGAA might not exist in the SPR goals hierarchy, yet its existence is to support the delegated goals. As a consequence, the intersection of the SA requirements of HA (SRHA) and AA (SRAA) for the team and mutual SA is not an empty set.

Table 1. Formalization of Existing Multiple-Agent SA Models and SSA Models Describing the Relation Among an Agent's Goals, Environment, and SA Requirements in a Two-Agent-Based System

Description	Team SA	Mutual SA	Shared SA	Supportive SA
Involved Agent: Agent 1 (HA) and Agent 2 (AA)	HA and AA	HA and AA	HA and AA	HA and AA (AA is SBA of HA)
Relation between Agent 1's goals (GHA) and Agent 2's goals (GAA)	$GHA \cap GAA = \emptyset$	$GHA \cap GAA \neq \emptyset$, where GHA, $GAA \neq GHA \cap GAA$	$GHA \cap GAA \neq \emptyset$, where GHA, $GAA \neq GHA \cap GAA$	$GHA \cap GAA = GAA$
Relation between the Environments (E) of HA (EHA) and EAA	$EHA \cap EAA = \emptyset$	$EHA \cap EAA \neq \emptyset$, where EHA, $EAA \neq EHA \cap EAA$	$EHA \cap EAA \neq \emptyset$, where EHA, $EAA \neq EHA \cap EAA$	$EHA \cap EAA = EAA$
Sub-goals (SG) and additional sub-goals (ASG) within intersecting goals (shared goals)	$\emptyset$	$SGHA \cup SGAA = GHA \cap GAA$, where $SGHA \cap SGAA = \emptyset$	$SGHA = SGAA = GHA \cap GAA$	$SGHA \cap SGAA = SGAA \setminus ASGAA$, where $SGAA \cup ASGAA = GAA$
SA requirements (SR) of HA and AA for shared goals	$\emptyset$	$SRHA \cap SRAA = \emptyset$	$SRHA = SRAA$	$SRHA \cap SRAA \neq \emptyset$

SSA, then, is expressed by 9 tuples to provide its formal description, SSA = {*SPR, SBA, GSPR, SPRE, GSBA, GSBA', SBAE, SRD,* and *SRA*} where SPR represents a supreme agent and SBA represents a subordinate agent. Furthermore, *SPRE* and *SBAE* represent the environments of *SPR* and *SBA*, respectively. The necessary SA elements from the environment depend on the agent's goals in a domain of problems. Therefore, within the environment, each agent has goals to be achieved. Regarding *SPR*, $E \rightarrow SPRE = \{o_1, o_2, ..., o_n\}$, where E denotes the hierarchy of identified goals that belong to *SPR* within *SPRE* and o represents a set of objects (SA elements) within the *SPRE*. Then, let *GSPR*:

$$GSPR = \bigcup_{i=1}^{n} GSPR^{(i)} = GSPR^{0} \cup \bigcup_{i=1}^{n} \{gspr_{1}^{(i)}, ..., gspr_{ji}^{(i)}\}$$

represents the structure of the identified goals of the *SPR*, where $GSPR^{(i)}$ represents a set of goals at level i and $gspr_{ji}^{(i)}$ represents the j-th goal at level i (see Figure 5) (Ma et al. 2010). Next, let $GSPR^{0} = SPR$ indicate that *SPR* owns the goals. To describe the parent-child relationship between two goals, we use $gspr_{j/i}$ to denote the j-th sub-goal of the goal $gspr_i$. The notation $gspr_j(gspr_{ji/j}, ..., gspr_{jm/j})$ is used to show that a goal $gspr_j$ is supported by a set of sub-goals $gspr_{ji/j}, ..., gspr_{jm/j}$, and $gspr_{ji} \prec gspr_j$ is used to indicate that $gspr_{ji/j}$ is a sub-goal of $gspr_j$.

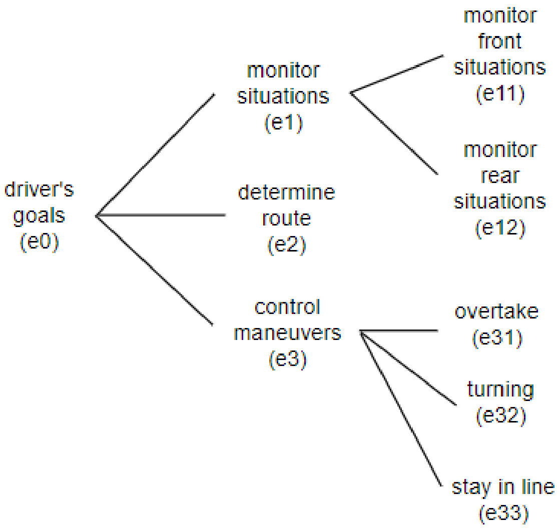


Figure 5. An example of the goal hierarchy

Furthermore, $GSBA \to SBAE = \{o_1, o_2, ..., o_n\}$, where $GSBA$ represents the hierarchy of delegated goals from the SPR that belong to the SBA within $SBAE$ and o represents a set of objects within SBA's environment. $GSBA$ also has the same structure as $GSPR$, therefore let GSBA:

$$GSBA = \bigcup_{i=1}^{n} GSBA^{(i)} = GSBA^{0} \cup \bigcup_{i=1}^{n} \{gsba_1^{(i)}, ..., gsba_{ji}^{(i)}\}$$

in which $GSBA^0 = SBA$ indicates that the goal hierarchy belongs to the SBA. Furthermore, $GSBA'$ represents an additional sub-goal hierarchy of $GSBA$.

The SA requirements for the SPR and SBA agents are represented by SRD and SRA, respectively, and denoted as follows: $SRD = (GSPR \times T)$, $SRA = ((GSBA + GSBA')) \times O)$, where $SRA \cap SRD \neq \emptyset$. Then, $T = ((L1 \cup L2 \cup L3) \times \{q1, ..., qx\})$. In this regard, T and O represent a set of SA requirements (q) for three levels of SA $(L1, L2, L3)$ for SPR and SBA, respectively, and x represents the number of requirements.

