ISSN 2754-1118
Vol 3, Issue 2

The Safety-Critical Systems Club

SAFETY-CRITICAL SYSTEMS eJOURNAL

SCSC
FOR EVERYONE WORKING IN SYSTEM SAFETY

Editorial to the 2024 Summer Issue

Welcome to the second issue of the third volume of the Safety-Critical Systems eJournal, published by the Safety-Critical Systems Club (SCSC) with Publication Number SCSC-196.

Last year, a correspondent wrote saying of this journal, "*If you can get through two years, you're doing OK. Three years and you're 'established'*". This is the final issue of that third year, so I hope that we are now indeed established. I would like to get to the point where there is always a (small) number of papers in the pipeline, so that the finished ones can get published in the next issue, and the others can be reworked with less time pressure than at present. I will be posting a call for new papers on LinkedIn, as well as writing directly to some potential authors.

We also need additional reviewers to expand our pool of talent, not only subject matter experts, but those with broad experience across many aspects of safety engineering and assurance. If you wish to take part, please register your interest on the web-site at: https://scsc.uk/journal/index.php/scsj/user/register. I intend to contact the current set of reviewers to ensure both that their registered e-mail address and list of specialist topics are still valid.

This issue contains three papers (one of which was postponed from the previous issue):

- Peter Ladkin (Germany) examines a guidance document prepared by ISO and IEC working groups on functional safety in the presence of "Artificial Intelligence" sub-systems in his paper "*Functional Safety and Oracular Subsystems: An Observation on ISO/IEC TR 5469: Artificial Intelligence — Functional safety and AI*". He considers critically the way the guidance expects AI subsystems to be interpreted architecturally and behaviourally, through considering the example of adaptive control. He also proposes some concepts which may be useful in characterising such subsystems.
- Niki Mok (UK) addresses alertness and attention of operators in safety-critical systems, in particular train drivers. In "*System Analysis on Driver Monitoring System for Mainline Railway*", she suggests new capabilities to augment the existing vigilance system for UK mainline passenger trains This paper was the basis of her presentation on day 2 of the last Safety Critical Systems Symposium in Bristol, SSS'24; see https://scsc.uk/re1007.76:1.
- Amit Sahu and Carmen Carlan (Germany) present "*Towards Defect-based Testing for Safety-critical ML Components", i*n which they propose a process for collecting adequate test data for Machine Learning components used in safety-critical applications. Two case studies are used to illustrate the method: stop sign recognition and railway track segmentation — both are implemented using ML components.

As ever, my thanks go to the authors for contributing their papers, and also to the anonymous peer-reviewers (at least three per paper) for suggesting improvements. Apologies also to those reviewers who made some recommendations that were not taken up.

Please also support the Club's Working Groups, which, *inter alia*, share industry best practice, develop guidance documents, and influence the development of standards. If you would like to find out more about these groups, please go to https://scsc.uk/g, which includes contact details.

John Spriggs, SCSC Journal Editor
July 2024

This collation page left blank intentionally.

ISSN 2754-1118 (Online)
ISSN 2753-6599 (Print)

Functional Safety and Oracular Subsystems

An Observation on ISO/IEC TR 5469: Artificial Intelligence — Functional safety and AI systems

Peter Bernard Ladkin

Causalis Ingenieurgesellschaft mbH, Bielefeld, Germany

Abstract

"AI" subsystems are finding their way into safety-critical systems. Use of this term has come to mean contemporarily software systems based on machine-learning (ML) techniques. It is an open question how such subsystems may be verified and validated in safety-critical applications. A guidance document for functional safety in the presence of AI subsystems has recently been published by ISO and IEC, ISO/IEC TR 5469:2024. This paper considers critically the conception expounded in TR 5469 of how such subsystems are to be construed architecturally and behaviourally, through considering the example of adaptive control. Some concepts which may be useful in characterising AI subsystems are proposed.

1 Introduction

1.1 AI Software and Subsystems

The area of computer science known as Artificial Intelligence has been around since John McCarthy coined the term in 1955[1]. The original idea was to mimic human capabilities computationally. Much of the early success was in so-called symbolic AI, with attempts to emulate logical reasoning by means of automated logic engines, such as the 1959 General Problem Solver of Newell, Shaw, and Simon (Newell et al 1959). In the 1960's, there were attempts to mimic the kinds of processing supposedly going on in human brains, with the 1969 book *Perceptrons* by Marvin Minsky and Seymour Papert an early example (Minsky and Papert 1969). Symbolic AI made considerable progress through the 1980's in areas such as automated reasoning, task planning, common-sense physics, symbolic machine learning, and expert systems. However, interest was increasing in statistical methods such as Dempster-Shafer Theory (Dempster–Shafer theory n.d.) and Judea Pearl's Bayesian Networks (Pearl 1988), as well as the successor to perceptrons, called artificial neural networks (ANN), with the general approach becoming known as connectionism (Rumelhart and McClelland 1986).

[1] The term became more generally known through the 1956 *Dartmouth Summer Research Project on Artificial Intelligence*, organised by McCarthy, Marvin Minsky, Nathaniel Rochester, and Claude Shannon.

Techniques of machine-learning, still addressed through the 1980's by symbolic methods, have made significant progress in the last quarter century through using the paradigm of ANN (I shall subsequently drop the "A") applied to increasingly large training data sets. Significant advances have been made in computer vision, pattern recognition, image analysis, and computational linguistics, as well as other fields, using the NN paradigm and large data sets. Nowadays this paradigm has more or less taken over the domain of machine learning (ML). In the *Volksmund*, "AI" has come nowadays to mean ML/NN techniques [2].

Software in critical systems based on symbolic AI techniques is mostly amenable to standard software verification and validation (V&V) techniques, whereas software based on ML is not. A trained piece of ML software is more often considered to be a "black box", and techniques such as proposed in the functional safety standard IEC 61508 (IEC 61508:2010) do not handle "black box" software whose internal structure and operation is not amenable to analysis. Statistical techniques could theoretically be used, but for the evaluation of safety-related software such techniques are controversial at time of writing. Nevertheless, the capabilities of, for example, automated vision systems entail that they are used in applications such as autonomous road vehicles, which by their nature are safety-critical. The challenge is how to encapsulate "black box" NN software in a safety-critical system, as well as how to increase our ability to derive meaningful V&V results from the internal structure of such software and its training history.

1.2 Early Success with DLNNs in Control and Vision Systems

For a number of years, systems claiming to be "artificially intelligent" (AI systems for short) have been used in safety-related applications. For example, such systems have been used in adaptive control, in which traditionally-designed control systems are enhanced by feedback from deep-learning neural networks (DLNN), which received the same sensor inputs as the traditional control algorithm and whose output is used to modify the control commands to the actuators.

Some of the earliest and most successful experiments with adaptive control were conducted in the US by NASA from the 1990's on. In the wake of the accident to United Airlines Flight 232 at Sioux City on July 19, 1989, in which the DC-10 aircraft lost the ability to actuate any of its aerodynamic control, and reverted to control purely through differential thrust from the engines, there followed a successful NASA project to control the flight of a large transport aircraft, an MD-11 (the successor aircraft to the DC-10), using differential engine thrust only (NASA 2002). The idea here was that, instead of the pilots learning how directly to use engine thrust commands to control flight, as the Sioux City accident crew did, the crew of the adaptive-control MD-11 gave the usual aerodynamic-control inputs for pitch, bank and yaw, but these were translated into differential-thrust commands with the "same" effect by an interposed DLNN. The DLNN was trained statically, but also engaged in dynamic "learning" (further adaptation) during flight, hence the term "adaptive control".

Figure 1, overleaf, shows the Propulsion Controlled Aircraft (PCA) MD-11 during trials.

[2] More recently, since the end of 2022 with the release of ChatGPT, it seems to have even more narrowly come to mean Large Language Models (LLMs). The AI subsystems considered in this paper are not LLMs.

Figure 1 ~ PCA MD-11 landing at Dryden under Propulsion Control, 1995-08-29

There followed a series of flight experiments in the early 2000's, with a modified military F-15 interceptor aircraft (known as a NASA NF-15), in which the adaptive control had two goals. First, that "normal" control input could be exercised by the pilot even when some aerodynamic control surfaces were no longer available (for example, due to battle damage). Second, to enhance certain manoeuvres beyond that which the pilot could achieve using standard control. The "adaptive" system interpreted pilot's intent and using a different combination of control surfaces and propulsion (NASA 2014, paragraph *Intelligent Flight Control System*).

Figure 2 shows the NF-15B in flight during trials of the Intelligent Flight Control System.

Figure 2 ~ The NASA NF-15B on an Intelligent Flight Control System Mission

Flight is in some sense an easier control environment than the ground, because there is air everywhere you look (and fly). Whereas using wheels on a surface there are routinely obstacles; not only hard objects barring the way, but different surfaces affecting traction and steering, not to speak of other mobile objects.

These adaptively-controlled aircraft were not "drones". They had human pilots and, although the flights took place over a sparsely-populated (or non-populated) area of the Californian Mojave desert, the range of possible grounding locations in the event of an in-flight upset is of the order of tens of kilometres from the (on-ground) location of the point of upset. The V&V is non-trivial: Nguyen and Jacklin (2010, Section 4.2) state: *"[t]he current approach is to verify a neural net adaptive flight control over an exhaustive state space using the Monte Carlo simulation method*".

In contrast, adaptive control in road vehicles can be pursued in environments from which humans are excluded, and therefore, with appropriate boundary controls, upsets can be contained without engendering harm. The first automated-road-vehicle-in-actual-environment experiments commenced with the DARPA Grand Challenge off-road competition in 2004. No vehicle finished. In 2005 (Figure 3), five vehicles finished the 212 km off-road course (DARPA Grand Challenge (2005) n.d.).

Figure 3 ~ On the Off-road — the DARPA Grand Challenge 2005

Figure 4 shows the award ceremony for the 2005 winner.

Figure 4 ~ The 2005 Winner — Stanley, from the Stanford AI Laboratory

In 2007, the DARPA Urban Challenge put competitors into a simulated urban traffic environment in which the moving obstacles were other competitors' vehicles, see Figure 5 overleaf. The "course" was 96km long and had to be completed in 6 hours. 6 teams finished.

Figure 5 ~ DARPA Urban Challenge 2007

"Assistance systems" built on such experience have now been incorporated into road vehicles which operate on public roads. The Tesla Autopilot has been involved in a number of road accidents, some of them fatal. Autopilot operation is intended to be actively monitored by the human driver, but this has not always happened. (The weaknesses of supervisory control, as this is called, have been well-studied for decades.) The US NHTSA[3] has investigated, and is investigating, a number of Tesla accidents. At least two fatal Tesla accidents happened when the car was under Autopilot control, failed to recognise an obstacle (a truck crossing the road at an intersection; respectively a metal guardrail) and the driver did not react. News reports said in March 2021 that the agency was currently reviewing 23 Tesla crashes (Shepardson 2021). We understand that the majority of these investigations were requested by the automaker Tesla itself.

On 2018-03-18, a woman walking her bicycle across a road in Tempe, Arizona, was hit and killed by an experimental automated road vehicle in the fleet of Uber. An evaluation of system behaviour up to and including the collision showed that she and her bicycle, while sensed, were interpreted as an obstacle by the system, but the classification was variable, rapidly changing and resetting in the five seconds before collision (Harris 2019). There was considerable technical discussion at the time of the design of the sensing system. Further, an autonomous braking system on the vehicle installed by the manufacturer, Volvo, had been disabled in favour of Uber's own system, which did not brake until a fraction of a second before collision. The US NTSB found that Uber's 40 experimental vehicles had been involved at this point in 37 "incidents", many of which were collisions, in the year and a half between September 2016 and March 2018.

In summary, AI subsystems are already in use in safety-related systems, in controlled, "test" circumstances, even when performing on public roads, and have not always performed perfectly. Accordingly, standardisation bodies are attempting to formulate general principles for the incorporation of AI subsystems into safety-related systems. ISO/IEC TR 5469 is one such document (ISO/IEC TR 5469:2024).

[3] NHTSA, The National Highway Traffic Safety Administration is an agency of the U.S. federal government, part of the Department of Transportation

2 General Observations on Standards, and What ISO/IEC TR 5469 Tries to Do

Standards documents are hierarchically ordered. They contain clauses, which are like individual sections treating one major subtopic, which themselves consist of hierarchically-numbered subclauses. They also contain a Bibliography, as well as Annexes, which amplify on matters contained in the main text. Clauses, and subclauses, may be normative or informative. A normative subclause contains a requirement which must be fulfilled by anyone claiming to follow the standard; they are nominally mandatory. An informative subclause contains guidance, but nothing nominally mandatory. The TR 5469 document (as I shall call it for short) is a *Technical Report*, that is, it is purely informative and there is nothing normative about it. However, courts and other organisations do take IEC standards documents, whether informative or normative, to provide guidance as to how the electrotechnical profession views the state of its practice, as to how "things should be". For outside organisations, then, the intra-standards distinction between informative and normative is not as significant. However, the writers of standards documents do typically distinguish between how (they think) things *must* be, and how they *could* be conformant with the state of the *praxis*.

The authors of TR 5469 appear to be conceptualising how an AI subsystem can be embedded in a safety-related system, and what assessments are required in order to determine if it is to perform safely. The question arises whether this conception applies generally.

3 Architectures for Feedback Control

3.1 Control Systems and Systems with AI Subsystems

Franklin et al (1994) state that:

> *Control is the process of causing a system variable to conform to some desired value, called a reference value.*

A control-block architecture for a traditional automobile cruise control is shown in Figure 6, which is derived from Franklin et al. (1994, Figure 1.3).

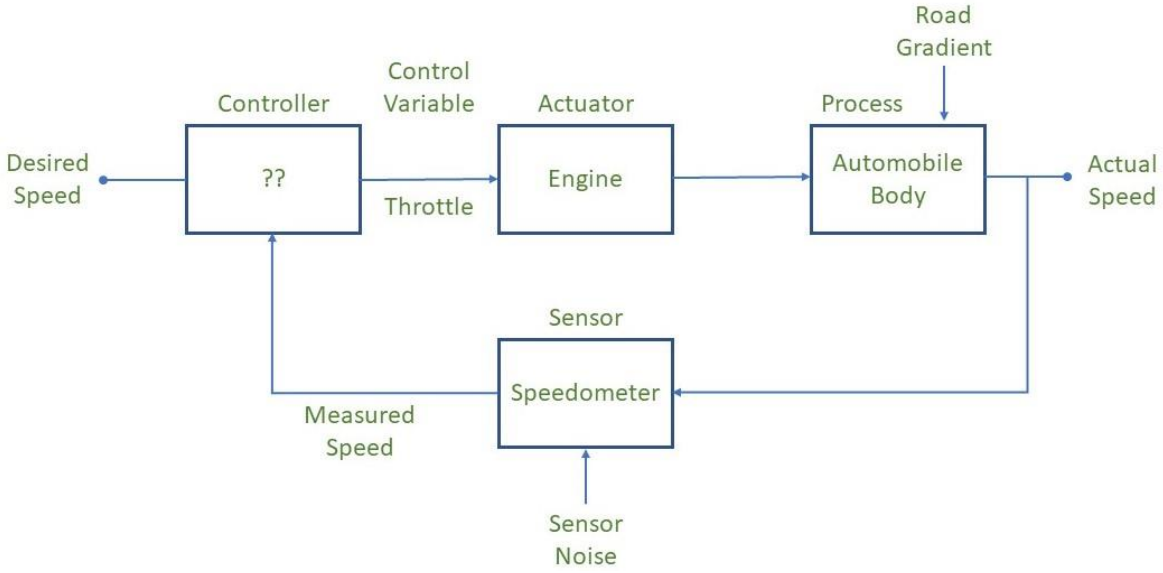

Figure 6 ~ Control Block Diagram for a Cruise Control System

This shows feedback control with an active component, a controller, which is further specified in a Function Block Diagram (FBD) in Figure 7 (derived from Franklin et al. (1994, Figure 1.4)), which specifies the mathematical relationships used by the controller to produce its output.

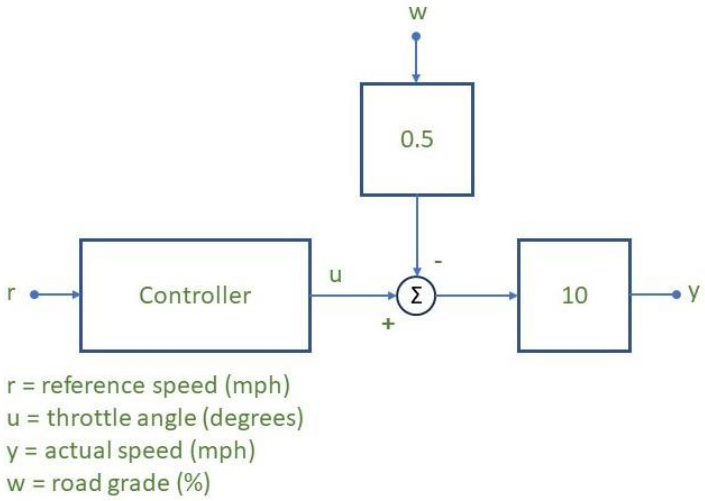

Figure 7 ~ A Function Block Diagram of a Cruise Controller

FBDs for specific examples of open-loop control and closed-loop control are shown in Figures 8 & 9, respectively (derived from Franklin et al. (1994, Figures 1.5 & 1.6)). The number *1/10* in the *Controlle*r block of Figure 8, for example, means that the input value designated by *r* is multiplied by *1/10* to become the value designated by *u*, i.e. $u = r/10$. On the other input branch, the input value designated by *w* has been multiplied by *0.5* and then turned negative: that is, $-0.5w$. These two quantities are then summed (the Σ) and the result is then multiplied by *10* to become *y*. This means in this simple case $y = 10(r/10 - 0.5w) = r - 5w$ (Franklin et al 1994, Chapter 1).

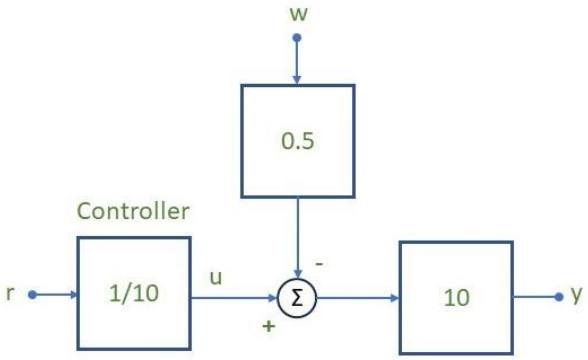

Figure 8 ~ Open-Loop Cruise Control FBD

The value of y in the closed-loop control example in Figure 9 is a little trickier. Input r is multiplied by 100 to form u*: $u = 100.r$*. This is then summed with $-0.5w$ as before, and the result multiplied by 10 as before. However, the result $10(100.r - 0.5w)$ is then "fed back" to be summed with r at the leftmost Σ, and the result $r - 10(100.r - 0.5w))$ then goes through the diagram again. This can be put into simultaneous equations and solved analytically (Franklin et al 1994 Chapter 1), but a good way to think of it is that there is an equilibrium value for fixed r and w, which is "distorted" when one of them changes slightly, and adjustment leads to a new value of y.

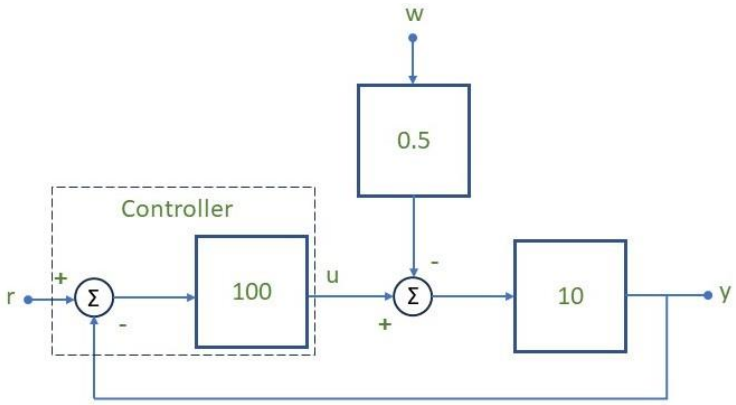

Figure 9 ~ Closed-Loop Cruise Control FBD

A modified Control Block Diagram, incorporating elements of an FBD, for the NASA NF-15B control system architecture is given in Figure 10, after Smith et al. (2010, Figure 2). A more detailed FBD for the combination of adaptive neural network and inversion controller is given in Figure 11, after Nguyen and Jacklin (2010, Figure 1). (These are included here for expository completeness; the rest of the text does not use any specific mathematically-annotated FBDs.).

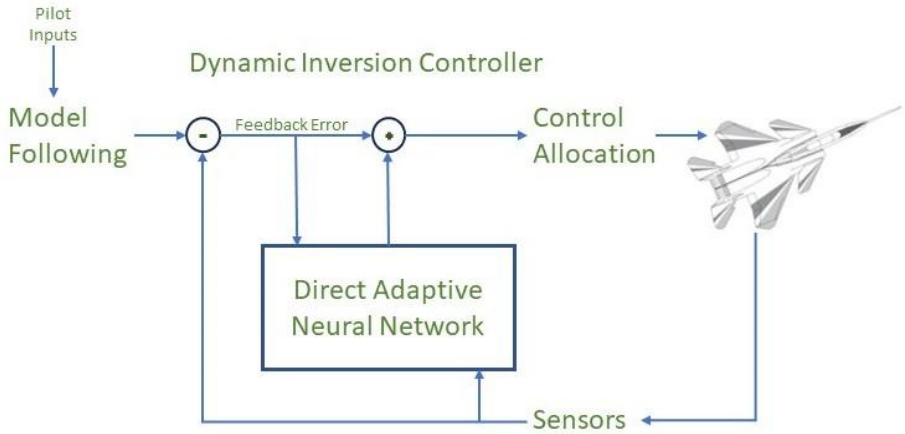

Figure 10 ~ IFCS System CBD+Abstract FBD

I shall show that these CBD-FBD architectures are not accommodated in the current architectural conception of TR 5469. It follows that the considerations of TR 5469 do not apply *per se* to these adaptive-control architectures.

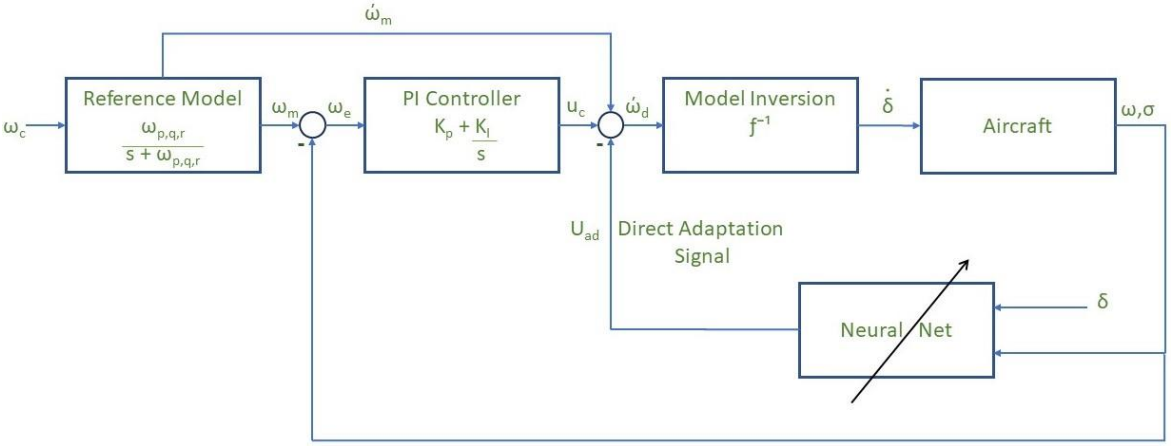

Figure 11 ~ The FBD for the NF-15B Adaptive Control

3.2 A Control Block Diagram for Adaptive Feedback Control

A controller has sensors which determine the state and behaviour of the controlled object (here, I shall call it a vehicle) and the state and behaviour of the environment. The behaviour (that is, change of state) of both entities can be captured in *history variables*, so, for example, from elementary differential calculus we know that rates may be captured by first derivatives, accelerations by second derivatives. The first and second derivatives can be included in the state as history variables, and usually are. They are derived by calculation from discrete location data over the previous short time period. The specification of the controller determines what quantities are required for control, so the history variables which are needed are determined by the specification of the controller input. Thus can "state and behaviour" be replaced by "state" alone. However, I mention both here.

Figure 12 shows the inputs to the sensors from the state+behaviour of the controlled object, the EUC[4] in the terminology of IEC 61508, here denoted as the vehicle, and from the state+behaviour of the environment.

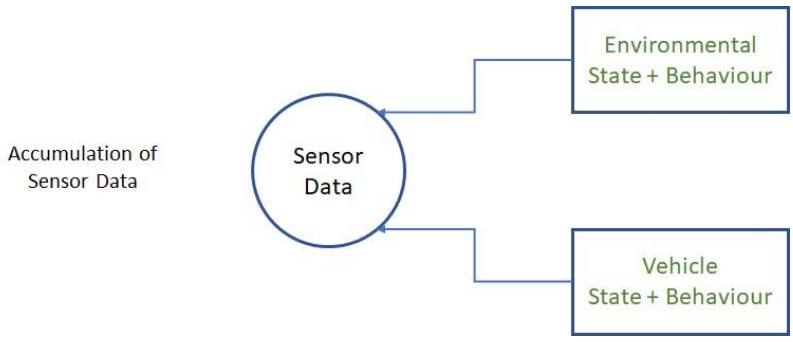

Figure 12 ~ Sensors Receive Input from Vehicle + Environment

The sensor data then needs to be incorporated into a model of the physical environment + vehicle, suitable for making control decisions, as in Figure 13.

[4] Equipment Under Control

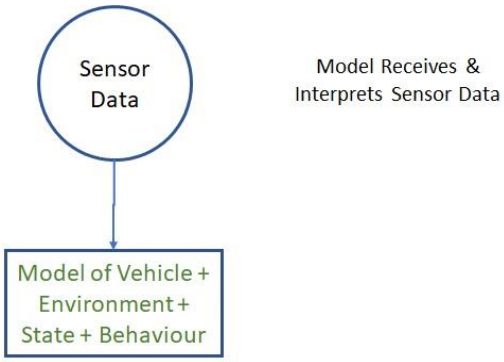

Figure 13 ~Sensor Data is Passed to a Model Component

The adaptive controller receives information from the model on the state+behaviour of the environment and the vehicle, and determines what actuator commands to output to the actuation subsystems, as well as possibly revising its own configuration, if it is a dynamically-trained NN (or not, if it is statically trained). This is shown in Figure 14.

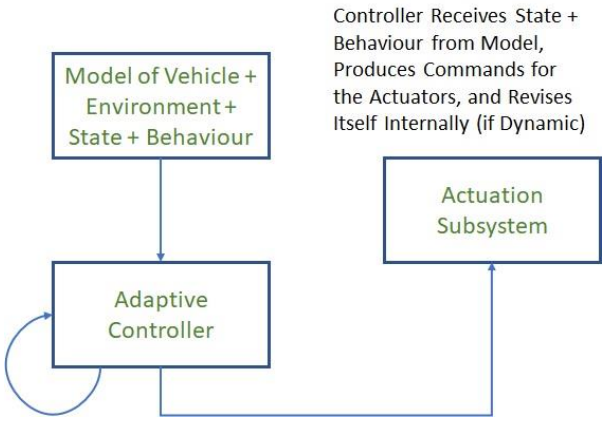

Figure 14 ~ Controller Receives Input from Model, Determines Output for Actuation

In Figure 15, the actuation operation is shown.

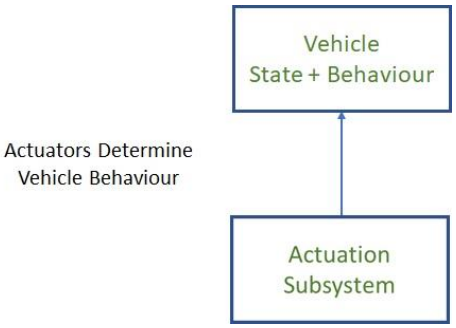

Figure 15 ~ Actuation

Finally, the entire feedback control loop looks as follows in Figure 16.

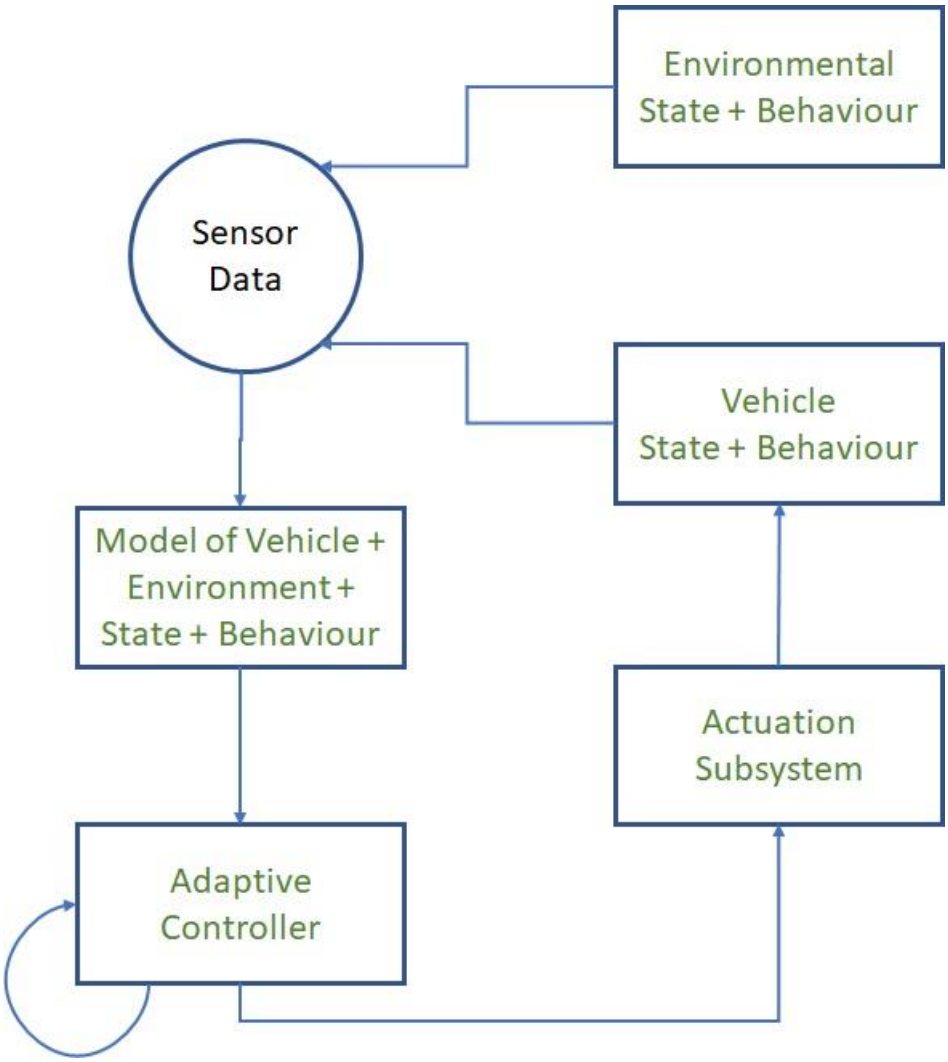

Figure 16 ~ The Feedback Control Block Diagram

4 System and Software Architecture in Standards Documents: A Comparison

4.1 System Architecture According to TR 5469

TR 5469 explains its system architectural concept in words (ISO/IEC TR 5469:2024): *" ……. the capability of AI is often achieved by the combination of an algorithm and a model. The model typically represents information that achieves the application's purpose, (e.g. knowledge about how various inputs are to be distinguished and recognized), while algorithms infer information from a model and inputs, (e.g. to make a prediction).*"

The model is a kind of "knowledge engine", in other words, and the algorithm a query facility for the knowledge engine. Another way of expressing this is to say that the model is an oracle for the domain about which it represents knowledge — an oracular subsystem. Figure 17 gives a control block diagram for this "*achievement*" of the "*capability of [AI]*".

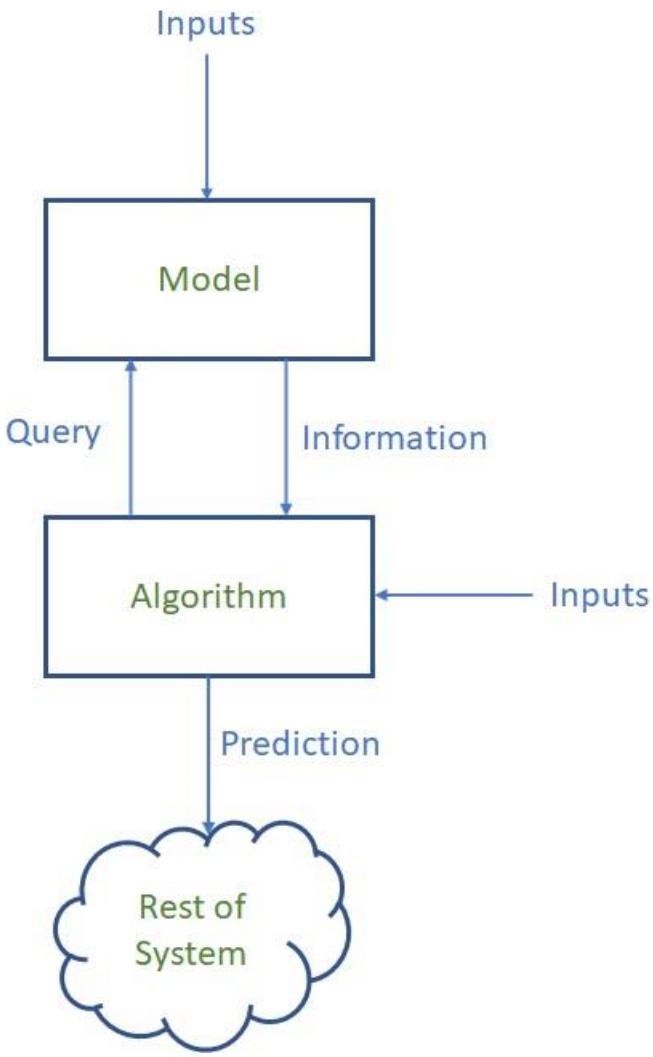

Figure 17 ~ Control Block Diagram of the "Capability of AI" as Model+Algorithm

The model receives inputs from somewhere, and uses these inputs to construct its representation of its knowledge. The algorithm uses knowledge from the model (the transaction is represented by a query from the algorithm and information passed from model to algorithm). It also may have (according to the text above) further "*inputs*". Its output, according to the text, is a "*prediction*". This is precisely what is represented in the control block diagram in Figure 17. I shall continue to use the example of a *prediction* output in what follows.

Let us compare the CBD in Figure 17 with that in Figure 16. I write the item labels in italics. There is a *Model* in both diagrams. In the feedback-control CBD in Figure 16, the *Model* represents knowledge about the actual state (+behaviour) of vehicle + environment. This is what would be in the *Model* also in Figure 17. The *Inputs* in Figure 17 would be the *Environmental State+Behaviour* and the *Vehicle State+Behaviour* from Figure 16. These parts of the CBDs cohere with each other. There is a question with feedback control as to what in Figure 16 might play the role of the block *Algorithm in* Figure 17. There are three main differences between the CDBs, as follows:

- In Figure 16, there is no *Query* from the *Adaptive Controller* to the *Model*. Any communication from *Adaptive Controller* to *Model* proceeds through *Actuation*, *Vehicle State+Behaviour*, and *Sensor Data*; in other words through the feedback loop. The

information that the *Adaptive Controller* gets from the *Model* in feedback control is determined in advance.
- The output from an *Adaptive Controller* is not *Prediction*, as in Figure 17, but is a set of actuation commands to the *Actuation Subsystem*.
- The *Adaptive Controller* does not receive any *Inputs* other than those which come from the *Model*, nor do any other elements on the path to *Vehicle State+Behaviour* (namely the *Actuation Subsystem*).