3.4. *SSA Methodology*

In this SSA methodology section, the three research questions to enhance SSA-based HAT performance are covered. However, before answering the questions, the SSA methodology needs to first develop the SPR and SBA's goals hierarchy by taking advantage of the hierarchical task analysis (HTA) method. The goals hierarchy, then, is modelled by using a use case diagram. After the goal hierarchy is set, the SSA methodology goes into the second step, which answers the first two questions about how to identify SA requirements to support team members' SA development and what if the team members experience SA development problems. In this second step, the extended goal-directed task analysis tool is used, given the goal hierarchy from previous step as the input. In the final step, this paper proposes a time-constraints-driven transparency framework to address the third question about determining transparency requirements to monitor SA development of an AA, so that its state and behaviour can be comprehended by the human teammate.

3.4.1. Modelling Goals of the SPR and SBA

Soliman and Mathna (2009) explained that SA is a principal cognitive construct associated with multitasking situations in which individuals should manage physical and cognitive workloads across tasks that often conflict among task demands and goals. Therefore, to obtain good SA requirements, mapping the goal hierarchy is important. Hence, to develop the goals hierarchy, this paper applies the HTA method, which infers the goals hierarchy from the functions hierarchy. In this context, a function can be described as a set of the agent's tasks in a system that serves a particular goal when performing teamwork with humans. Stanton NA (2006) explained that the main principle of HTA is to describe goals for functions such that each function is described in terms of its goals. Moreover, HTA is a means to break down sub-functions in a hierarchy, and as a consequence, these sub-functions are described in term of sub-goals.

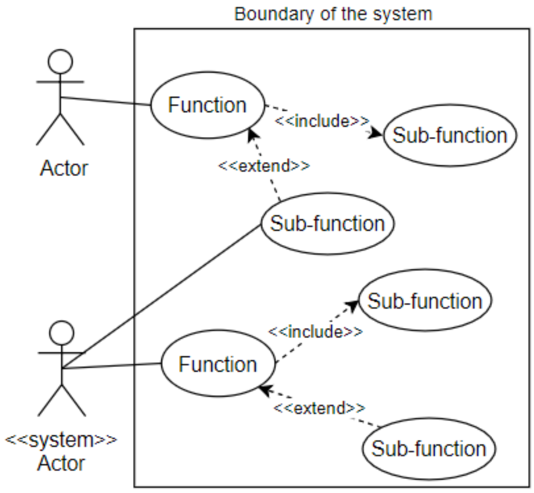


Figure 6. Example of use case diagrams

In software development, one technique that can be used to model the relation between functions is a use case diagram (see Figure 6), which was introduced by Jacobson (1993). In this technique, a use case that represents a function could have three types of relationships: a generalized relationship (GR), a normal relationship (NR) and an extended relationship (ER) (Chanda et al. 2009). A GR is intended to relate functions and their sub-functions. An NR between two use cases means that the behaviour of a use case is explicitly incorporated by another use case incorporating it (Shen and Liu 2003). When the behaviour is extended to another use case in the case of a special event, it can be described as an ER. Furthermore, an agent performing functions or sub-functions in a use case diagram is referred to as an actor, which can be either a human or non-human agent. A line is used to connect actors to functions or sub-functions.