In respect of the first and third points, Figure 16 with *Adaptive Controller* regarded as the *Algorithm* could be regarded as a special case of the architecture in Figure 17, namely an instance without *Query* from *Algorithm* to *Model* and without *Inputs* to *Algorithm*. However, that the output from *Algorithm* is to be a *Prediction* in Figure 17, but is *actuation commands* in Figure 16, is an irreconcilable difference. ("*Prediction*" is, according to the text from TR 5469 above, only one example of an algorithm output. I have not explored other examples, since for my purpose here it suffices to show one irreconcilable difference between the architectures.)

Furthermore, the *Model* in Figure 17 is supposed to contain the entire AI component, and the *Algorithm* is a "traditional" software object. According to TR 5469 Section 7.2:

> *"Usually, algorithms that interact with the models contain only a limited amount of knowledge or implications about the application's goals. This is quite similar to the role of foundational software libraries or programming environments (compilers, etc.) in non-AI software. That is, the algorithm itself does not play a functional safety role, but its correctness is critically important for functional safety to be achieved. In this way, the integrity of algorithms in AI technology can often be handled with existing principles of functional safety as specified in the IEC 61508 series ....., similar to that of non-AI software components."*

There are four points of difference here between the *Adaptive Controller* in Figure 16 and this description of *Algorithm*.

- Most importantly, the Adaptive Controller does not "*contain only a limited amount of knowledge or implications about the application's goals ..... quite similar to the role of foundational software libraries or programming environments*". It embodies all "*knowledge and implications about the application's goals*" in that it is the controller.
- Second, it contains AI itself, namely a DLNN exercising the adaptive control. It is not at all "*similar*" to a "*software librar[y]*" or a "*programming environment[ ]*."
- Far from "*not play[ing] a functional safety role*" the Adaptive Controller is one of the main actors in terms of the safety of the system. If it gives the wrong actuation commands, the vehicle's behaviour will change and may cause harm. Any safety functions to be introduced to avoid or mitigate the risk must be introduced between *Adaptive Controller* and *Actuation Subsystem*, and presumably receive inputs from the *Model*, maybe under *Query* requests.
- It is implausible that its integrity "*can often be handled with existing principles of functional safety as specified in the IEC 61508 series ....., similar to that of non-AI software components.*" This is exactly why TR 5469 is being written — because the safety integrity of AI-technology based components is not seen to be encompassed by IEC 61508 as it is.

4.2 Other Applications of the "Model-Algorithm" Architecture

We have just seen that the architecture of Figure 17 is inappropriate for describing adaptive control using DLNNs. However, it is appropriate for describing various other "knowledge" architectures. Theorem provers, although used in AI, are largely associated with attempts at system and program verification, having been largely supported by the US Department of Defense. A history up to about twenty years ago may be found in (MacKenzie 2001). Decision procedures are used for various decidable mathematical theories in most theorem provers nowadays. Various mathematical theories are decidable, for example Presburger arithmetic and Skolem arithmetic, and the theories of finite fields and of real-closed fields, but some are not, for example group theory (although Abelian group theory is decidable). The question of how to incorporate decision procedures into general theorem-proving mechanisms was addressed in two definitive papers by Nelson and Oppen (1979) (1980). Shostak also devised a method of incorporating mathematical domain-specific theories into a general theorem prover in (Shostak 1984). The "Shostak prover" was the inference engine in SRI International's EHDM[5] specification and verification system.

A decision procedure, or a domain-specific mathematical theory, can fulfil the role of *Model* in Figure 17, and the general inference mechanism the role of *Algorithm*. Thus the CBD in Figure 17 describes also the architecture of such cooperating theorem provers, which is nowadays a popular architecture for theorem provers.

4.3 The Definitions of "Model" and "Algorithm"

Despite their central use in the above description of an "AI architecture" in TR 5469, the terms "algorithm" and "model" are not defined in the Terms and Definitions section of the TR.

There is a definition of "model" in the ISO/IEC standard defining AI concepts and terms (IEC 22989:2022, subclause 3.1.23).

> **model**
>
> physical, mathematical or otherwise logical representation of a system, entity, phenomenon, process or data
>
> [SOURCE: ISO/IEC 22989:2022, 3.1.23]

There is no definition of "algorithm" *per se* in either the draft TR or IEC 22989; rather, there is a definition of "machine learning algorithm" (IEC 22989:2022, subclause 3.3.6).

> **machine learning algorithm**
>
> algorithm to determine *parameters* (3.3.8) of a *machine learning model* (3.3.7) from data according to given criteria
>
> EXAMPLE: Consider solving a univariate linear function $y = \theta 0 + \theta 1x$ where y is an output or result, x is an input, $\theta 0$ is an intercept (the value of y where x=0) and $\theta 1$ is a weight. In *machine learning* (3.3.5) the process of determining the intercept and weights for a linear function is known as linear regression.
>
> [SOURCE: ISO/IEC 22989:2022, 3.3.6]

[5] Enhanced Hierarchical Development Methodology

There are two definitions of "algorithm" (unqualified) in the International Electrotechnical Vocabulary (IEV n.d.).

> **algorithm**
>
> completely determined finite sequence of instructions by which the values of the output variables can be calculated from the values of the input variables
>
> Note 1 to entry: The behaviour of a system with discrete-value input and output variables (for example a switching system) may be described completely by an algorithm. For a system with continuous-value and continuous-time input and output variables the algorithm is given by or derived from the mathematical relationship between the input and output variables.
>
> Note 2 to entry: This entry was numbered 351-21-37 in IEC 60050-351:2006.
>
> [SOURCE: IEV 351-42-27][6]
>
> **algorithm**
>
> finite set of well-defined rules for the solution of a problem in a finite number of steps
>
> [SOURCE: IEV 714-21-02][7]

Besides these definitions, there is yet another definition of "algorithm" in IEC standards, which is a variant of IEV 714-21-02:

> **algorithm**
>
> finite set of well-defined rules for the solution of a problem in a finite number of operations
>
> [SOURCE: IEC 61499-1:2012 *Function blocks — Part 1: Architecture*; also IEC 61804-2:2018 *Function blocks (FB) for process control and electronic device description language (EDDL) — Part 2: Specification of FB concep*t (SC65E).

The two IEV definitions are homonyms and nominally require resolution (the IEV is supposed to be homonym-free)[8]. It is clear from these definitions that a "machine learning algorithm" is different from an "algorithm" as defined (in either version) in the IEV, or as defined in Function Block standards. To which definition TR 5469 is intended to adhere is not specified in the document.

5 Models, Precision and Vagueness

5.1 A Set of Concepts for Properties of "Models"

We have seen that TR 5469 says that a **model** "*represents information that achieves the application's purpose*". It follows that a model is a knowledge-representation artifact, an

[6] Section 351 of the IEV is Control technology.

[7] Section 721 of the IEV is Digital technology — Fundamental concepts

[8] Annex SK of ISO/IEC Directives (2021). gives Rules for terminology work. Section SK 2.3, p104, contains the requirement *The rule "one concept – one definition" shall be applied*. Terms with multiple definitions are to be distinguished by *adding a specific use* to the term (p104), denoted in angle-brackets "<...>".

oracle for a knowledge domain. The terms "*knowledge representation*" and "*knowledge representation and reasoning*" (KR&R) are common in AI. The Wikipedia page for KR&R, however, only adduces symbolic-AI approaches to KR&R. It does not include NNs or ML approaches (Knowledge representation and reasoning n.d.). This is appropriately traditional in the AI subfield.

When computing, there is typically a collection of subject-matter domains, within which certain kinds of computations are performed. I have noted that one design of theorem prover, for example the Shostak prover or the Nelson-Oppen approach, uses a general logic engine and a collection of decision procedures for formal mathematical domains (such as "Presburger arithmetic" for arithmetic). Obviously, a decision procedure is a means of "representing knowledge" about the domain over which it decides assertions; arithmetic, say, in the case of Presburger arithmetic, or Skolem arithmetic, or quantifier elimination for finite fields. Such a procedure can be regarded as an oracle for its domain.

It seems worthwhile to define "domain"[9]. A **domain** is a collection of types of mathematical, or logical objects, or such which are representations of real-world objects, with explicit properties, relations and operations, which are nominally closed with respect to those operations. Associated with the domain is a **domain language**, which has terms for the types, the properties, relations and operations and can express assertions concerning them (for example, using a many-sorted logical language; the sorts correspond to the types of object). Let me call *answering questions and deciding (the truth of) assertions* in a domain **decision making** or **making decisions** in the domain. A model is then a *computational means of decision making in the domain language*.

A domain is **precise** in so far as questions and assertions in the domain language have determinate real-world answers, respectively truth-values, and **vague** in so far as it is not precise. A model is **precise** or **vague** in so far as its domain is precise or vague. A model is **accurate** in so far as its answers to questions and the truth-values of its assertions cohere with the real-world answers. A model is t**raceable** in so far as the computations which give the answers to questions and determine the truth-values of assertions produce as output (upon request) a chain of reasoning which determines that the answers, respectively the truth-values, are correct and/or justified.

A model may be vague and nevertheless accurate, for example a DLNN decision procedure that determines whether Magnetic Resonance Tomography scans are "normal" or anomalous. There are two artifacts in play in such an example. The first translates pixels into the model domain. The domain consists of the parts of the bodily anatomy, as well as anomalous or parasitical objects and properties. The accuracy of the model likewise consists in two phenomena. First, the veridicality of translation from pixels to model domain ("image recognition"). Second, the designation of areas of concern (deviations from "normal"). Model accuracy is determined by experience, and coherence with expert judgements and outcomes.

A model may be vague and nevertheless traceable, in that, for every decision, it can output intermediate or auxiliary decisions which provide a humanly-recognisable rationale for its decision. One can imagine the model for Magnetic Resonance Tomography scans outputting on request the anatomical configurations it has identified, along with the areas of concern, and the characteristics of those areas of concern. A human doctor can assess the anatomical configuration for accuracy in a more or less precise manner. The

[9] The definitions given in this and the following paragraph of *domain*, *domain language*, *decision making*, *making decisions*, *model*, *precise* (for domain and model), *vague* (for domain and model), *accurate*, and *traceable* are mine.

identification of areas of concern is more-vague. One can imagine that the usefulness of such an identification lies in simply directing the attention of a human to those areas, rather than any more complex calculation such as automatically proposing a conclusion from within the model.

Nelson-Oppen-style decision procedures form precise models, as do Bayesian networks (and other statistical methods). Nelson-Oppen-style decision procedures are accurate. However, Bayesian networks are not necessarily accurate.

DLNNs and other ML techniques are used to build vague models. They may be accurate, or not. They may be traceable, or not.

Bayesian networks are partially traceable. The weights on edges constitute assertions as to how strongly the phenomena named in the node labels are correlated. A trace consists of the conjunction of these correlations for all adjacent node pairs.

5.2 Concerning Algorithms

A model M may be a subsystem or subcomponent of a computation-based system S. Typically, not all the representational features of M will be used in S, but only a subset. It seems to me from the text above that the authors of TR 5469 want an *algorithm* in the AI sense to be query engine for M in S. That is consistent with the current definitions of *algorithm* in the IEC glossary.

5.3 Other Desirable Properties

If you train a DLNN on data, you intuitively want the DLNN to exhibit the same decision making no matter in what order you give it the data in training. Call the DLNN **commutatively invariant** if so. Ross Anderson's group has discovered public-facing DLNNs that are not commutatively invariant (Schumailov et al. 2021). D'Amour et al. (2020) have suggested that some DLNNs are underspecified. I suspect that non-commutative-invariance is a subcategory of underspecification.

5.4 Partial Conclusion on Model Properties

It seems that precision/vagueness, accuracy and traceability capture the qualities of a model which are relevant for assessment for functional safety properties. The various properties of specific DLNNs arising out of the work of both Shumailov et al. and D'Amour et al. seem to be highly relevant to determining accuracy and traceability.

5.5 Observations on Safety

To determine safety properties of a system, it is necessary to perform system-level and subsystem-level hazard analyses. These are required as part of “Hazard and Risk Analysis” (H&RA) in subclause 7.4 of IEC 61508-1. A causal analysis (the left part of the “Bow Tie”) is required to find out how subsystems can contribute to causal factors of a hazard. If we are talking an adaptive controller, then it trivially seems as if we require it to be accurate on all inputs in its operational envelope. If we are talking a model of another sort, along with an algorithm which queries it, then the answers to queries (may) form part of a causal chain leading to a hazard. So it is necessary to know what queries are raised, and whether the model can be argued to be accurate on those queries. Given the conceptualisation in TR 5469, this seems to me to be all that is required.

Exactly how this is operationalised for various representations of risk is a further question. Since functional safety in standards is phrased in terms of risk, this is the big question. I do not address it further here.

6 Summary

Adaptive control was one of the first applications of machine learning and DLNNs in safety-critical systems. However, the general architecture of ISO/IEC TR 5469 does not fit the architecture of adaptive control systems. The functional safety of adaptive controllers has not been adequately addressed in electrotechnical standards and it follows that it will not be adequately addressed through publication of TR 5469. The notions of precision/vagueness, accuracy, and traceability apply (or not) to such oracular systems as are built using AI and it seems as if desirable properties for functional safety can be phrased in terms of them.

Correspondence Address

Corresponding e-mail address: ladkin@causalis.com.

Acknowledgments

I thank Stuart Russell for apprising me of D'Amour et al's work, and Ross Anderson for apprising me of Shumailov et al's work.

The images of Figures 1 and 2 are from the NASA Dryden Flight Research Center[10] photo collection. Figures 3, 4 and 5 are from the Defense Advanced Research Projects Agency of the US Government (DARPA), and available from Wikimedia.

References

D'Amour A., Heller K., Moldovan D., Adlam B., Alipanahi B., Beutel A., Chen C., Deaton J., Eisenstein J., Hoffman M. D., Hormozdiari F., Houlsby N., Hou S., Ferfel G., Karthikesalingam A., Lucic M., Ma Y., McLean C., Mincu D., Mitani A., Montanari A., Nado Z., Natarajan V., Nielson C., Osborne T. F., Raman R., Ramasamy K., Sayrea R., Schrouff J., Seneviratne M., Sequeira S., Suresh H., Veitch V., Vladymyrov M., Wang X., Webster K., Yadlowsky S., Yun T., Zhai X., and Sculley D. (2020). *Underspecification Presents Challenges for Credibility in Modern Machine Learning*. Preprint, 2020-11-06, available at https://arxiv.org/abs/2011.03395. Accessed 29th August 2023.

DARPA Grand Challenge (2005). (no date). In Wikipedia: https://en.wikipedia.org/wiki/DARPA_Grand_Challenge_(2005) . Accessed 29th August 2023.

[10] Now the NASA Armstrong Flight Research Center.

Dempster–Shafer theory. (no date). In Wikipedia: https://en.wikipedia.org/wiki/Dempster–Shafer_theory. Accessed 29th August 2023.

Franklin G. F., Powell D. J., and Emami-Naeini E. (1994). *Feedback Control of Dynamical Systems*, Third Edition. Addison-Wesley, Boston MA.

Harris M. (2019). *NTSB Investigation Into Deadly Uber Self-Driving Car Crash Reveals Lax Attitude Toward Safety*. IEEE Spectrum, 2019-11-07. Available at https://spectrum.ieee.org/cars-that-think/transportation/self-driving/ntsb-investigation-into-deadly-uber-selfdriving-car-crash-reveals-lax-attitude-toward-safety. Accessed 29th August 2023.

ISO/IEC Directives. (2021). *Procedures for the technical work — Procedures specific to IEC*. Directives Part 1 + IEC Supplement, Edition 17, 2021-05. International Organisation for Standardization/International Electrotechnical Commission, Geneva.

IEC 61508. (2010). *Functional safety of electrical/electronic/programmable electronic safety-related systems*. IEC 61508, in 7 parts[11], 2nd Edition, 2010. International Electrotechnical Commission, Geneva.

IEV. (no date). *International Electrotechnical Vocabulary*. International Electrotechnical Commission, IEC 60050. Available from https://www.electropedia.org. Accessed 29th July 2023.

ISO/IEC 5469. (2024). *Artificial Intelligence – Functional safety and AI systems*. ISO/IEC TR 5469:2024, 1st Edition, January 2024. International Organization for Standardization/International Electrotechnical Commission, Geneva.

ISO/IEC 22989. (2022). *Information technology — Artificial intelligence — Artificial intelligence concepts and terminology*. ISO/IEC 22989:2022, 1st Edition, July 2024. International Organization for Standardization/International Electrotechnical Commission, Geneva.

Knowledge representation and reasoning. (no date). In Wikipedia: https://en.wikipedia.org/wiki/Knowledge_representation_and_reasoning. Accessed 29th August 2023.

MacKenzie D. (2001). *Mechanizing Proof*. M.I.T. Press, Cambridge MA.

Minsky M., and Papert S. (1969). *Perceptrons*. M.I.T. Press, Cambridge MA.

NASA. (2002). *MD-11 Propulsion Controlled Aircraft (PCA)*. National Aeronautics and Space Administration (NASA). Available at https://www.nasa.gov/centers/dryden/multimedia/imagegallery/MD-11PCA/MD-11PCA_proj_desc.html. Accessed 29th August 2023.

NASA. (2014). *NASA Armstrong Fact Sheet: NF-15B Research Aircraft*. National Aeronautics and Space Administration (NASA). Available at https://www.nasa.gov/centers/armstrong/news/FactSheets/FS-048-DFRC.html. Accessed 29th August 2023.