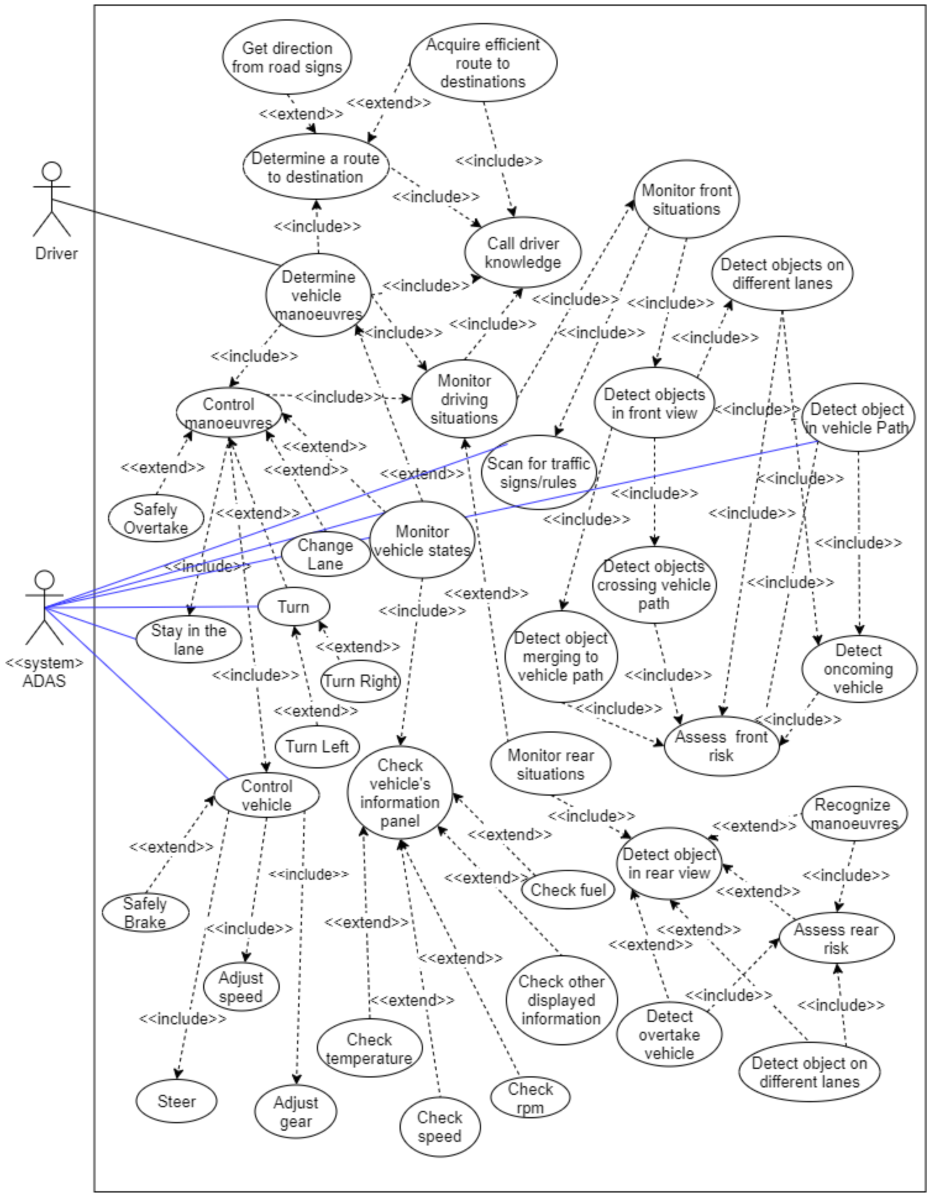

Get direction from road signs
Acquire efficient route to destinations
Determine a route to destination
Call driver knowledge
Monitor front situations
Detect objects on different lanes
Driver
Determine vehicle manoeuvres
Monitor driving situations
Detect objects in front view
Detect object in vehicle Path
Control manoeuvres
Scan for traffic signs/rules
Safely Overtake
Change Lane
Monitor vehicle states
Detect objects crossing vehicle path
Turn
Stay in the lane
<<system>> ADAS
Turn Right
Detect object merging to vehicle path
Detect oncoming vehicle
Turn Left
Assess front risk
Control vehicle
Check vehicle's information panel
Monitor rear situations
Recognize manoeuvres
Detect object in rear view
Check fuel
Safely Brake
Assess rear risk
Adjust speed
Check other displayed information
Detect overtake vehicle
Check temperature
Detect object on different lanes
Steer
Adjust gear
Check speed
Check rpm
<<include>>
<<extend>>

Figure 7. Relationships between goals and sub-goals in driving with a use case diagram

When developing an HAT scenario in the SSA context, it is critical for a system designer of SBA to develop SPR's goal hierarchy such that it can cover SPR's overall responsibilities and functions. This is important to determine into which level the goal hierarchy will be detailed. After that, the system designer will specify which goals/sub-goals will be supported by SBA. As an example, we develop a use case diagram to model a driver's goal hierarchy (see Figure 7). Referring to the proposed SSA model, Figure 7 shows how the formal description is applied to characterize the relationship between SPR's goals and SBA's goals which follows equation $GHA \cap GAA = GAA$. First, a driver should decide a route with an intended destination and use his knowledge to make such a decision. Once the route is set, the driver determines the vehicle's manoeuvres following the specified route. In other words, determining a vehicle's manoeuvres is triggered by the driver's decision on the route given a destination, which can be described as an ER. When the driver does not have a complete understanding about which route to take to the destination, obtaining directions from road signs can be conducted. Additionally, when the driver knows several route options, the most efficient route can be selected.

To determine the vehicle's manoeuvres, a driver must incorporate some sub-goals, such as using their knowledge, controlling manoeuvres, and monitoring driving situations. A driver's knowledge in this context is one that is associated with how to make a proper response given a driving situation. To stay in a lane is the basic manoeuvres in driving and, therefore, it is incorporated in the goal to control a vehicle's manoeuvres. In the meantime, other goals, such as safe overtaking, lane changing, turning, and monitoring vehicle states, are optional depending on many factors, such as the chosen route or traffic situations. Hence, they have an ER with their parent goal. Another sub-goal of controlling the vehicle's manoeuvres is to control the vehicle's operations, which incorporates steering, adjusting the vehicle's speed, and adjusting the vehicle's gears, but it has an ER with a braking function.

Furthermore, to delegate the SPR goals into the SBA, other actors can be added to use a case diagram that acts as the SBA. In our case of collaborative driving, the new actor for the SBA is the ADAS. After that, among the driving goal hierarchy that has been identified for the SPR, the actors could be delegated partially to the SBA. This step can be done in the diagram simply by connecting the actor symbol representing the SBA to the intended use case logics by using a line. However, it should be noted that in Figure 7, the SPR is connected to only one use case logic which can be considered the root goal (*to determine vehicle manoeuvres*). In the model, when a line is drawn to connect an actor to a goal, it means that this actor is also responsible for its sub-goals. As a consequence, the model in Figure 7 indicates that the SPR holds responsibilities for all goals/sub-goals in the hierarchy. Meanwhile, SBR is linked to only six sub-goals: *control vehicle*, *stay in the lane*, *turn*, *change lane*, *scan for traffic signs/rules*, *detect object in vehicle path*, and *monitor vehicle states*. This means that the boundary of SBR's responsibilities is limited to these six sub-goals along with their sub-ordinate goals.

For example, the use case logic representing *to control manoeuvres* has six subordinate goals, yet only five of them were delegated to the SBR (*control vehicle*, *stay in the lane*, *turn*, and *change lane*, *monitor vehicle state*); the *overtake* function was not included. Therefore, regarding *control manoeuvres*, five lines were drawn to connect the SBR as an actor to those five use case logics delegated to the SBR. However, by connecting the SBR to a use case logic representing *control vehicle*, the SBR automatically assigned all sub-ordinate goals under it (*safely brake*, *steer*, *adjust speed*, and *adjust gear*).