Nelson G., and Oppen D. C. (1979). *Simplification by cooperating decision procedures*. ACM Transactions on Programming Languages and Systems, 1(2):245–257, 1979.

[11] Parts are typically denoted with hyphen and part number, e.g., IEC 61508-3:2010 is Part 3, dealing with software development requirements.

Nelson G., and Oppen D. C. (1980). *Fast decision procedures based on congruence closure*. J. Ass. Comp. Mach., 27(2):356–364, 1980.

Newell A, Shaw J. C., and Simon H. A. (1959). *Report on a general problem-solving program*. Proceedings of the International Conference on Information Processing, Paris 15–20 June 1959. Available at http://bitsavers.informatik.uni-stuttgart.de/pdf/rand/ipl/P-1584_Report_On_A_General_Problem-Solving_Program_Feb59.pdf. Accessed 29th August 2023

Nguyen N., and Jacklin S. A. (2010). *Stability, Convergence, and Verification and Validation Challenges of Neural Net Adaptive Flight Control*. In: Schumann J., and Liu Y. (editors) *Applications of Neural Networks in High Assurance Systems*. Studies in Computational Intelligence, vol 268. Springer, Berlin, Heidelberg.

Pearl J. (1988). *Probabilistic reasoning in intelligent systems: Networks of plausible inference*. Morgan Kaufmann, Burlington MA.

Rumelhart D. E., and McClelland J. (1986). *Parallel Distributed Processing: Explorations in the Microstructure of Cognition. Foundations*. M.I.T. Press, Cambridge MA.

Shumailov I., Schumaylov Z., Kazhdan D., Zhao, Y., Papernot N., Erdogdu M. A., and Anderson R. (2021). *Manipulating SGD with Data Ordering Attacks*. Preprint 2021-04-19, available at https://arxiv.org/abs/2104.09667. Accessed 29th August 2023.

Shepardson D. (2021). *U.S. safety agency reviewing 23 Tesla crashes, three from recent weeks*. Reuters, 2021-03-18. Available at https://www.reuters.com/article/us-tesla-crash-idUSKBN2BA2ML. Accessed 29th August 2023.

Shostak R. E. (1984). *Deciding combinations of theories*. Journal of the ACM, 31(1):1–12, 1984.

Smith T., Barhorst J., and Urnes Sr. J. M. (2010). *Design and Flight Test of an Intelligent Flight Control System*. In: Schumann J., and Liu Y. (editors) *Applications of Neural Networks in High Assurance Systems*. Studies in Computational Intelligence, vol 268. Springer, Berlin, Heidelberg.

This collation page left blank intentionally.

Appendix A. Some IEC Definitions in Functional Safety and AI Systems

artificial intelligence
AI
<discipline> research and development of mechanisms and applications of *AI systems* (3.1l.4)

Note 1 to entry: Research and development can take place across any number of fields such as computer science, data science, humanities, mathematics and natural sciences.

[SOURCE: ISO/IEC CD 2 22989:2022, 3.1.3]

system
set of interrelated elements considered in a defined context as a whole and separated from their environment

> Note 1 to entry: A system is generally defined with the view of achieving a given objective, for example by performing a definite function.
>
> Note 2 to entry: Elements of a system may be natural or man-made material objects, as well as modes of thinking and the results thereof (for example forms of organisation, mathematical methods, programming languages).
>
> Note 3 to entry: The system is considered to be separated from the environment and the other external systems by an imaginary surface, through which pass the links between them and the considered system.
>
> Note 4 to entry: The term "system" should be qualified when it is not clear from the context to what it refers, for example control system, calorimetric system, system of units, transmission system.
>
> Note 5 to entry: This entry was numbered 351-21-20 in IEC 60050-351:2006.
>
> [SOURCE: IEC 60050, the International Electrotechnical Vocabulary, 351-42-08]

system of systems

set of operationally and managerially independent systems that are operated together for a period of time to achieve one or more stated purposes

[SOURCE: IEC 60050, the International Electrotechnical Vocabulary, 871-05-03]

systematic failure

failure, related in a deterministic way to a certain cause, which can only be eliminated by a modification of the design or of the manufacturing process, operational procedures, documentation or other relevant factors

[SOURCE: IEC 61508-4, ed. 2.0 (2010), 3.6.6]

machine learning
ML
process of optimizing *model parameters* (3.3.8) through computational techniques, such that the *model's* (3.1.23) behaviour reflects the data or experience

[SOURCE: ISO/IEC 22989:2022, 3.3.5]

machine learning algorithm
algorithm to determine *parameters* (3.3.8) of a *machine learning model* (3.3.7) from data according to given criteria

> EXAMPLE Consider solving a univariate linear function $y = \theta 0 + \theta 1x$ where y is an output or result, x is an input, $\theta 0$ is an intercept (the value of y where x=0) and $\theta 1$ is a weight. In *machine learning* (3.3.5) the process of determining the intercept and weights for a linear function is known as linear regression.

[SOURCE: ISO/IEC 22989:2022, 3.3.6]

deep learning
deep neural network learning
<artificial intelligence> approach to creating rich hierarchical representations through the *training* (3.3.15) of *neural networks* (3.4.8) with many hidden layers

> Note 1 to entry: Deep learning is a subset of *ML* (3.3.5)

[SOURCE: ISO/IEC 22989:2022, 3.4.4]

artificial intelligence system
AI system
engineered system that generates outputs such as content, forecasts, recommendations or decisions for a given set of human-defined objectives

> Note 1 to entry: The engineered system can use various techniques and approaches related to *artificial intelligence* (3.1.3) to develop a *model* (3.1.23) to represent data, *knowledge* (3.1.21), processes, etc. which can be used to conduct *tasks* (3.1.35).
>
> Note 2 to entry: AI systems are designed to operate with varying levels of *automation* (3.1.7).

[SOURCE: ISO/IEC 22989:2022, 3.1.4]

model
physical, mathematical or otherwise logical representation of a system, entity, phenomenon, process or data

[SOURCE: ISO/IEC 22989:2022, 3.1.23]

Appendix B. Extract from Draft of TR 5469 from June 2022

8 Properties and related risk factors of AI systems

8.1 Introduction

8.1.1 General

Clause 7 describes how the definition of desirable properties is the first step of the three-stage realization principle. The properties are related to topics and eventually to detailed methods and techniques addressing those topics. Acceptance criteria are then identified from the set of the detailed methods and techniques.

This Clause provides guidance on the properties that characterize systems using AI technology and their related risk factors. Such properties and risk factors include degree of automation and control (Clause 8.2), degree of decision transparency and explainability (Clause 8.3), environmental complexity and vagueness of their defining specifications (Clause 8.4), resilience to adversarial inputs (Clause 8.5), system hardware considerations (Clause 8.6) and technological maturity (Clause 8.7).

Details of the properties and risk factors of systems using AI technology, and their related aspects and challenges, are discussed in this Clause.

8.1.2 Algorithms and models

On a technological level, the capability of AI is often achieved by the combination of an algorithm and a model. The model typically represents information that achieves the application's purpose, (e.g. knowledge about how various inputs are to be distinguished and recognized), while algorithms infer information from a model and inputs, (e.g. to make a prediction). This means the functional safety of applications using AI technology depends on both.

Example types of algorithms include linear functions, logical calculi, dynamic Bayesian networks and artificial neural networks. The models can either be handcrafted by an engineer, or can be synthesized from data by machine learning algorithms that themselves use a systematic analysis process. The algorithms are usually implemented as an executable representation, such as machine code (in the case of software), or special hardware, such as field programmable gate arrays (FPGAs).

Usually, algorithms that interact with the models contain only a limited amount of knowledge or implications about the application's goals. This is quite similar to the role of foundational software libraries or programming environments (compilers, etc.) in non-AI software. That is, the algorithm itself does not play a functional safety role, but its correctness is critically important for functional safety to be achieved. In this way, the integrity of algorithms in AI technology can often be handled with existing principles of functional safety as specified in the IEC 61508 series [16]-[19], similar to that of non-AI software components. The same holds for the logic involved in the translation of the algorithm and the model.

By contrast, models often contain knowledge related to the objective of the systems involving functional safety. There are several different ways of constructing models and different approaches can be used for assessing the completion of risk reduction measures to ensure functional safety.

For example, when models are created manually by engineers, the models can likely reflect the engineers' knowledge about the application, which can be assessed during the management processes used within functional safety lifecycles. In these cases, the lifecycle of existing functional safety International Standards can be followed (AI technology Class I as described in Clause 6.2). It is often feasible to create models manually for simple algorithms such as simple linear functions or logical calculi.

In some cases, models derived from data by machine learning algorithms can be analysed and verified after their creation. Alternatively, models derived by machine learning algorithms can be analysed, the underlying parameters extracted and used to extend general engineering knowledge, that, in turn, can be used to develop further models. With the application of validated engineering knowledge, the lifecycle of existing functional safety International Standards can again be applied (e.g. treating these models as AI technology Class I as described in Clause 6.2).

In other cases, models derived from data by machine learning algorithms can be too complex to be understood, analysed and verified. This is often the case for complex types of models, such as neural networks, because representations of models in these types do not necessarily reflect human understanding or reasoning. The use of different approaches for assessing the risk reduction for functional safety is appropriate in these cases, which and can constitute a major challenge for the use of AI technology in implementation of functional safety systems.

ISSN 2754-1118 (Online)
ISSN 2753-6599 (Print)

System Analysis on Driver Monitoring System for Mainline Railway

Niki Mok

TÜV Rheinland UK Ltd.

Abstract

Driver alertness and attention are factors in nearly 50% of Signal Passed At Danger (SPAD) events that could lead to railway accidents. My research aims were to carry out a system analysis on increasing the capability of the existing vigilance system for UK mainline passenger trains to include active detection and actuation based on driver's alertness. The current vigilance system has existed for a long time in UK railway history, and requires optimisation of its original capability. The current vigilance operation can be tricked, isolated, or become a routine gesture. There is a delay time of 60s with 5s action time for the driver to reset the 'lack of activity' trigger, which is far too long for trains operating at a typical line speed of 125mph. This new capability is efficient in reducing the reaction time of detection on driver's falling asleep by 97% (from 65s to 2s) for drivers who cannot be woken by audio alarm, and by 93% (from 30s to 2s) for drivers who can be woken up by a beeping sound. Additional functionalities of the proposed design include detecting early signs of falling asleep, microsleep and eyes-off-road, both intentional and unintentionally.

1 Introduction

1.1 Context

Train drivers bear safety-critical responsibility for hundreds of passengers on mainline routes, especially for high-speed intercity trains.

Among all fatal train collisions and derailment accidents in Europe, human error contributed to three quarters of these accidents, with a significant proportion caused by train drivers (Evans 2017). Within Signal Passed at Danger (SPAD) events[1], 49% were caused by driver alertness and attention factors, with an additional 4% involving drivers being incapacitated or asleep (RSSB 2021).

Drivers are responsible for noting of any danger on the line, such as obstacles, which is challenging when the weather is dark or foggy, or when the driver become drowsy due to lack of sleep associated with shift-work pattern. Most drivers try to carry on working no matter how drowsy they are. 55% of drivers deal with tiredness by caffeine input (RSSB 2005a), which is just a temporary measure.

[1] Note: Appendix B provides a short table of acronyms as a reference for non-railway industry readers.

The existing Driver Safety Device (DSD) monitors drivers' alertness in a passive way by either a deadman's switch or an acknowledge button with a prolonged, one minute, delay time before actuation. Periodic acknowledgement of the button easily becomes a routine gesture. It does not detect microsleep nor distraction. These devices can even be intentionally tricked or deactivated.

A recent fatal accident caused by a driver who experienced microsleep occurred in Croydon, causing 7 passenger casualties, which led to an industrial review for railways. Since then, an anti-sleeping "Guardian Device" entered trials in UK tramways (Townsend 2019), and eye-tracking glasses went under review for ETCS[2] trains (SMI 2014).

Previous research is outdated, and does not consider the capability of the current vigilance system as a whole. This paper thus focusses on improving the capability of the driver monitoring system such that it is fit-for-purpose; it lays the foundation for a future design process from an all-round perspective. It should also be of interest to those in other industries in which continual operator vigilance is important.

1.2 Aims & Objectives

This paper aims to carry out a system analysis to build a foundation for the development of an increased capability of the driver vigilance system for UK mainline passenger trains, to include active detection and actuation based on driver's alertness.

The objectives are to:

- Understand stakeholder needs within the system boundary;
- Establish high-level requirements for both functional and non-functional aspects, including Measures of Effectiveness (MoEs) on usability;
- Develop sub-system measures of performance with traceability to high-level requirements;
- Identify critical interfaces and functional limitations of the system; and
- Explore short-term and long-term potentials of the monitoring system development in the UK.

1.3 Scope

UK mainline railway passenger service trains are chosen as the type of transport for analysis. Metro drivers work in a controlled environment and arrive at a stop every few minutes. Metro drivers are thus less likely to lose awareness due to the higher need to focus on frequent operation tasks.

This paper is applicable for diesel and electric traction trains where drivers work alone and have comparatively modern and comfortable seats. When driving aboard a steam locomotive, a driver is less likely to fall asleep (RSSB 2014).

The proposed driver monitoring system considers only trains in running mode that are moving during normal passenger service. Other scenarios, such as degraded mode where redundancy of main components fails, shunting mode, and emergency situation or service disruption are excluded, because the processes involved are more complicated in a less controlled environment.

[2] European Train Control System

A system analysis approach is adopted in this paper to study driver monitoring systems from a range of perspectives. This is not a comparison review between off-the-shelf devices available on the market. Cost and commercial aspects are excluded.

The analysis focuses on the system front end, including sensing and actuating functions with direct interaction with the users. Backend functions, such as perceiving and decision making, belong to software development and are only discussed at a high-level. Details of how data is stored and manipulated are considered out of scope.

Drivers can suffer from "highway hypnosis" where they lose memory for certain period (RSSB 2014). This state of mind is often encountered with driving on repeated routes. It is a mental stage that can only be detected by brainwave activities, and thus is not studied in detail.

1.4 Method

Background research on the current human factors and safety procedures were reviewed using available resources and reports from institutions and government agencies. Literature review on existing driver safety system was conducted based on conference proceedings, reports from European Union agencies and previously published articles.

Technical review on existing technologies adopted by various industries were investigated through studying journals and academic research papers published by universities, and commercial catalogues.

System analysis using a capability systems engineering approach was adopted. Ordinary office tools such as Microsoft Word, Excel and Visio were used, as well as purpose-built toolsets such as Sparx Systems Enterprise Architect for developing matrices, diagrams and modelling (Graves 2009).

2 The Importance of Vigilance

2.1 Background

Human factors contribute to 75% of fatal train accidents in Europe (Evans 2017). Train drivers' errors, such as SPADs and over-speeding, contribute together to 44% of the total number of train collisions and derailments (Evans, 2017).

A recent study found that 49% of those SPAD incidents are due to drivers' inattentiveness and lack of alertness (RSSB 2021). Another 4% even involve drivers being incapacitated or asleep. According to Reason (1990), human error, including attention distractions, is one of the reasons leading to occurrence of accidents.

Train drivers work on shifts with either early starts or late finishes at midnight. A survey in Canada (Nicol and Seglins 2014) reported that 3 out of 4 train operators admitted that they feel exhausted and encountered "nodding off" while driving. Tiredness due to high workload and boredom due to lack of tasks required are undesirable extremes created by the nature of train driving. Distraction and fatigue are the most common causes of driver's error.

As part of the research, a driver's cab walkthrough was originally considered one of the good methods to understand the driver's operation. However, due to the recent pandemic physical visits were not allowed during the period of when this paper was produced.

Eventually, I interviewed a train driver[3] who had worked for one of the largest UK Train Operating Companies (TOCs) for about four years to understand their work shifts pattern, and challenges faced in their daily work[4]. He expressed that he easily feels tired after a few days of shifts with early-starts time, or late finish. Drivers can report sick to a Driver Resource Manager (DRM) if needed, however, in reality they tend to just "*drink coffee and get on with it*". Tiredness can sometimes become a challenge, especially when there is no person to talk to, and the consequence might even be that he misses a station.

In fact, a separate survey found out that caffeinated drinks were used as a method to deal with fatigue in UK railway by 50% of the train drivers, and around 5% used caffeine tablets (RSSB 2005a).

2.2 Existing Technology

To ensure that drivers are alert and fit for work, driver's vigilance systems are a complementary safeguard to train protection and control systems (RSSB 2014). SPADs are effectively managed by a Train Protection and Warning System (TPWS) that applies emergency brakes automatically (NRIL 2017).

The DSD, or deadman's switch, has been adopted in Great Britain on locomotive driving cabs for over 50 years (RSSB 2014). All manually driven trains are required to be equipped with a dead person's device, or vigilance system, for mass transit railway and high-speed trains in UK and Europe; see BS EN 13452-1:2003, BS EN 15734-2:2010 and (EU) No 1302/2014. The system checks the driver's vigilance to detect any loss of consciousness with linkage to the train's automatic stop function. If no pressure is applied on the pedal or the handle, the brakes are applied after a pre-set time period bringing the train to a stop.

The vigilance system function is achieved by collecting data from the DSD, deadman's handle or foot pedal, Automatic Warning System (AWS) acknowledge button, and movement of the power and brake controller. The driver's activity is monitored by also noting their action on the Train Control and Monitoring System (TCMS) as per BS EN 14033-1:2017.

A Driver Vigilance Device (DVD) steps in when the DSD is not reset by the driver within the given timeframe. Designed according to standard BS EN 14033-1:2017, and the Locomotive & Passenger Rolling Stock Technical Specification for Interoperability (LOC&PAS TSI)[5], the required operating principle is based on a multiple step process starting from monitoring the driver's vigilance (See Figure 1, which is based upon the standard and TSI just referenced).