Moreover, other sub-goals reflecting additional sub-goals can be added to support the parent functions. As presented in Figure 8, when trying to obtain an efficient route to the destinations, a driver could obtain additional support from the ADAS. Previously, a driver relied only on his/her own knowledge regarding the most efficient route to the destination. Then, the ADAS can perform a set of sub-goals to acquire the travel time, current traffic condition, and travel distance for each route. The most efficient route information can be generated for drivers to help them determine the desired route to the destination. These additional sub-goals refer to the formal descriptor written in Table 1 defining that $SGHA \cap SGAA = SGAA \setminus ASGAA$, where $SGAA \cup ASGAA = GAA$. In this regard, the additional sub-goals of SBA represented

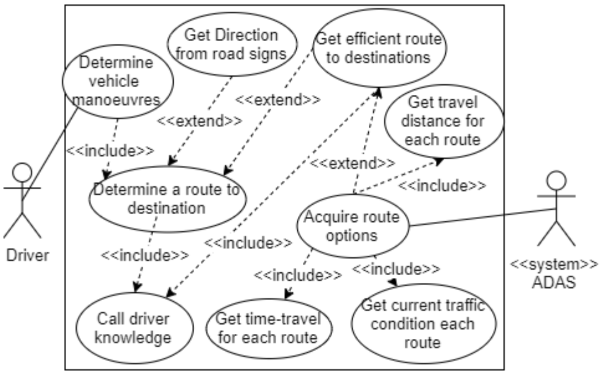

Get Direction from road signs
Get efficient route to destinations
Determine vehicle manoeuvres
<<extend>>
<<extend>>
Get travel distance for each route
<<include>>
<<extend>>
<<include>>
Determine a route to destination
Acquire route options
Driver
<<include>>
<<include>>
<<system>> ADAS
<<include>>
<<include>>
Call driver knowledge
Get time-travel for each route
Get current traffic condition each route

Figure 8. Goals to determine a route with the involvement of the ADAS (SBA)

by ASGAA may not exist as sub-goals of SPR (denoted by SGHA) because the human does not have such capabilities. However, ASGAA exists as a part of SBA's sub-goals (SGAA).

3.4.2. Identify SA Requirements and SA Failure Handlers

The goals modelled in the previous step provide us a boundary within an environment and, therefore, only environment variables within this boundary can be considered as SA elements, thus requirements to form SA. As described in Table 1, SSA-based HAT considers that SBA's SA requirements at a certain level of the goal hierarchy will be the same as SPR's. Nevertheless, SBA can have additional sub-goals along with SA requirements for these sub-goals. As SA requirements are very critical in the SA development process, how to identify SA requirements of team members is posted as the first research question in this paper. To address this question, we can take advantage of an analytical tool called the goal-directed task analysis (GDTA) by Endsley (2001). In Figure 9, the GDTA format is presented and based on this format, it can be seen that GDTA requires the goals hierarchy as the input for the analysis process.

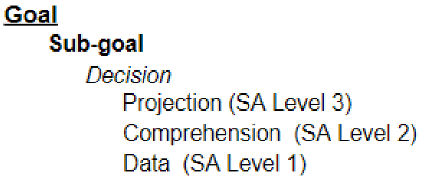

Goal
Sub-goal
Decision
Projection (SA Level 3)
Comprehension (SA Level 2)
Data (SA Level 1)

Figure 9. Goal-directed task analysis format (Endsley 2001)

The GDTA processes establish the link between human cognition and human action by first focusing on the required information rather than on how the information can be collected, and second, by integrating the information to comprehend and project the situation to develop SA. Additionally, the results of the GDTA process include necessary information to acquire benefits from technological supports, such as the status of current situations, hazardous situation alerts, suggestions, or performance feedback. However, it should be noted that in the SSA-based HAT context, a team-mate can also be considered as an environment variable. In this regard, GDTA is not sufficient to analyse the two factors related to the possible SA failures of team members

Table 2. eGDTA on Keeping the Vehicle in the Lane and Detecting Objects in the Vehicle Path

Parent Goals	Sub-Goals	SA Level 1	SA Level 2	SA Level 3
Control Manoeuvres				
	Stay in the lane	Road lines, vehicle's position in the lane	Vehicle's position status between lines	
		Possible SA failures: SPR: Inattentive driving SBA: Difficult to detect road lines	**Possible SA failures:** SPR: Unconsciously driving the vehicle outside of the lane, vehicle is over the line for a long time because of pre-overtaking manoeuvres and over-reliance on the SBA when performing this function SBA: Violations due to data availability problems	
		Failure handlers: H1: Sending inattentive driving alerts H2: Informing the driver that the lines are unreadable	**Failure handlers:** H1: SBA sends inattentive driving and lane departure warnings H2: Notify driver to interfere the vehicle's control	
Detect an object in front				
	Detect object in the vehicle path	Objects in front in the vehicle path, distance, vehicle speed, knowledge about safe margin	Safe margin level with the vehicle ahead	
		Possible SA failures: SPR: Inattentive driving, observation failure SBA: Observation failures due to curvature of the road	**Possible SA failures:** SPR: Fails to maintain safe margin at a given speed, over-reliance on the SBA to maintain the safe margin without collision risk, habitual problem of driving the vehicle near the vehicle ahead SBA: data availability problems	
		Failure handlers: H1: SBA sends inattentive driving alerts H2: Sends inattentive driving alerts, notifies driver to interfere with the vehicle's control	**Failure handlers:** H1: SBA sends alerts, executes emergency brakes H2: Sends inattentive driving alerts, notifies driver to interfere with the vehicle's control	

and SA failure handlers to mitigate the effects of the team members' SA development problems. Mitigating such effects is the problem of the concern posted in the second research question.

Therefore, the extended-GDTA (eGDTA) technique is applied in this paper to address the first two research questions. This technique will cover how to identify SA requirements, possible SA failures, and possible SA failure handlers of all involved agents at all SA levels. In our case of the collaborative driving context, SA failure handlers consist of two parts: H1 and H2. In H1, it is assumed that the driver as the SPR is in charge of vehicle control; therefore, failure handlers are provided by the SBA to help a driver in his/her SA failures. In H2, it is assumed that the SBA is controlling the vehicle and the SA failure handlers will focus on how the driver would retreat back into the control loop to handle the situations.

Based on the developed model illustrated in Figure 7, this section provides two examples of how eGDTA is used on two goals comprising keeping the vehicle in the lane and detecting objects in the vehicle path. In Table 2, it can be seen that for the first example, road lines and the vehicle's position in the lane become critical elements during SA level 1 development. However, at this level, the situational awareness failures of the SPR and SBA are mostly caused by inattentive driving and difficulties in detecting road lines, respectively. Therefore, sending inattentive driving and unreadable line warnings to the driver are suggested as the failure handlers for such problems.