If no action is detected for 5 seconds, vigilance monitoring starts. After 20 seconds of inactivity, a visible alarm flashes, followed by ringing of the bell (EKE-Electronics Ltd.

[3] The interview with the train driver was conducted during 12th to 30th July 2019 via Facebook Messenger, an informal social media platform for communication.

[4] Note that evidence from a single informal exchange of messages with one train driver does not constitute a reliably accurate portrayal of the effects of fatigue on the driving community.

[5] See (EU) No 1302/2014.

2017). During the typical 5 to 7-second action time the foot pedal must be released and depressed in order to stop the countdown. If delay time exceeds 60s, automatic stopping of the train is involved; see BS EN 14033-1:2017, GMRT2185 and (EU) No 1302/2014. A message is sent to the signalling centre via GSM-R[6] radio system (The Train Guard 2014). An extraction of the relevant clauses can be found in Appendix A.

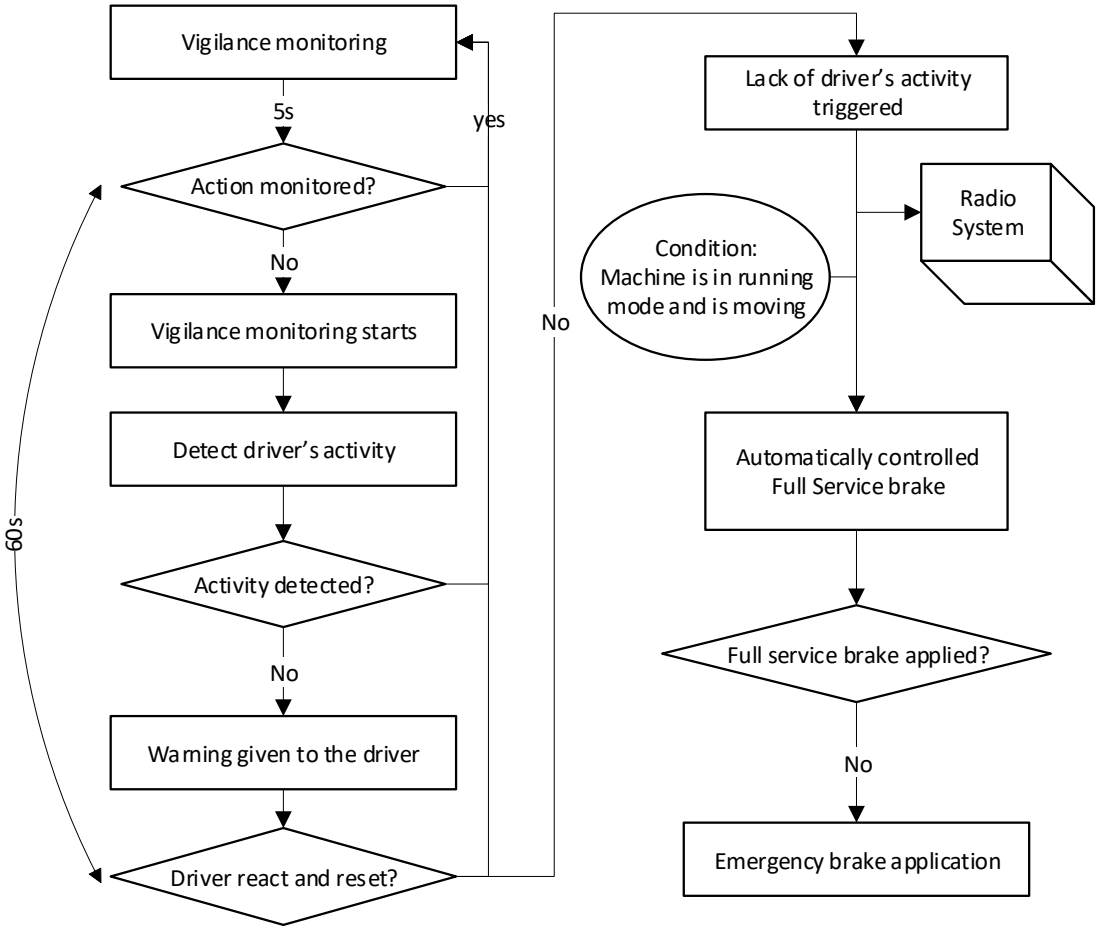

Figure 1 ~ Logic Flow of Current DVD Working Principles

An automatic watch device, as part of the DSD fitted on control panels used in UK mainline railway, include a push-button, a pedal on the floor, an acoustic electric warning device, and a warning light. The system also includes multiple speed measuring devices and a timer device (DEUTA-WERKE 2019). The vigilance system is wired to both the vigilance isolation switch and the vehicle brake circuit.

2.3 Inherent Shortcomings

The current requirement of 60s for determining lack of driver's activity is considered too long compared to the speed of modern trains, and the reaction time required for other driving cab controls. UK mainline trains, such as the Pendolino Class 390, runs at 125mph (201km/h) (de Castella 2015) and so travels 280m in 5 seconds.

[6] Global System for Mobile Communications-Railway

Longer reaction time is related to driver fatigue level. The delay time of a minute is also far longer than the required driver's reaction time for acknowledging a safety system massage, such as TPWS horn, which is 2.5 seconds. This reaction time for the train protection system was set to be short enough to avoid an error leading to a braking application (El Rashidy and Van Gulijk 2016).

The device can be isolated during operation if it fails, only if a trained Guard or second driver is present to operate the emergency brake if the first driver loses consciousness (The Train Guard 2014). The driver can isolate the driver's vigilance equipment when the warning cannot be reset (RSSB 2018).

A deadman's pedal is not reliable on detecting driver's attentiveness; for example, some drivers who find them annoying would override the system illegally, freeing themselves for other activities. To overcome its weaknesses, a vigilance control, the DVD, was fitted as an automatic watch device. Vigilance buttons are to be pressed periodically and can be activated to remind the driver when stopping at a red signal. However, DVD is also not fully effective because driver can disable it without knowing (RSSB 2014).

For certain forms of sleep, such as microsleep, which occurs when the driver becomes inattentive, the deadman's handle does not in all cases result in the train being stopped. Microsleep has been the immediate cause of a number of derailment accidents in history. One example was in April 2003 at Apeldoorn in the Netherlands (RSSB 2014) the train was fitted with a deadman's handle, however that did not result in the train being stopped.

2.4 Accidents

In 2016, the Croydon tram collision led to 7 people killed. The Rail Accident Investigation Branch (RAIB) report noted that, "*In common with most trams and trains in the world, there was no device fitted that was capable of reliably detecting drivers' loss of awareness*" (RAIB 2016).

The accident report revealed that, although the driver had already lost consciousness, there was still a downward pressure applied on the Traction/Brake Controller (TBC) while he approached the junction. The fact that the DSD has not been disabled (RAIB 2016) indicated that the DSD system had an inherent design defect.

Another catastrophic accident, the Santiago de Compostela derailment, occurred in 2013 and led to eighty fatalities in Spain. The driver claimed that he was distracted as he was talking on the telephone with staff at the control centre (Langer 2016). Another incident happened in the Hunter Valley, Australia in August 2013 where two coal train drivers employed by Pacific National disabled the vigilance device and read newspaper while driving (RSSB 2014).

Based on the above, it seems that neither the deadman's switch nor the driver reminder appliance can avoid microsleep, or intentional distraction such as reading a newspaper or browsing mobile telephones. These accidents suggest that the current safety devices are out of date, and need to be revisited and re-designed to meet the required safety standard.

After the Croydon Tram crash, an anti-sleeping device, The Guardian Device, was developed to be installed to track driver's facial features real-time to recognise events such as microsleep and distraction (Townsend 2019).

Although light rail driving relies on line-of-sight, it still raises the question whether mainline trains require a similar device that monitors driver's fatigue and distraction. This paper therefore looks at how one can improve the capability of the current vigilance system.

3 Literature Review

3.1 Driver Inattentiveness

3.1.1 Driver Tasks

Train drivers perform several tasks during their shifts. After a train's departure, they operate the train on the set route, respond to signals, obey speed restrictions, and comply with a common set of railway rules and regulations, which are provided by the rail company or their regulators. Additionally, drivers have to stop at the stations on the route and keep to the timetable as strictly as possible. They also stay in contact with control rooms, station staff, and on-train colleagues to report any problems on the line or train (Kyriakidis 2013).

Cognitive ability of drivers is exceptionally important for road and train drivers. Both require high attention, visual perception and reaction ability (Guo et al. 2019). Train drivers work under a high level of stress due to more stringent safety requirements, and they are responsible for hundreds of passengers' lives. Yet train drivers follow tracks, without the need of changing lanes or using steering wheels (Guo et al. 2019), thus they are more susceptible to loss of concentration due to the routine driving tasks.

3.1.2 Levels of Driver Readiness

Driver's readiness can be categorised into different levels of arousal, as in Table 1, which is derived from Whitlock (2002). Level 5 indicates the driver is at full alertness and is performing all necessary operating tasks. Level 4 means the driver is multi-tasking but is still sufficiently alert. For level 3, although the driver is sufficiently alert, he or she is not attending to the train operating tasks. Levels 2 and 1 of deficiency in driver's readiness are what this paper is mostly interested in. Drivers at level 2 are awake, but drowsy and not sufficiently alert. Drivers at level 1 are incapacitated, asleep or dead.

Table 1 ~ Level of Driver Readiness in Decreasing Order from 5 being Attentive to 1 being Inattentive

Level of Arousal	Not Inappropriately Time-sharing between Operating Tasks	Attending to the Train Operating Tasks	Sufficiently Alert	Alive and Awake
5	✓	✓	✓	✓
4	×	✓	✓	✓
3	×	×	✓	✓
2	×	×	×	✓
1	×	×	×	×

3.1.3 Fatigue

To understand how to detect driver's fatigue, it is critical to understand the explicit signs that occur when a driver feels extremely tired. A fatigued driver shows symptoms such as

repeated yawning, feeling irritable, delayed reactions, daydreaming, difficulty in keeping the eyes open, shallow breathing (Meiring and Myburgh 2015) and blinking rate reduces (Haq and Hasan 2016). Increase in Percentage of Eye Closure Over Time, excluding the time spent on normal closure, is a wide known measure of the drowsiness (Dinges et al. 1998) that characterises driver fatigue (Ji and Yang 2002).

3.1.4 Distraction

Distraction can be visual, cognitive, auditory, or biomechanical (Meiring and Myburgh 2015) or a combination of these stimuli. Modern trains are equipped with a Driver Advisory System (DAS) with enhanced functionality. Increased usage of information display units, such as DAS on energy-saving suggestions, causes drivers to be required to multitask while driving, leading to potential distraction.

We can thus predict that the common methods for detecting driving fatigue and distraction are monitoring of eyes movement, head tilt and position, facial expression, and change in heart rate (see Figure 2, which is based on Meiring and Myburgh (2015) and Haq and Hasan (2016)).

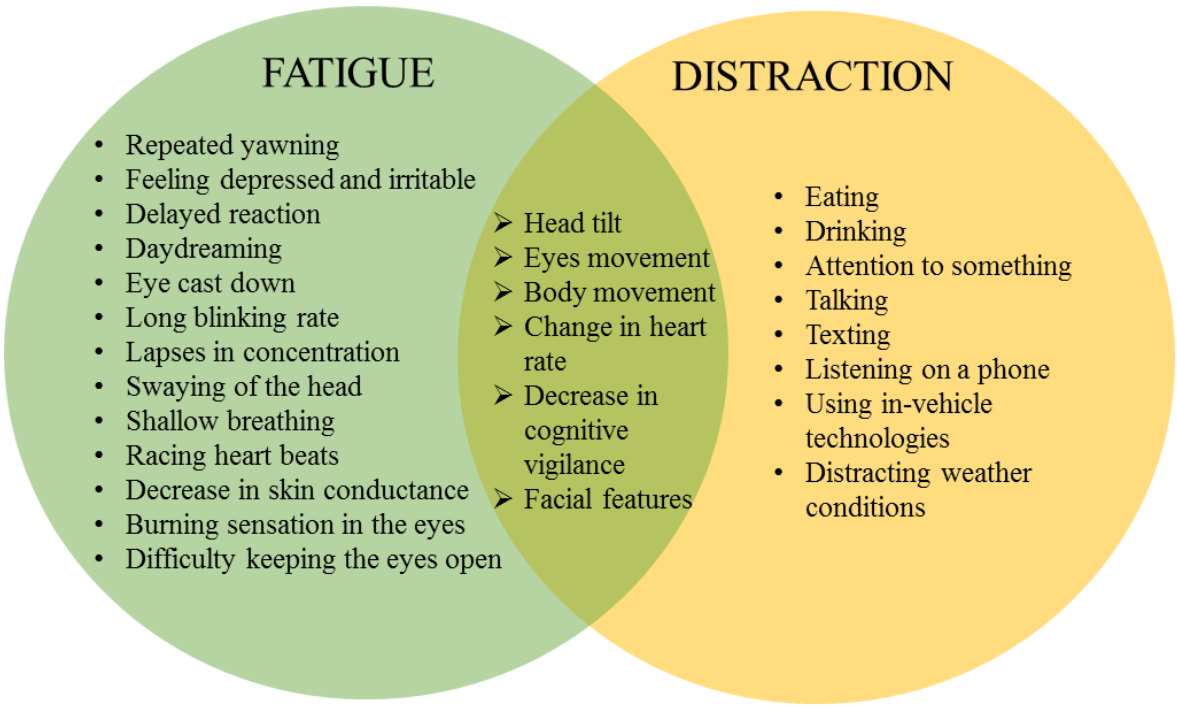

Figure 2 ~ Common Characteristics of Fatigue and Distraction

3.1.5 Highway Hypnosis

When driving on repeated routes, drivers can suffer from highway hypnosis, wherein they lose memory for certain periods (RSSB 2014). There is minimal safety hazard for road vehicles in this scenario, because the driver can still react to stimulus (Commissaris 2019). There is a certain degree of safety implication for railway drivers to lose location awareness, because they rely on route knowledge for interpretation of signals and signs ahead (Whitlock 2002).

Studies on the correlation between brainwave activity and drivers who ‘zone out’ are undertaken in order to create a warning system (Magazine Monitor 2013). Although

considered as one of the distracted driver's phenomena, highway hypnosis is a mental stage, rather than possessing physical symptoms, therefore is not discussed in detail in this paper.

3.2 Previous Work

The Rail Safety and Standard Board (RSSB) carried out a detailed research on DVDs (Whitlock 2002) nearly two decades ago. Thirteen technologies were explored using feasibility criteria with three technologies deemed applicable, but yet to be used in the railway industry. The scientific criteria provided were at qualitative level where they cannot be measured and verified.

Another study regarding driver alertness monitoring system was issued for ScotRail (RSSB 2014). It was concluded that there are concerns over different kinds of devices, such as the intrusiveness of a camera. Train controls, including AWS, TPWS and eventually the European Railway Traffic Management System (ERTMS). with existing vigilance systems already provide a certain degree of protection. Further development is needed to justify the balance between vigilance system and control system.

The above studies only compared different commercial-off-the-shelf products. Another paper on fatigue detection technologies provided an all-round analysis of driver-state monitoring systems, however, it focus on road vehicles rather than rail (von Jan et al. 2010).

This paper therefore conducts a comprehensive analysis considering the whole problem using systems engineering approach with measurable requirements that can be verified.

3.3 Monitoring Systems

3.3.1 Rail

The European Union Agency for Railways explored the potential applications of measuring the level of competence of train drivers by monitoring eye movements to observe their behaviour (D'Agostino 2016).

The Croydon, UK, Tram crash in November 2016 killed 7 passengers and caused serious injuries to many others. The RAIB recommended safety measures, including installation of a monitoring system to detect the onset of fatigue or distraction of tram drivers (RAIB 2019).

To improve safety after the 2016 Croydon incident, London Trams and Tram Operations Limited announced the implementation of an anti-sleeping tool called "The Guardian Device" for tracking symptoms of real-time fatigue and distraction of tram drivers (Townsend 2019). When these symptoms are detected, an audio alarm and seat vibration are activated to alert the driver. A few seconds prior to the alarm, the recorder is switched on to allow incident investigation. Techniques involve tracking of micro-movements of a driver's eyes, head and facial expressions to identify driver's behaviour.

The Croydon tram was fitted with Guardian driver monitoring device 8 months later, as a result of the accident that happened the previous year (Booth 2017). The device is composed of four hardware components, namely the in-cab guardian sensor, a forward-facing camera, a computer/controller, and a vibration motor (Seeing Machines 2019). The installation is pictured in Booth (2017).

The infrared sensor can detect through sunglasses; the system assesses drivers' fatigue and distraction by eye, face and head position. An audio alarm goes off when the driver is looking at the sides for certain period of time. The seat vibrates when eyes are closed for a few seconds, which indicate that the driver is under fatigue.

3.3.2 Road

Technologies for predicting driver's alertness are evolving in road vehicles. Singapore Mass Rapid Transit trialled a biometric device to collect bus drivers' behaviour for analytic purpose, identifying high-risk drivers (NEC 2014). The company engaged data scientists to observe the drivers' eyes and concentration levels. However, the bus company did not allow the use of cameras, so instead they used telematics data from road performance data and driver training records to predict the bus drivers' behaviour (Griffith 2017).

Facial recognition technology developed by Microsoft was adopted in Bangkok, Thailand, to establish a programme called "AI for Road Safety". Drowsiness (evidenced by eyes closing and blinking time) and distraction (evidenced by yawning, eyes-off-road and mobile device using), detected by computer vision, are sent for interpretation using machine learning (Microsoft 2019) to alert drivers and to inform risk Key Performance Indicators.

3.4 Monitoring Technologies — Sensing Devices

3.4.1 Video-based Sensors

Images of the driver are captured for analysis through computer vision processing to identify driver's behaviour and events. These technologies can be categorised into eye movement, facial features, and head position detection.

Eye Movement: Eyes blinking and position are used as the major detection means of driver state in the automotive industry. Looking away for at least 1.8s can be considered a driver visual distraction for road vehicles (Tango and Botta 2013).

The sensor is integrated into the drivers' cab of the vehicle. The advantage of this method is that it does not have to be a wearable device, nor even have any physical contact with the user. As a result, the accuracy depends highly on the maturity of the image processing technology. To ensure high reliability and sensitivity, there needs to be robust algorithms to avoid false alarm. It also needs to overcome different lighting conditions and physical characteristics, such as eye colour and spectacles. Therefore other physiological measures are also used to provide additional confidence and indications (Tango and Botta 2013).

The Guardian device uses infrared sensors, such that it can see through sunglasses (Seeing Machines 2019), and in the dark. The amount of infra-red light is so low that it is less than 2% of the sunlight that reaches the driver's eyes and face (Murray 2017).

Apart from remote sensing, eye tracking glasses are also available for drivers to wear, such that their eye movements can be tracked with a wearable pair of glasses. SMI Eye Tracking Glasses are equipped with insertable sunglasses and corrective lenses (SMI 2014). However, this is a wearable device that may become an extra burden to drivers who do not usually wear corrective spectacles.

Facial Features: Driver drowsiness level can be measured using facial expressions, including inner and outer brow rise, yawning, jaw drop as well as lip stretch (Doudou et al. 2018). Behaviours including yawning and talking on the telephone can be captured by the feature extraction technique.

Head Position: Head tilting down and head nodding are signs of drowsiness (Doudou et al. 2018). Head position detection is often used in combination with eye movements as seen in the Guardian device implemented for the Croydon trams (Seeing Machines 2019). Head nod events alone are not a good indicator of drowsiness, because microsleep occurs without head nods (Whitlock 2002).

No matter what the target objects are, either they are the eyes or the head, camera-based technologies capture images of a person's face, therefore are intrusive in nature, and can make one feel that they are under surveillance. They also require robust algorithms to deal with variable sunlight, driver's spectacles, and other "noise sources".

3.4.2 Telematics Recording

Telematics recordings are of vehicle-based sensors that detect a train's dynamic data, such as acceleration and speed. Research found that there are correlations between a driver's alertness level and the speed of the vehicle, the acceleration, and pressure on the acceleration pedal. Drowsy drivers tend to increase acceleration (Doudou et al. 2018) and increase speed variability (Dorrian 2008).

A telematics system has been implemented for bus drivers that notifies them when they accelerate or decelerate quickly, or overspeed for a prolonged period. The system is able to identify high-risk driving events using data collected from driving behaviour and historical route-related information (NEC 2014).

Telematics records can be analysed and used to improve driving performance in a similar way in the railway context. The Institution of Railway Research (IRR) studied the applicability of assessing driver competence performance using real-world On-Train Monitoring Recorder (OTMR) data. Originally recorded for accident investigation purpose, these data include indicators such as Emergency Bypass Switch, Driver's Reminder Appliance and Driver reaction time (El Rashidy and Van Gulijk 2016).

3.4.3 Physiological Signals Sensors

Physiological signals from human organs such as brain, eyes, muscles, and heart are collected to determine wakefulness and fatigue level (Doudou et al. 2018). The advantages of physiological based technologies are that they are more reliable and accurate, and have lower level of noise and artefacts. It is also possible to detect early signs of drowsiness, and thus give alert prior to any accident happening (Doudou et al. 2018).

Skin Conductance: An electrodermal wristwatch operates by measuring skin conductance response. The Vigilance Telemetric Control System (Neurocom 2019) uses galvanic skin resistance to predict driver's alertness. The Steer wearable device (Created Mode 2020) also uses electrodermal activity, but is worn on the finger rather than the wrist. Steer warns the driver when skin conductance decreases, which indicates fatigue. It was later found that the particular electrodermal activity monitor worn was not responsive enough to be able to detect sleepiness when the subject falls asleep.

Respiration Rate: Respiration rate sensing device measures respiratory effort, nasal and oral airflow, and blood gas (Doudou et al. 2018). An anti-sleep driver seat sheet (Toto 2012) has built-in pulse monitoring and respiration sensors installed. This Sleep Buster originated from Japan; it measures the driver's bio signals every 18 seconds and can warn drivers 10 minutes before they fall asleep. This device is yet to be proven on the market. According to academic research, these sensors are difficult to use in real-time due to a delay in detection of sleepiness (Tango and Botta 2013).

Heart Rate: Cardiac activity can be monitored through electrocardiography and blood pressure signals. These technologies are used in lie detectors, or polygraph, by the police and other agencies, but are not common in society. Sensed every 2 seconds, when heartbeat is detected to be lowered by 10 units, the STEER wearable device warns the driver about increase in fatigue level (Created Mode 2020).

Brain Signals: Detecting brain signals by a brain waves cap (Deayea 2019) and wireless wearable electroencephalograph (Zhang et al. 2017) are plausible methods; they need to be worn but do not accommodation religious head-dresses — or drivers' uniform hats.

To summarise, physiological signals sensors require certain amount of physical contact with the user. These technologies are thus more intrusive and invasive, and are more difficult to use (from the point of view of the user) compared to vehicle and telematics sensors (Doudou et al. 2018). Built-in sensors, e.g. installed inside the driver's seat, are less invasive and provide early warning, however, this technology development has not been promoted to widespread usage yet due to a low response rate.

Note that physiological sensors do not detect distraction, unlike camera-based sensors which serve both purposes for drowsiness and distraction identification.

3.4.4 Mobile Telephone Detectors

To prevent drivers from being distracted, usage of mobile telephones in the cab is restricted (ORR 2010). Mobile telephone detection technologies can be used to ensure compliance with regulations. These devices can distinguish signals from mobile telephones and from the train and notify a control centre when mobile telephone signals are detected (Wi-Tronix 2016).

3.4.5 Cognitive Vigilance Device

In Australia, drivers are asked to carry out cognitive task, e.g. mathematical calculations, to confirm their cognitive vigilance (RSSB 2014). This type of vigilance approach, however, imposes distraction to the drivers to respond to their operating task. A report concluded that these devices are only suitable for train driving with high automation where trains travel at fixed speed for prolonged period, which is not in case in UK and Europe (RSSB 2014).

A study also suggested that secondary tasks are not suitable to be used as DVDs because the drivers get used to the task, and do not stay alert anymore after a period of time (Dorrian 2008).

3.4.6 Summary

Video-based sensors can be used for both distraction and drowsiness detection. These are more suitable than physiological signals sensors, of which the majority are only effective

for alertness prediction. Physiological sensors outperform the visual techniques on accuracy; however they lack sensitivity and inhibit drivers' operation as they are mostly wearable. Telematics recording can be used as an extended function for estimating driver performance. Mobile telephone detectors can be fitted in the cab to verify the use of mobiles, without interfering with drivers' privacy.

3.5 Monitoring Technologies — Actuating Devices

3.5.1 Visible

Flashing lights and warning messages on a display are common types of warning system. Drivers' eyes are closed when they lose awareness, and thus are less sensitive to light level changes. Therefore, visible alarms can only be used as an initial reminder, as illustrated with the STEER wearable device (Created Mode 2020), but not the main form of alarm to restore wakefulness.

3.5.2 Audio

Audio alarms exists in multiple forms, such as a buzzer or an announcement. Drivers easily get used to multiple ringing and repeated messages. Recorded audio announcements can be used to remind a driver of the importance of keeping vigilant and to seek for a break if necessary. Such a recording can be embarrassing if heard by passengers, which can ultimately affect the TOC's reputation.

Another type of audio reminder is a telephone call from the control centre. Drivers talking on the telephone can be easily distracted, as in the Santiago accident in Spain (Langer 2016). It is thus deemed not suitable as a vigilance alarm during normal operation. It can be used as a confirmation from a driver's manager at the control centre to check if the driver needs a break, and to assist with necessary arrangements for stopping of the train.

3.5.3 Vibration

Various forms of vibrating alert technologies are available on the market. A fatigue detection system research project used a massage chair for experiments, then designed driver seats with built-in vibration motor (Zhang et al. 2017). Vibration seat technology is also used in the Guardian device on the Croydon trams (Seeing Machines 2019).

The actuating device can be worn on the driver's arm as in STEER wearable device (Created Mode 2020), or on fingers as in Anti-sleep alarm (Stopsleep 2020). These wearable devices are not preferred, because they hinder body movements when performing driving tasks.

3.5.4 Electric Shock

Electric shock has been used as an alarm clock to train people to wake up at a particular time. Eventually people develop a habit of waking up on time, probably due to a fear response. Shock Clock invented by Pavlok has a three-step alarm starting with vibration, then beep and finally "zap". This technology has then been adopted in road vehicle for drivers such as the STEER wearable device (Created Mode 2020). Electric shock technology is controversial, especially if becomes mandatory, because of ethical issues.

The voltage, however mild, causes discomfort, and therefore is unacceptable for the purpose of maintaining a train driver's vigilance.

3.5.5 Summary

Visible and audio alarms are basic forms of warning used in the existing vigilance systems, together with automated application of brakes as the last resort. Audio alarms can be used to alert distracted drivers when they look away for more than 2 seconds. Visible alarms are less useful for distracted drivers who may be looking away from the flashing light. A buzzer sound is better than a recorded message, as it serves the purpose without the passengers realising the associated meaning.

Microsleep occurs typically between 0.5 to 30 seconds (Poudel et al. 2014). Drivers who experience microsleep do not react to sound or light (Created Mode 2020). It would be beneficial to install vibration alarms for drowsy drivers to increase their wakefulness. The control centre can give the driver a telephone call to establish whether the driver is fit to continue.

3.6 A Driver's Opinion

From an interview with a Thameslink driver, he felt that the Guardian device on Croydon Trams was an invasion to drivers' privacy, and felt like this is a device for surveillance purposes. The drivers' union has also argued that the fundamental issue needs to be dealt with first, which is long shifts and health regimes, before such implementation(LBC 2018). The drivers are concerned with long periods of time being exposed to infrared beams. Some drivers complained of headaches, dry eyes, and blurred vision while being illuminated by infrared beams (Murray 2017). However, the spokesperson from Seeing Machines claimed that their infrared beam has no health risks to humans.

In a customer opinion survey, it was found out that a doze-off alarm is thought more important than a fatigue detector, and there is a high customer acceptance of technology in Germany (von Jan et al. 2010). The users' needs have to be satisfied to ensure that the system provides tangible support.

Train drivers' opinions are key in any modification to the drivers' cab. There are three main train drivers' unions in the UK, which have tens of thousands of members. They are namely RMT, ASLEF and TSSA [7]. Changes made to a train driver's working environment has to go through Rail Delivery Group (RDG) and representatives from these trade unions.

3.7 Ergonomics

Human factors aspects of the driver's seats associated with modifications to install a built-in vibration motor need to be assessed to ensure that the seats remain adjustable to accommodate a range of body dimensions.

For video-based equipment during operating tasks, it is fundamental to ensure the field of vision of the camera covers the range of movement of the driver.

No foreseeable changes to the cab workstation layout are identified, except for the new equipment that are to be fitted in the cab, such as the camera and processing unit.

[7] The National Union of Rail, Maritime and Transport Workers, Associated Society of Locomotive Engineers and Firemen, and Transport Salaried Staffs' Association, respectively.

It is also important to communicate with the users about any health concerns or uncomfortable feelings that they might encounter with sensors that use infrared technology for facial feature extraction.

3.8 Software Development and Data Analysis

3.8.1 General

A software processor uses various computer vision techniques to process images captured from the sensor to identify drivers' behavioural events that indicate drowsiness and distraction.

Some technologies, such as Convolutional Neural Networks (Masood et al. 2018), show nearly perfect accuracy in distraction recognition, and identification of the cause of careless road driving, especially on use of mobile telephones.

Other machine learning approaches have been implemented in recognising driver fatigue and distraction, and used in manoeuvre detection, and identification of driving behaviour and performance (Meiring and Myburgh 2015).

The Sensitivity and Specificity requirements of the proposed operating concept will be described in Section 5.3 of this paper.

3.8.2 Limitations

Occasional inaccurate outputs generated from software predictions are expected in the early stages of development. False-positive occurs when a driver is misunderstood by the model predicting as having a fatigue or distraction event when they have not. The driver could become discontented about the nuisance caused by the frequent alarm despite him maintaining alertness. To prevent the driver from losing confidence in the new system, events have to be verified to "train" the system's accuracy.

False-negative, on the contrary, is an event being missed, possibly because the sensitivity of the system is not good enough.

3.8.3 Validation

Performance of a software model can be evaluated by a number of validation methods to ensure a good quality of prediction algorithm model that is fair, robust, and is able to protect privacy.

Sensitivity analysis can be carried out (RSSB 2019) together with data review, validation testing, stress testing, and independent review to validate the robustness of the algorithms.

3.8.4 Security and Resilience

With the development of modern technologies, concerns are raised over digital resilience and cyber security. Vigilance systems, alongside other railway subsystems, contain user information and are prone to digital threats. In order to maintain high availability, integrity, and confidentiality, the technology system needs to be well managed to allow quick recovery when dealing with unforeseeable situations. Vigilance system, if installed

with a camera, possess privacy information of train drivers and personal data needs to be protected.

The ISO/IEC 27000 series of standards for Information Security Management Systems form the basis of cyber security for organisations across industry. The IEC 62443 series of standards, on security for industrial automation and control systems, sets out requirements on network and information system. Implementation of changes and system architecture are to be reviewed against CENELEC TS50701, on Cyber Security for Railway Applications, to conform with cyber security management system to identify all threats and mitigation measures.

3.9 Regulatory Challenges

The safety management system conforms to regulatory and legislative frameworks. There can be legal obligations for different parties such as supplier, operating and maintaining of the asset, as well as those responsible for regulating and supervising these activities. It is crucial to identify the areas of overlap between these regulations and the "prediction of driver's risks using machine learning" boundary. The railway industry needs to identify the impact of this change to ensure its implementation meets legal obligations and abides by the Common Safety Method on Risk Evaluation and Assessment (CSM-RA) from (EU) No 402/2013.

Before entering the trial period, any change which may affect safety needs to demonstrate that the resultant risk is acceptable (ALARP). An independent Assessor Body has to be appointed to demonstrate its fulfilment with CSM-RA if the change is deemed significant. Technical and Operational Standards needs to be met in accordance with TSIs, national safety rules and other relevant safety requirements, with approval from the Office of Rail Regulation (ORR).

The DSD system Safety Integrity Level (SIL) should be appropriate for the level of risk being controlled. The functions developed to the SIL level will require an appropriate safety case. Changes on any of these functions need to follow a formal change management process and CSM-RA for approval by the authorities.

4 Method

The vigilance system requires optimisation and extension of its original capability. The current vigilance operation can be tricked, isolated, or become a routine gesture. There is a delay time of 60-second with 5-second action time for the driver to reset the "lack of activity" trigger, which is far too long for trains operating at a typical UK line speeds of 125mph.

The aim of this paper is to improve the capability of the current system, rather than replacing with a new one. Therefore, a Capability Systems Engineering approach is used for this system analysis.

Capability Systems Engineering is commonly used in the military industry and fits into the decomposition stage of the V-lifecycle (Kemp and Daw 2014). Unlike products or systems concerned with outputs or performance, capabilities are more focused on delivering outcomes or effects. The benefits can be understood by realisation through people, processes, information, and equipment. The respective four principles are shown in Figure 3, which is derived INCOSE UK guidance (Kemp and Daw 2014):

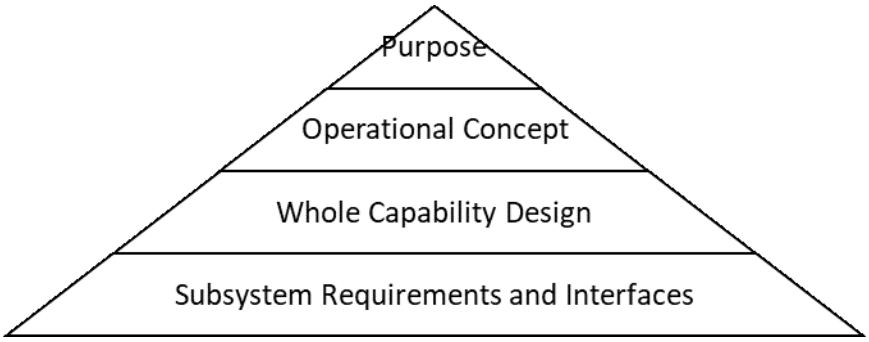

Figure 3 ~ Four Stages of Capability Systems Engineering

The first stage is to establish a clear single statement of purpose and define the scope. Through identification of stakeholders, the client's needs are understood and can help develop the MoEs. These high-level requirements elicited in the MoEs matrices serve as the output of the foundation of future driver monitoring system design process.

Then an Operational Concept is developed to describe the answers to what, why, how, where, when, and who of the new capability.

Figure 4 on the next page (based on INCOSE UK guidance (Kemp and Daw 2014)) shows the process flow of the method used for this paper.

With the aid of Model Based Systems Engineering technique, Use Cases and Activity Diagrams are produced to aid understanding using software tools such as Visio and Enterprise Architect. An example Use Case is presented in Figure 5 overleaf.

In the Whole Capability Design stage, the requirements are verified to ensure that they are captured as required. Preliminary subsystem requirements and interfaces are then developed and linked back to the MoEs for traceability.

5 Assessment / Analysis

5.1 Purpose

The purpose is to improve the capability of the driver's vigilance system on UK mainline railway. Capabilities of how the vigilance system can detect driver's distraction and fatigue, via physiological or cognition state, and actuate accordingly to prevent undesirable events from occurring is being explored.

5.2 Identification of Stakeholders

The purpose of enhancing the Driver's Vigilance System is ultimately to improve the safety of railway passengers. The travelling public are stakeholders that the railway aims to satisfy.

The primary users of the system are train drivers, as employees of the TOCs in UK mainline railway. The initiative can be driven by regulatory bodies such as RSSB, who define the set of system requirements. The drivers are the end user and are the main stakeholder who needs to be kept satisfied.