Regarding SA level 2 development, it would be very helpful, particularly for the SBA, if the vehicle's position status between the lines can be obtained, as such information is the core input to ensure that the vehicle stays in path. It often occurs that a driver unconsciously drives the vehicle out of the lane. Moreover, the driver sometimes makes an overtaking preparation by taking the vehicle over the road line, but he/she forgets to go back to a safe position between the lines when the overtaking manoeuvre is cancelled. Such unconscious problems, then, can be considered as SA failures at the comprehension level in addition to SPR's over-reliance mental model problem. Sending inattentive driving alerts and lane departure warnings and notifying drivers to interfere with the vehicle's control can be an option for H1 and H2, respectively, to overcome such SA level 2 failures.

The second example for eGDTA is depicted from the goals related to detecting objects in the vehicle path. For the SBA, the core module to perform this function is its ability to recognize objects. Once the object can be recognized, the distance to this object can be measured. After that, by combining the distance information with the knowledge about the safe margin and current vehicle speed, the safe margin level can be calculated to help the SPR and SBA in the SA level 2 development process. As presented in Table 2, SA level 1 failures related to the given goal mostly come from inattentive driving and observation failures. In the meantime, some examples of SA level 2 failures are a driver who often fails to maintain a safe margin at a given speed, the driver's habit to drive very close behind the vehicle ahead, and over-reliance on SBA to keep the safe margin. In this regard, inattentive driving alerts, loop control requests, and emergency braking procedures are the critical handlers associated with the goal of detecting objects in a vehicle path.

3.4.3. Identify Transparency Requirements

In the SSA-based HAT context, understanding the SBA's behaviours can be considered as one critical factor to enhance collaboration performance and improve the SPR trust on SBA. In this regard, it is necessary to make SBA's states and behaviours transparent during its decision-making process. The mechanism to make the non-human agent transparent is widely known by the term of transparency. However, as posted in the third research question, the kind of information regarding transparency to help SPR comprehend SBA becomes one problem that needs to be determined.

Previously, Chen et al. (2018) proposed the situation-awareness-based transparency (SAT) framework to deal with transparency requirements. There are three levels of transparency in the SAT framework. The first level is associated with the SBA's goals, which include its current status, action, and plans. Related information at this level includes goal selection, progress, performance, and environment/teammate, such as a driver's distraction status and the status of collision risk with the vehicle ahead. At the next level, transparency information is related to the SBA's reasoning process and any environmental constraint information (e.g., delivering a notification to the driver when it fails to recognize the traffic light state). Finally, level 3 is about the SBA's projection of future outcomes, including the likelihood of success or failure and performance.

Nevertheless, there is a critical factor in providing transparency information due to the human ability to absorb information, particularly in a short time length. In fact, in a case such as driving, situations can have various time intervals, so that for a certain situation with a short duration, it becomes difficult to absorb all information in SAT level 1. Therefore, as an effort to answer the third research question, this paper introduced the time-constraint-driven transparency (TCDT) framework that considers time constraints in determining transparency requirements.

As presented in Figure 10, the TCDT framework has two axes in which the x-axis represents the priority of transparency themes that needs to be conveyed against given time constraints represented by y-axis. The framework divides transparency themes into four groups: explanation and performance; plans, decision, and coordination; intention; and outcome prediction. In the meantime, blue lines indicate the visibility of each transparency theme on each time constraint group. When a situation has a strong time constraint, i.e., entering traffic light situations, four transparency themes are suggested, including the non-human agent's decision-projection (DP), likelihood of task failure (LoF), perception status of surrounding focal objects (FOS), and task intention (TI). These four transparency themes are also recommended to be visible at all times as the default themes.

In the case of traffic light situations, the example information that belongs to DP is 'stopping' and 'keep moving' depending on the detected traffic light state (i.e., red, green). In this regard, traffic light states can be considered as the example of FOS. When SBA fails to recognize the traffic light state, there is a possibility for the SBA to react properly based on the lead vehicle ahead, particularly when the actual state is red. With no vehicle ahead detected within the range, the SBA by design will make the keep going manoeuvre when the TL state remains unknown and a potential road incident exists in such a situation, i.e., violating a red light and colliding with other vehicles. Such a decision design can be considered as a trade-off between safety and other road users convenience because if the SBA were to stop every time the traffic light state is unrecognized, it will cause problems to other drivers. Therefore, it is necessary to deliver LoF and TI to the driver to mitigate the effect of the design trade-off.

Furthermore, when the human has more spare time because the situations have medium or low time constraints, other transparency themes can be delivered to help comprehending the non-human agent. In TCDT framework, transparency themes for medium constraints include perception of non-focal surrounding objects (EP), agent's

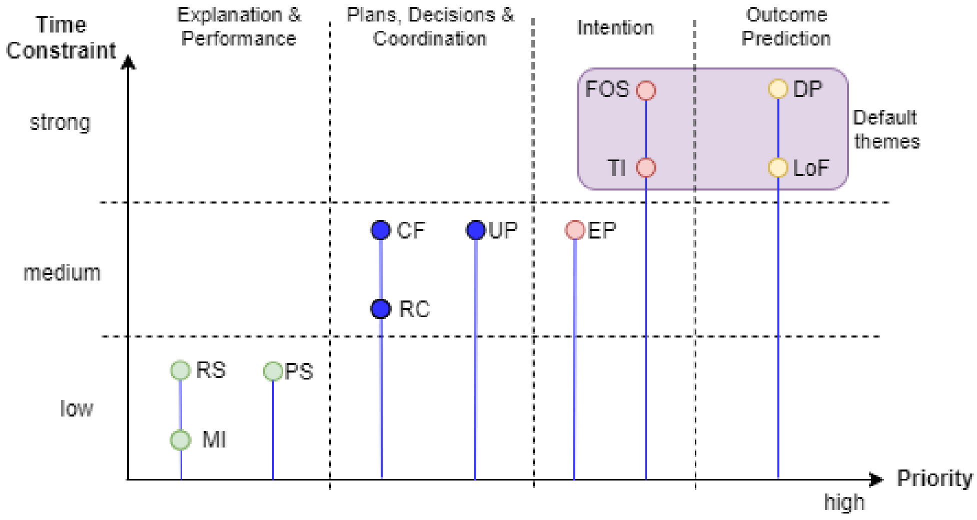


Figure 10. Time-constraint-driven transparency framework

plans and their updates (UP), decision options/recommendations (RC), and confirmations (CF). Then, for low time constraints, transparency themes are focused on process states (PS), reasoning states (RS), and any miscellaneous information (MI). The low time constraint group is intended to provide the detailed explanation about the agent's activities in conjunction with current states recognition of objects/situations through PS and RS. When an agent has a complex system consisting of numerous recognition processes, it would be a challenge to provide absorbable information given such complexities. In the meantime, MI is used to convey any other problems such as hardware failures or other abnormal behaviours. Additionally, it can be any information such as weather.

3.5. *SSA Evaluation*

SA measurement is one way to evaluate SA, in which over 30 different methods are available in the current literature and can be grouped into six categories: freeze probe, real-time probe, observer rating, self-rating (SART), process indices, and performance measurement (Salmon et al. 2009). The combination of observer rating and performance measurement becomes a preferred option to evaluate SSA. By using the performance measurement, the SPR and SBA can be measured from their recognition of factors such as hazardous situations and collision risks (Gugerty 2011). Particularly, as the SBA has the subset environment of the SPR, ideally, both agents should demonstrate equal SA on this environment. For this purpose, the observers can provide useful help to measure the SA performance of both SPR and SBA.

4. Application of the SSA Model

This section presents a real-world application of the SSA model to highlight the implementation of the SA failure handlers and the transparency of the SBA's situational awareness development. For this case study, goals associated with detecting objects in a vehicle path (see Table 2) are used and, then, the related situations are illustrated in Figure 11. In such situations, the necessary information for SA level 1 development includes the relative speed, position, and acceleration of both the host and lead vehicles, which can then be used to acquire the relative distance between the host and lead vehicles. After a relative distance value is generated, it is combined with the host vehicle speed to calculate the time to collision (TTC) value and classify the frontal collision risk (FCR). A membership function presented in Figure 12 serves as an example of how to determine the level of the FCR using the TTC value and an $\propto$-*cut*. Every TTC value that has the degree of membership above the $\propto$-*cut* will be considered as a safe margin. Otherwise, it will be an unsafe margin and, therefore, the FCR is high. In recent autopilot features, drivers can also set up the desired TTC value (in x seconds), in which the driver considers this value as a convenient time to return back to the control loop safely.

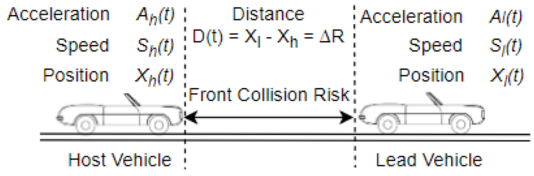


Figure 11. Front situations within the vehicle path

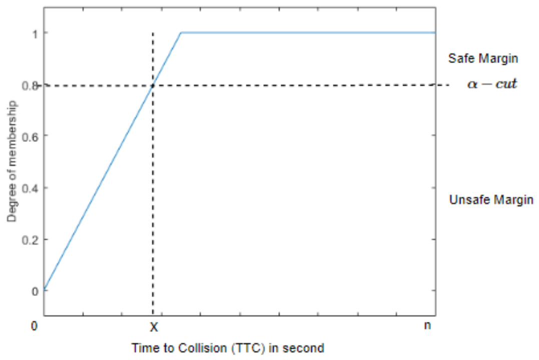
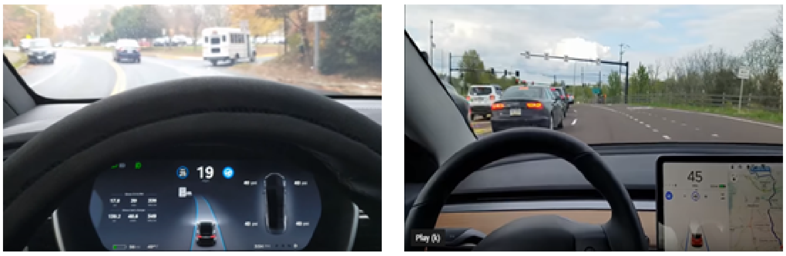


Figure 12. Membership function of the FCR using the TTC Value

Due to its limitation, the ADAS might have problems recognizing TTC risk with lead vehicles when dealing with environmental constraints, such as the curvature roads. For example, in Figure 13a, a leading vehicle in such a road is recognized as a vehicle in different lanes. Additionally, Figure 13b shows how a curved road causes limited sight, which makes the autopilot features fail to recognize a lead vehicle that stops because of a red traffic light. As a consequence, the autopilot does not reduce the host vehicle's speed. Furthermore, another environmental constraint associated with the frontal situation is traffic light state recognition, which may be due to traffic light types and placement. In this regard, unsupported road infrastructure, which was designed before autopilot technology existed, creates an unfriendly environment for such technologies.

Based on the frontal situation problems above, two examples of cognitive support and SA failure handlers in manual mode are sending an alert to the driver and executing emergency brake procedures. Commonly, the emergency brake procedures will be executed when hazardous TTC values, inattentive driving, and no braking input from



(a) Lead vehicle recognition problems. (b) Traffic light recognition issues.

Figure 13. Curvature problems.

drivers are recognized. Such mechanisms, therefore, become a way for the ADAS to observe the driver's behaviour and measure its outcome. In the meantime, the SA development of the ADAS can be observed by providing a transparency mechanism. To illustrate the implementation of transparency in the ADAS, this study models the SA development by providing several decision points associated with selected situations, which can be seen in Table 3.