The secondary users are staff based in the control centres to which the system reports. They receive real-time intervention messages, and regular event summary reports.

The tertiary users include project engineers, who design and implement the system according to requirements, maintain the system to ensure availability, and support in-service system updates.

The TOCs, who are the proposer of the change, need to satisfy the train drivers and their trade unions. The TOCs need to ensure that the system manufactured and installed by the Supplier meets all the requirements, and ensure compliance is met such that approval from the regulator, ORR, can be gained. This requires communication and balance between the interests of the different parties.

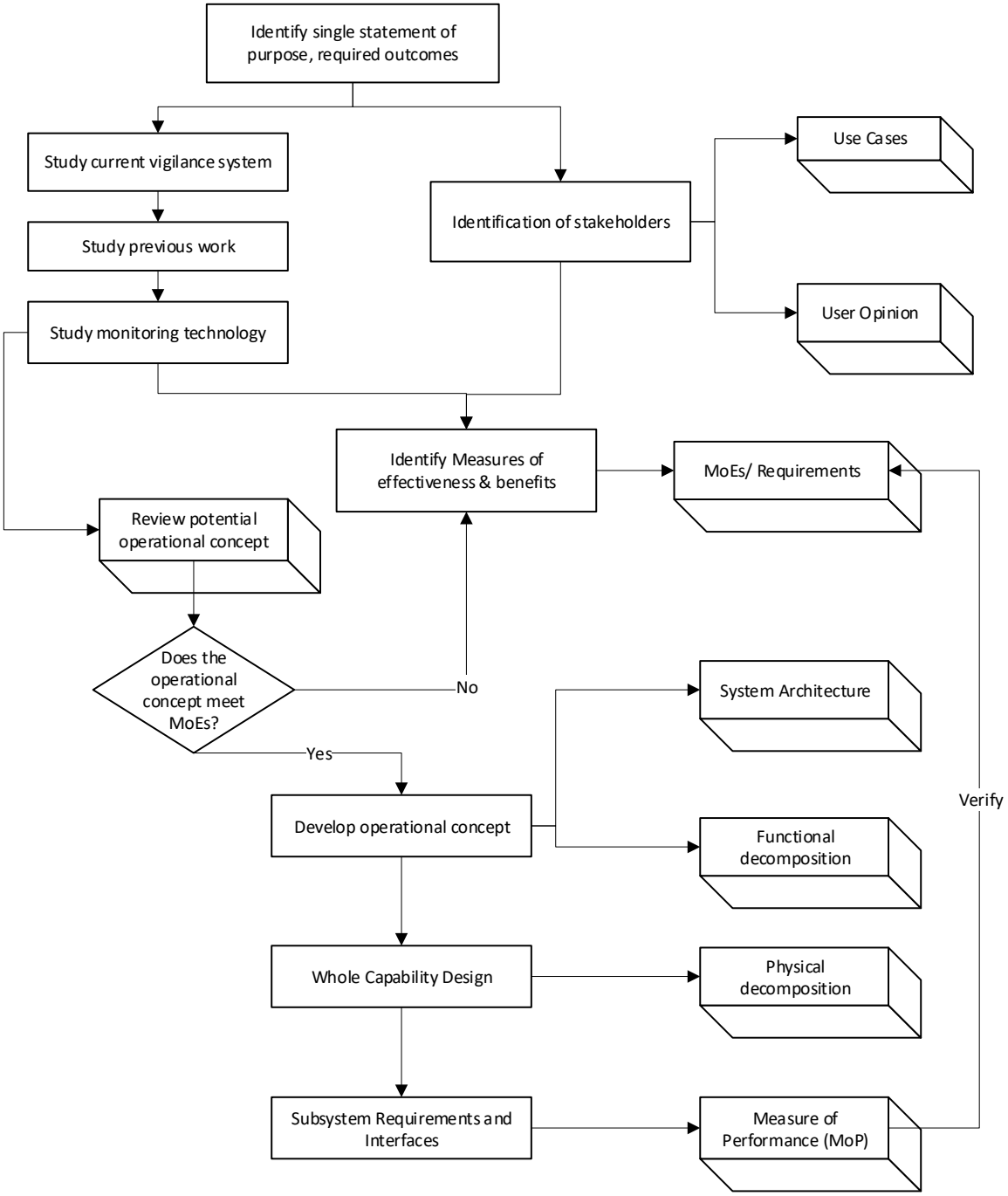

Figure 4 ~ System Review using Capability Systems Engineering Approach

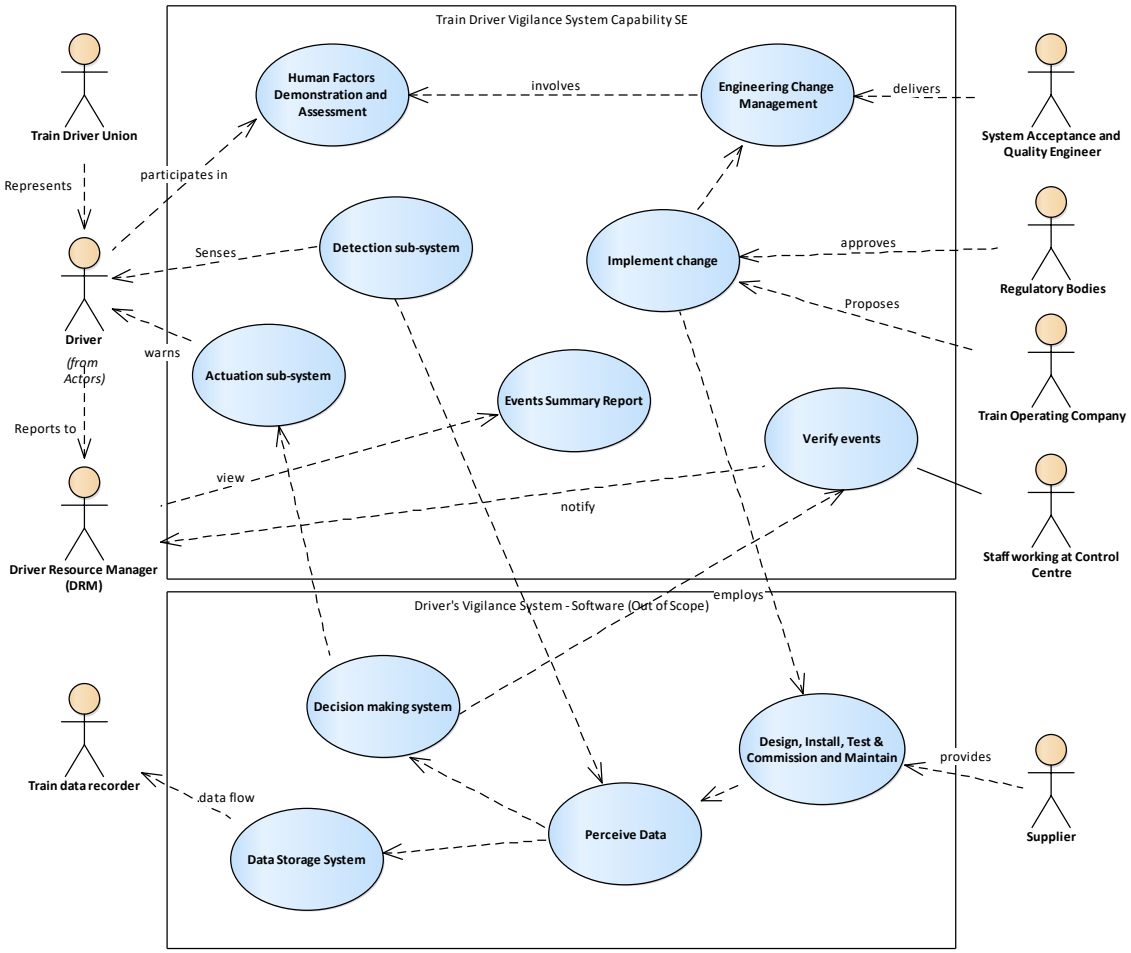

Figure 5 ~ Use Cases Diagram of Driver's Vigilance System

5.3 Measures of Effectiveness

5.3.1 Usability MoEs

MoEs set out the high-level requirements of the system at the operational concept level. The primary users', i.e. the drivers', concerns are collated in the following list of usability MoEs (Table 2). Please note that these high-level requirements are some MoEs suggested by this analysis; further analysis will be required to develop a complete set of the requirements.

The source of reference whence the requirements are derived are recorded in the Traceability column. Some requirements have been taken from, for example, light-rail or automotive applications, and therefore need to be adjusted to the mainline operational environment when being adapted to the system under discussion.

Table 2 ~ High-level Requirements (MoEs) on Usability

No.	Usability Measure of Effectiveness (MoE)	Traceability
Primary User MoEs		
1. U1	Must accommodate the driver to carry out all operation tasks and associated movements. These devices shall provide constant vigilance without any action from the driver. They can be video-based, vehicle based or physiological signals sensors.	Based on Sub-section 3.5 of this paper, and Usability Criteria set out in RSSB research (Whitlock 2002)
2. U2	Interface between the sensing device and the driver must be physically remote to each other such that there is no adverse impact on driver's health and safety due to the device.	Based on section 3.6 of this paper, and Usability Criteria set out in RSSB research (Whitlock 2002)
3. U3	Must accommodate religious head-dresses (e.g. Sikhs' turbans).	Usability Criteria set out in RSSB research (Whitlock 2002)
4. U4	The area of visibility of eye tracking device shall be able to cover a range of driver's height based on an eye level of between 740mm to 855mm above the seat level.	BS EN 14033-1:2007, Clause 14.6 Area of visibility
5. U4	The field of vision and depth of field must accommodate body sizes that range from 5% female to 95% male.	Requirements of camera (von Jan et al. 2010)
6. U5	Any other adjustable features necessary for driving the train under normal conditions shall be positioned: a) With the seat back in an upright position. b) With the seat facing forward and adjusted vertically and longitudinally to place a 50th percentile male in an appropriate driving position for a person of that size.	Railway group standard GM/RT/2100 Clause F.2.5 & F.2.6
7.	Infrared sensor shall emit light below the threshold for human eye safety level, i.e. near infrared (NIR) of wavelength 780 nm.	IR emitted shall be less than the human eye safety threshold (Tufuor 2017). To minimise interference from light sources beyond IR light; and to maintain uniform illumination (Ji and Yang 2002)
8. U6	The device shall accommodate sunglasses worn for driving duties that comply with the relevant requirements of BS EN ISO 12312-1:2013.	Eye and face protection - Sunglasses and related eyewear for general use BS EN ISO 12312-1:2013

No.	Usability Measure of Effectiveness (MoE)	Traceability
Primary User MoEs		
9.	The device shall accommodate corrective spectacles that are in line with Railway Group standard on Train Drivers — Suitability and Medical Fitness Requirements.	Rail Industry Standard RIS-3451-TOM: 2016
10. U7	Calibration shall be within 5 minutes.	Usability Criteria set out in RSSB research (Whitlock 2002)
11. U8	The device shall have a react and reset mechanism.	Based on BS EN 14033-1 Clause 14.11 for current system
12. U12	Need to accommodate different skin colours, and eye and face make-up.	Transport for London (TfL) reported make-up of female drivers have passed trial (Booth 2017)

5.3.2 Functional Requirements

Functional requirements are initially specified at high level, as the capability design is established in detail in later stages. Most importantly, the system shall be able to detect distraction and fatigue of drivers, be able to perceive images, alert on occurrence of events, and notify them accordingly when such events are identified.

Table 3 ~ Functional Requirements of Driver's Vigilance System

No.	Measure of Effectiveness (MoE) / High-level requirements	Traceability
13.	The device shall be able to detect fatigue. Signs include extended eye closure of more than 1.5 seconds.	Eye detection on Microsleep (Poudel et al. 2014)
14.	The device shall be able to detect visual distraction, i.e. eyes off the road of more than 1.8 seconds.	Real-Time Detection of Driver Distraction (Tango and Botta 2013)
15.	The system shall be notified if the driver is absent, that is, out of detection zone for more than 1 second.	Guardian device (Seeing Machines 2019)
16.	The system should be able to measure train speed and acceleration.	Guardian device (Seeing Machines 2019)
17.	The system shall provide a warning to the driver when a distraction event or a fatigue event is detected.	Guardian device (Seeing Machines 2019)
18.	Human machine interface shall provide a function for verifying fatigue events.	To distinguish false-positives manually as part of intervention plan

5.3.3 Non-Functional Requirements

Non-functional requirements (NFRs) are developed based on Performance, Scalability, Availability, Feasibility, Flexibility, Regulation, Operability, and Security, as suggested by the Enterprise Architecture Framework (Deighton 2014) developed from TOGAF, The Open Group Architecture Framework. Cost and commercial aspects are not considered in this study and thus Feasibility is not included in the development of these NFRs.

Table 4 ~ Non-functional Requirements of Driver Vigilance System

No.	Measure of Effectiveness (MoE) / High-level requirements	Traceability
Performance MoE		
19.	The device shall be tested under train driving tasks through: • Train simulator; and • Trial operation.	Scientific Criteria 1, RSSB research on driver vigilance devices (Whitlock 2002)
20.	Detection method shall be validated against recognised measurement methods. Eye blinking rate detection to be validated by the following defined as a fatigue event: • percentage of eye closure over time increases to over 30%, excluding the time spent on normal closure; and • eye closure speed increase to above 0.5s.	Drowsiness detection method (Dinges 1998); and eyelid movement monitoring (Ji and Yang 2002)
21.	Detection method shall be validated against recognised measurement methods. 'Eyes-off-road' to be validated by timing 1.8s defined as a distracted event.	Driver distraction definition (Tango and Botta 2013) is 1.8s for road vehicle. This is assumed to be the same for train driving.
22.	Events data shall be stored locally on the device in the driving cab for at least 24 hours for review.	New data overwrites old data after 24 hours (Seeing Machines 2019)
23.	Sensitivity – The device shall be able to quantify the times it misses an event. • Sensitivity = 100% × hits / (hits + misses) (False-negative) Data regarding the percentage of missed events shall be higher than 90%.	Prototype delivers satisfactory sensitivity of above 90% calculated on a minute basis (von Jan et al. 2010).

No.	Measure of Effectiveness (MoE) / High-level requirements	Traceability
24.	Specificity – The device shall be able to calculate the number of times that it presents a false alarm. • Specificity = 100% × correct rejection / (correct rejection + false alarms) (False-positive) Data regarding the percentage of false alarms shall be higher than 90%.	Prototype delivers satisfactory specificity of above 90% calculated on a minute basis (von Jan et al. 2010).
25.	Reaction time shall be short enough, i.e. less than 2.5s, to provide real-time feedback to alarm the driver.	Train at 125mph travels 139m in 2.5 seconds. Reaction time requirement for acknowledgement of TPWS horn is 2.5 seconds (El Rashidy et al. 2016)
Scalability MoEs		
26.	The ability of the backend system to handle increasing or decreasing volumes of services and data of up to 500 driving cabs or trains.	14,025 current operating trains in UK (RSSSG 2018) divided by 28 TOCs which equals an average of 500 trains per TOC
Availability MoE		
27.	Reliability study considering the failure mode of components and the failure rate of the function shall be provided. Mean Time between Failure (MTBF) of Driver Vigilance System shall be at least 100,000 hours.	Reliability to be re-assessed based on changes in overall system architecture and new components. Failure of monitoring system is service affecting. Based on (EU) No 1302/2014 requirement for reliability study and EKE-Electronics Ltd (2017) MTBF figure, see Appendix A
28.	Mean Time to Repair (MTTR) of Driver Vigilance System shall be within 1 hour.	Active repair time excluding logistics and testing
29.	The control centre shall be notified when signal from the train cab is lost for more than 30s seconds.	Half the time of 60s detection for lack of driver's activity as defined in BS EN14033-1 Clause 14.11

No.	Measure of Effectiveness (MoE) / High-level requirements	Traceability
Flexibility MoE		
30.	The backend system shall possess an interface to allow change and extension of functionality including: • Set up equipment for a new train; and • Change intervention plan.	Staff are able to sustain and manage the system on a daily basis
Regulation MoEs		
31.	Meet UK standards in accordance with EN50126 on RAMS, EN50716 on Software and EN50129 on hardware, or equivalent.	European Standards EN50126, EN50716 and EN50129
32.	Shock/vibration compliance according to EN61373 Rolling stock equipment — Shock and vibration tests.	European Standard EN 61373
33.	The Driver's safety device (DSD) that operates by stopping trains if driver become incapacitated must meet the appropriate SIL.	Manufacturer datasheet as an example where the existing DSD meets SIL 3: Safe-plus module instruction handbook (DEUTA-WERKE 2019)
34.	The design should allow deactivation of the DSD function in case of equipment failure, e.g. when the DSD does not reset.	Railway Group Standard on monitoring device (RSSB 2021)
35.	Meet UK Safety regulation Common Safety Method for Risk Evaluation and Assessment (CSM-RA).	Implementing Regulation (EU) No 402/2013
36.	Meet UK EMC requirement: EN 50121-3-2:2016 for Rolling stock apparatus. • The device should be robust against magnetic (or otherwise) interference from traction motors, overhead power cables and the electrified third rail; and • If the device uses telemetry for transferring data, interference of data transfer between current and future train-to-track wireless communications shall be within standard requirements.	EN Standard 50121-3-2:2016, and UK Railway Context Criteria (Whitlock 2002)
37.	The device shall be compliant with GM/RT/2000 Engineering Acceptance of Rail Vehicles.	Railway Group Standard GM/RT/2000

No.	Measure of Effectiveness (MoE) / High-level requirements	Traceability
Operability MoEs		
38.	The device shall have the facility to connection to other train interface components, if required, e.g. • Traction Brake Controller (TBC); • AWS; and • OTMR.	Edited based on UK Railway Context Criteria (Whitlock 2002)
39.	To record and store verified events for up to 12 months for retrieval.	Footage of the event are available to view for up to 1 year (Seeing Machines 2019)
40.	Eye tracking devices shall be compatible with behaviours adopted by drivers in trains with in-cab signalling. Eye-tracking devices should be able to accommodate 'head-down' activities as well as 'head-up' activities.	UK Railway Context Criteria (Whitlock 2002)
41.	The device shall be able to operate in both air-conditioned and non-air conditioned environments, i.e. an airflow of at least 30 m^3 per hour per person.	EN14033-1: 2007 Clause 14.4 Heating, cooling and ventilation
42.	The device must: (a) Tolerate a temperature range between -20℃ to +50℃ inside the vehicle when non-operational; and (b) Tolerate a range of 18℃ to 23℃ in the cab when operational.	EN50125-1:2014; and Thermal load of the new system shall be managed such that it has sufficient hardware capacity and a cooling mechanism to prevent overheating. EN14033-1: 2007 Clause 14.4 Heating, cooling and ventilation
43.	The device shall be able to operate under typical light condition of a driving cab which is between 30lx to 60lx of internal lighting.	EN14033-1: 2007 Clause 14.5 Internal Lighting
44.	The device manufacturer shall provide an operating manual that details 1. System description; 2. Operational instructions; and 3. Maintenance procedure.	Engineering Experience on Rolling Stock project
45.	Drivers have sufficient training to be familiar with the change in system in accordance with train driver mental workload guidance note.	Driver mental workload (RSSB 2005b)

No.	Measure of Effectiveness (MoE) / High-level requirements	Traceability
Cyber Security MoEs		
46.	The system shall protect information confidentiality and integrity. • Normal driving video footage shall be kept in memory for processing purpose only, i.e. no data storage or retention; and • Driver identity including name and staff ID shall only be available upon request from DRM.	Ethical concerns referenced from Guardian Device (Seeing Machines 2019) Confidentiality and Integrity Algorithms for UMTS and LTE (ETSI 2018)
47.	Wireless communication must be encrypted using secured network, such as 4G LTE. Additionally, the system shall be designed for data to be encrypted before transmission.	InfoSec

5.4 Operational Concept

The existing system is proposed to be re-designed with enhanced capability to monitor drivers' fatigue and distraction events and give alert accordingly, as illustrated in Figure 6.