Table 3. Decision Points due to the Behaviour of the ADAS in Responding to the Frontal Situation

Decision Points	Description
D1	Is there a vehicle in front?
D2	Is the current road trajectory curved?
D3	Is the vehicle ahead within the host vehicle's path?
D4	Is the TTC value of the lead vehicle above that of the user's setting?
D5	Are there any traffic lights ahead?
D6	Can the traffic light state be recognized?

Using a set of sensors and learning machines, the answer for each decision point can be inferred. For example, the data captured by an in-vehicle camera can be learned to support D1, D3, D5, and D6. Additionally, GPS and distance sensors can supply the inputs for D2 and D4, respectively. With these decision points, a decision tree can be modelled producing 21 different states and 40 path pieces (see Figure 14). As the decision points reflect the states of SA development of the ADAS, transparency information can be generated at the critical path connecting these decision points.

For example, the P14 scenario in the decision tree represents a state in which a vehicle is detected ahead (path 1) while the road trajectory is straight (path 4). As a consequence, the ADAS has a very high confidence level on its recognition result to answer the decision point D3 (path 10). Then, the P14 scenario also illustrates that the host vehicle is also under traffic light situations (path 37), and it is assumed that traffic light states can be recognized (path 39). Based on this scenario, several situational transparency information items can be generated. Starting at path 1, the vehicle's information panel can display frontal situations presenting the host vehicle position between two road lines and the position of the detected vehicle ahead, either within or outside of the host vehicle's path. Furthermore, given the traffic light detected at path 37, an icon representing a traffic light pops up (i.e., in grey only). Once the traffic light state is detected, a different colour for the traffic light icon appears to tell the driver that such a situation was well recognized. When the ADAS fails to capture the traffic light state within a particular distance, as in the P15 scenario, an alert can be sent to the driver to go back to the loop control. In this regard, such

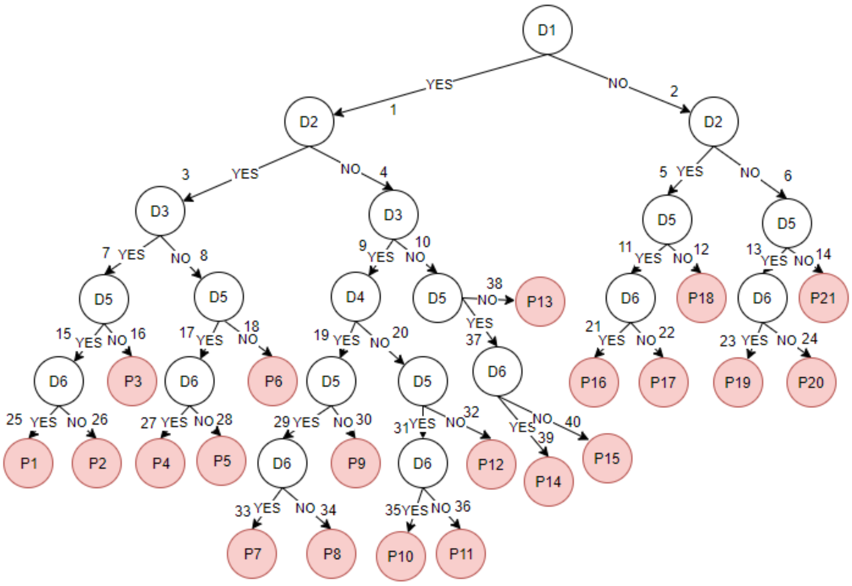


Figure 14. Decision tree regarding frontal situations

colour changes on the light icon represent the transparency of the traffic light state recognition performance which helps comprehend overall situations.

As previously mentioned, a curved road may lead to the poor recognition of lead vehicles. For such a situation (i.e., path 7 of the P3 scenario), transparency information about the lead vehicle recognition performance given the road trajectory type can be generated either in a textual or visual mode. This information can help the drivers maintain their situational awareness and set a proper trust on the ADAS behaviour and performance.

5. Conclusion

Collaborative driving can be considered as a form of HAT since autonomy has been increasingly added to the ADAS. Rather than merely a type of automation, the ADAS can also be viewed as a teammate providing, i.e., the backup when the driver experiences SA development failures and an assistance to reduce the driving workload with its collision avoidance and autopilot features respectively. However, in such collaborative driving, another SA problem arises due to over-reliance mental models. This study investigates the underlying SA concept of HAT in collaborative driving to overcome such a mental model issue. As a result, several key contributions have been made that consist of the formalization of existing multiple-agent SA models, a new SSA model to cater to collaborative driving, and a methodology to enhance the SSA-based HAT performance that considers mechanisms such as SA failure handlers and the transparency to be applied during AA's SA development. This paper also introduced TCDT framework for transparency requirements. The new SSA model is significant for enriching the existing underlying SA concept. Additionally, applying a transparent system can be a future direction of ADAS development.

The new SSA model can exist in other domains, such as military and robotics. Therefore, future research related to the SSA model can focus on enhancing team performance with multiple autonomy agents as subordinates, in which they have various SA model relationships, such as team SA, mutual SA, and shared SA. Additionally, future research can also be directed to develop an approach for transparency mechanisms to extract information regarding AA states and behaviours.

Acknowledgement(s)

This work was supported by Indonesia Endowment Fund for Education (LPDP) [Grant No. PRJ-4648/LPDP.3/2016]. The author would like to thank @nickpuschak and @aDigitalNomad.net who provided a comprehensive review on autopilot technologies; two figures (Figures 13 a, b) in this manuscript were captured from their videos.

References

Banks VA, Eriksson A, O'Donoghue J and Stanton NA (2018) Is partially automated driving a bad idea? Observations from an on-road study. *Applied ergonomics* 68: 138–145.

Bellet T, Bailly-Asuni B, Mayenobe P and Banet A (2009) A theoretical and methodological framework for studying and modelling drivers mental representations. *Safety Science* 47(9): 1205–1221.

Casner SM, Hutchins EL and Norman D (2016) The challenges of partially automated driving. *Communications of the ACM* 59(5): 70–77.

Chanda J, Kanjilal A, Sengupta S and Bhattacharya S (2009) Traceability of requirements and consistency verification of uml use case, activity and class diagram: A formal approach. In: *2009 Proceeding of International Conference on Methods and Models in Computer Science (ICM2CS)*. pp. 1–4. .

Chen JY, Lakhmani SG, Stowers K, Selkowitz AR, Wright JL and Barnes M (2018) Situation awareness-based agent transparency and human-autonomy teaming effectiveness. *Theoretical issues in ergonomics science* 19(3): 259–282.

Demir M, McNeese NJ and Cooke NJ (2017) Team situation awareness within the context of human-autonomy teaming. *Cognitive Systems Research* 46: 3–12.

Endsley MR (1995) Toward a theory of situation awareness in dynamic systems. *Human Factors* 37(1): 32–64. .

Endsley MR et al. (1997) The role of situation awareness in naturalistic decision making. *Naturalistic Decision Making* 269: 284.

Endsley MR (1999) Level of automation effects on performance, situation awareness and workload in a dynamic control task. *Ergonomics* 42(3): 462–492.

Endsley MR (2000) Situation models: An avenue to the modeling of mental models. In: *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, volume 44. SAGE Publications Sage CA: Los Angeles, CA, pp. 61–64.

Endsley MR and Jones DG (2011) Automation and situation awareness. In: *Designing for Situation Awareness*. CRC Press. ISBN 978-1-4200-6355-4, pp. 169–192. .

Endsley MR (2001) Designing for situation awareness in complex systems. In: *Proceedings of the Second International Workshop on symbiosis of humans, artifacts and environment*. pp. 1–14.

Endsley MR (2017) From here to autonomy: lessons learned from human–automation research. *Human factors* 59(1): 5–27.

Fastenmeier W and Gstalter H (2007) Driving task analysis as a tool in traffic safety research and practice. *Safety Science* 45(9): 952–979. .

Freiman M, Caisse M, Ball J, Halverson T and Myers C (2018) Empirically identified gaps in a situation awareness model for human-machine coordination. In: *2018 IEEE Conference* *on Cognitive and Computational Aspects of Situation Management (CogSIMA)*. IEEE, pp. 110–116.

Gugerty L (2011) Situation awareness in driving. *Handbook for driving simulation in engineering, medicine and psychology* 1.

Jacobson I (1993) *Object-oriented software engineering: A use case driven approach*. Pearson Education India.

Johannsdottir KR and Herdman CM (2010) The role of working memory in supporting drivers' situation awareness for surrounding traffic. *Human Factors* 52(6): 663–673.

Jones DG (2014) A practical perspective on the utility of situation awareness. *Journal of Cognitive Engineering and Decision Making* 9(1): 98–100. .

Körber M, Baseler E and Bengler K (2018) Introduction matters: Manipulating trust in automation and reliance in automated driving. *Applied ergonomics* 66: 18–31.

Ma J, Lu J and Zhang G (2010) Team situation awareness measure using semantic utility functions for supporting dynamic decision-making. *Soft Computing* 14(12): 1305–1316. .

Ma R and Kaber DB (2005) Situation awareness and workload in driving while using adaptive cruise control and a cell phone. *International Journal of Industrial Ergonomics* 35(10): 939–953. .

Matthews M, Bryant D, Webb R and Harbluk J (2001) Model for situation awareness and driving: Application to analysis and research for intelligent transportation systems. *Transportation Research Record: Journal of the Transportation Research Board* (1779): 26–32.

McAree O and Chen WH (2013) Artificial situation awareness for increased autonomy of unmanned aerial systems in the terminal area. *Journal of Intelligent & Robotic Systems* 70(1-4): 545–555.

McNeese NJ, Demir M, Cooke NJ and Myers C (2018) Teaming with a synthetic teammate: Insights into human-autonomy teaming. *Human factors* 60(2): 262–273.

Pew RD (1994) An introduction to the concept of situation awareness. In: Gilson RD, Garland DJ and Koonce JM (eds.) *Situation awareness in complex system*. Daytona Beach, Florida: Embry-Riddle Auronautical Press., pp. 17–26.

Salas E, Prince C, Baker DP and Shrestha L (1995) Situation awareness in team performance: Implications for measurement and training. *Human Factors* 37(1): 123–136. .

Salmon PM, Stanton NA, Walker GH, Jenkins D, Ladva D, Rafferty L and Young M (2009) Measuring situation awareness in complex systems: Comparison of measures study. *International Journal of Industrial Ergonomics* 39(3): 490–500.

Saner LD, Bolstad CA, Gonzalez C and Cuevas HM (2009) Measuring and predicting shared situation awareness in teams. *Journal of Cognitive Engineering and Decision Making* 3(3): 280–308.

Shen W and Liu S (2003) Formalization, testing and execution of a use case diagram. In: *International Conference on Formal Engineering Methods*. Springer, pp. 68–85.

Shu Y and Furuta K (2005) An inference method of team situation awareness based on mutual awareness. *Cognition, Technology & Work* 7(4): 272–287.

Soliman AM and Mathna EK (2009) Metacognitive strategy training improves driving situation awareness. *Social Behavior & Personality: An international journal* 37(9): 1161–1170.

Stanton NA, Salmon PM, Walker GH, Salas E and Hancock PA (2017) State-of-science: Situation awareness in individuals, teams and systems. *Ergonomics* 60(4): 449–466. .

Stanton NA, Stewart R, Harris D, Houghton RJ, Baber C, McMaster R, Salmon P, Hoyle G, Walker G, Young MS, Linsell M, Dymott R and Green D (2006) Distributed situation awareness in dynamic systems: Theoretical development and application of an ergonomics methodology. *Ergonomics* 49(12-13): 1288–1311. .

Stanton NA (2006) Hierarchical task analysis: Developments, applications, and extensions. *Applied Ergonomics* 37(1): 55–79.