Symptoms of distraction and fatigue events can both be captured using image processing technique by analysis driver's eye and head movements. The benefits of camera sensors outweigh all other kinds of technologies in that it is easier to use, less invasive, and more accurate. Video-based sensors can thus be used for capturing images of the driver.

Figure 6 shows that visual distraction is detected when the driver looks away for more than 1.8s; this actuates an audio alarm. This is within the reaction time requirement for driver's performance of 2.5s on acknowledging a train protection equipment button. To reset the system, the driver looks back at the centre, and the ringing sound is turned off.

A fatigue event is defined as the drivers' percentage of eye closure over time increasing to above 30%, excluding the time spent on normal closure (Dinges 1998), and eye closure speed increases to more than 0.5s (Ji and Yang, 2002). This actuates the vibration alarm at the driver's seat, as well as the audio alarm. A vibration alarm is used for fatigued drivers because people who experiences microsleep do not react to light or sound (Created Mode 2020). Once the driver reopens his or her eyes, the system is reset, and the alarm goes off.

The delay time of reacting to a fatigue event and warning being reset is thus reduced from 5 seconds to 2 seconds. The improved system allows detection of drivers who softly close eyes who can still react to sound. There is thus constant vigilance that actuates in real-time without interfering with driver's operations. The 2 seconds estimate is based on 1.8s for visual distraction plus digital transmission time.

The delay time for drivers who merely softly close their eyes and can be returned to being attentive by audio beeping is reduced from 30s (*5s initiation + 20s detection + 5s after alarm flashes*) to 2s. The reset time is typically 5 to 7 seconds and the shortest time is used for worst case analysis.

For drivers who are asleep and cannot be woken up by audio beeping, but can be woken up by seat vibration, the delay time is reduced from 65s (*5s initiation + 60s delay time*) to 2s.

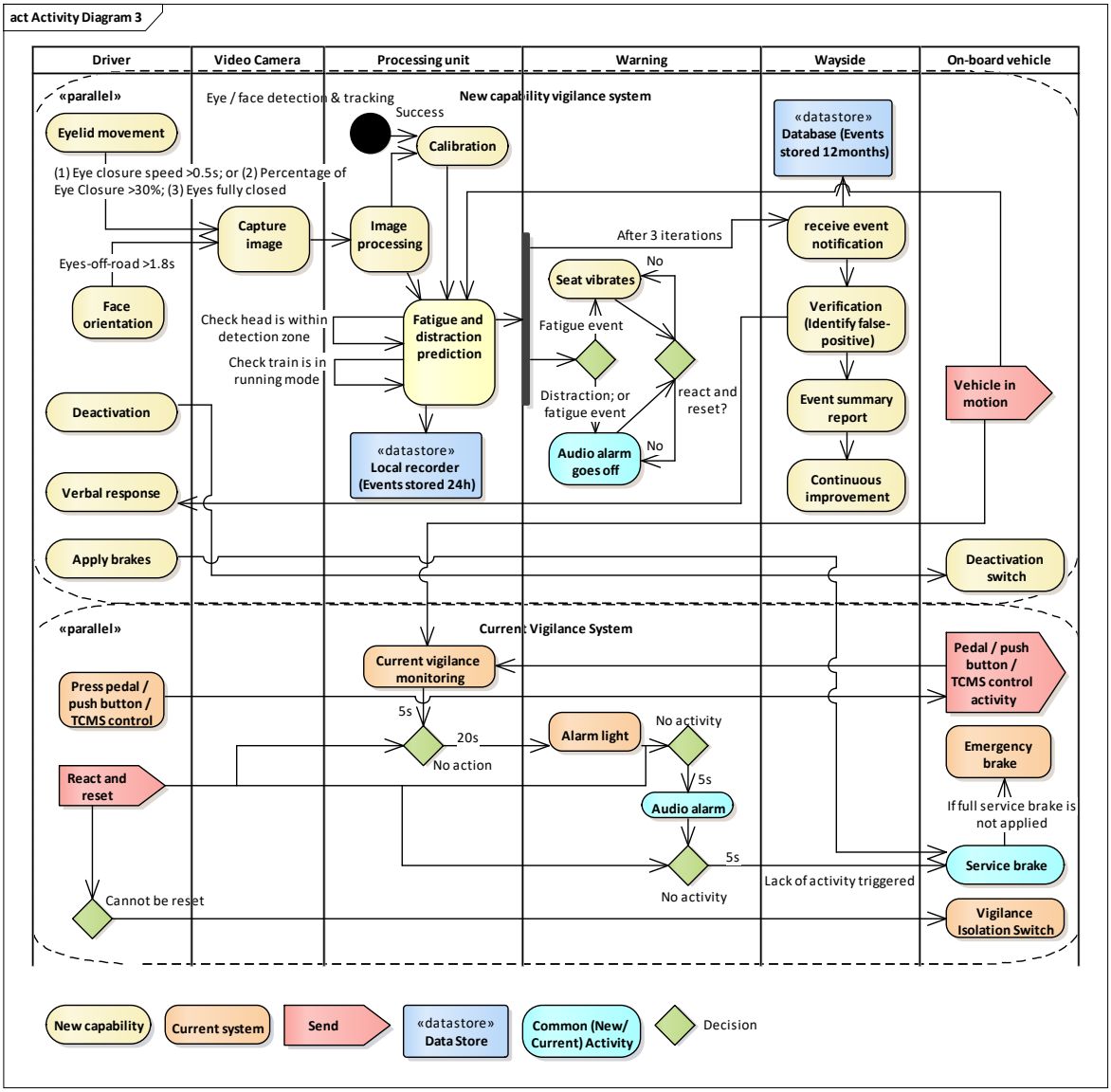

Figure 6 ~ Operational Concept of the New Capability

The proposed new capability of the vigilance system serves as a redundancy system to the existing DVD to monitor whether the driver has lost consciousness. In Figure 6, the new capability to detect driver's alertness by signs of distraction and fatigue is denoted in yellow. The existing vigilance system is denoted in blue, which forms the baseline of the discussion.

If consecutive fatigue events are detected, wireless messages are transmitted to the control centre to notify the staff of a fatigue event. It is determined mutually between the driver and the control centre staff to stop the train, or to be decided by the staff unilaterally to apply brakes if no reaction is received from the driver.

For drivers who are extremely tired, and would like to take a break, the response time is 129s [*3 iterations × (2s event triggered × 1s actuator reacts) + 60s verification + 60s contact DRM*]. The number of 3 iterations are based on the Guardian Device in which two fatigue events are allowed before notifying the supervisor, however, this can be negotiated to suit each TOC's management policy if required.

The reaction time for drivers who become incapacitated or dead remain the same at 65s because only the current DSD is connected to the emergency brakes, and can automatically stop the train without any delay in verifying or communicating with the DRM.

The drivers are able to deactivate and isolate the equipment when it fails to reset. The driver can continue train operation until completion of train journey where the failed device would be repaired, or replaced at the depot.

Distraction and fatigue events are summarised in a regular event summary report for fatigue management as part of the operator's continual improvement programme.

5.5 Whole Capability Design

5.5.1 Physical Decomposition

The proposed system with improved functionality is able to track fatigue, unintentional and intentional distraction, such as playing video games on smartphones, reading newspaper, or microsleep events.

The physical decomposition of the new capability can be structured into three parts. They are the driver's cab subsystem, the vehicle on-board subsystem, and the wayside subsystem, see Figure 7.

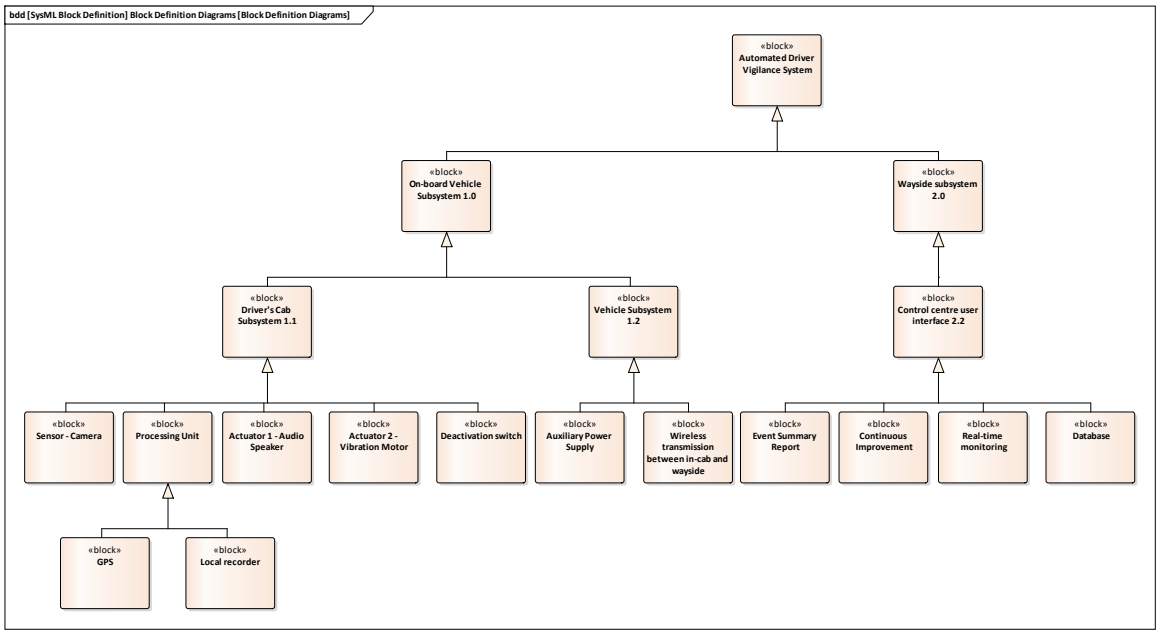

Figure 7 ~ Physical Decomposition of the New Capability

5.5.2 Driver's Cab

This subsystem consists of components installed in the driver's cab.

A sensing device is used to record images of the driver's head. Once initiated, it calibrates each individual's features when using the device for the first time. The sensing subsystem is made up of a sensing device, a controller, and a power module (Toto 2012), all installed inside the cab.

The system uses an infrared (IR) illuminator which serve various purposes. It eliminates the impact of the variable light conditions during driving; day or night or inside tunnels. IR detects bright pupil effects even with sunglasses on (Ji and Yang 2002).

Near-infrared (NIR) is barely visible to the driver, minimising interference to their daily operation. As long as the wavelength is set at 780nm or more, the beam is invisible to human eye, does not cause damage to human when exposed for long periods of time, and is considered safe (Tufuor 2017).

A driver-facing video camera with an NIR illuminator can thus be fitted in the driving cab to serve these functions while satisfying MoEs 44, 8 and 7 respectively.

The software part of the Vigilance System carries out biometric analysis. Data collected are sent to the processing unit for extraction, transformation, and information classification. The processing unit has a GPS[8] sensor, which measures train speed and acceleration. Normal video footage of driver shall be overwritten after 24 hours inside the driving cab storage unit, as defined in MoE 22.

The audio speaker is used for giving hearable alarms to the driver. Any modifications to the audio speaker should aim at minimal disturbance to existing systems, such that impact to other safety-critical systems can be avoided. The vibration alarm is provided by a dedicated motor located inside the driver's seat.

To allow isolation of the equipment if it fails, a breaker switch is installed in the driving cab for the driver to deactivate the system when required.

5.5.3 On-board Vehicle

The equipment on-board the vehicle needs a low voltage power supply from the vehicle, which comes from the Auxiliary Power Supply, part of the vehicle's traction system.

Wireless transmission equipment is fitted to allow signals to be communicated between on-board vehicle systems and the wayside. To demonstrate that the vehicle system is live, a signal is sent to the wayside every 5s. The system receives a signal lost notification when the train passes through a tunnel, or an area with poor network, as required in MoE 29. A few seconds queuing delay is allowed for the signal to be detected once network is resumed.

The current system architecture of the DSD with connection to the Service and Emergency brake can be maintained with no modifications required for the new capability. This is to minimise safety impact on critical systems.

The device shall obtain inputs from a speedometer from the TCMS to determine whether the train is in running mode, and in motion.

To summarise, subsystems on-board train are comprised of the Auxiliary Power Supply, Processing Unit (with GPS sensor), Wireless communication, and connection with a speed indicator.

[8] Global Positioning System

5.5.4 Wayside

An interface is needed for the monitoring team to be notified about fatigue events. The design of the user interface shall allow staff to classify events into either false-positive or confirm as actual fatigue events.

Records of fatigue events are stored for up to 12 months for analysis purposes, as required in MoE 40. Frequency of events' occurrence can be compared between different work shift patterns to inform the fatigue management programme.

The user interface for control centre staff thus needs to possess three main functions: Real Time Monitoring, Event Summary Reporting, and a Continual Improvement Programme. The DRM can use analysis results for fatigue management and analysis in the longer term.

5.6 Subsystem Requirements

Physical subsystems are assessed through a set of Measures of Performance (MoPs) which list out the minimum threshold for subsystem requirements. Traceability is provided to justify how they relate to higher level performance requirements (i.e. the MoEs). Please note that these requirements are examples suggested by this analysis for demonstration purposes. Further analysis is required to develop a complete set of the requirements.

Table 5 ~ Subsystem Measures of Performance (MoPs)

Domain	Requirement	Threshold	Justification
Video sensor (camera)	Camera resolution is higher than…	720 x 576 pixels with refresh rate of 12 frames per second.	Images are clear enough for evaluation. Rail Industry Standard RIS-2703-RST Driver Controlled Operation (DCO) On-Train Camera/Monitors (OTCM); and Driver Monitoring Systems (DMS) and Occupant Monitoring Systems (NXP 2020)
Video sensor (camera)	Camera field of view (FOV)	60 degree	Cover head position for a range of user's height. Driver Monitoring Systems (DMS) and Occupant Monitoring Systems (NXP 2020); and Rail Industry Standard RIS-2703-RST Driver Controlled Operation (DCO) On-Train Camera/Monitors (OTCM)
Video sensor (camera)	Calibration time shall be less than	5 minutes	To be completed alongside Start of Mission (SoM) of the vehicle. RSSB usability criteria (Whitlock 2002)
Video sensor (camera)	Monitor condition every…	2 seconds	Heart rate detection by Steer wearable device (Created Mode 2020)
Video sensor (camera)	Noise emitted during operation is lower than	1dBa	Video fatigue and distraction monitoring (Progress Rail 2015)

Domain	Requirement	Threshold	Justification
Video sensor (camera)	Narrow bandpass NIR filter shall be attached to the front of the lens to attenuate light below…	780 nm	To minimise interference from light sources beyond IR light; and to maintain uniform illumination (Ji and Yang 2002). IR with wavelength 780nm is invisible to human eye and is below the threshold for human eye safety (Tufuor 2017).
Vibration motor	Vibration rate must be higher than	15Hz	Low frequency vibration corelates to poor driving performance and drowsiness (Azizan and Ittianuwat 2016).
Audio alarm	Alarm sound shall be at least … higher than noise level in the driving cab	6dBa	Acoustic warning devices requirements in (EU) No 1302/2014
Monitoring System	Reaction time is less than	1s	Short enough to provide real-time feedback to alarm the driver.
Processing unit	Support transmission speed of at least	5Mbps	720p streaming of high-definition video require a bandwidth of at least 5Mbps (Zen 2019)
Real-time intervention	Response time for verification is less than…	1 minute	Short enough to verify fatigue event and obtain mutual agreement on intervention action. Video fatigue and distraction monitoring (Progress Rail 2015)
Human-machine interface	Event summary report is available every …	1 week	Align with Video fatigue and distraction monitoring (Progress Rail 2015)

MoPs of hardware and frontend subsystems that directly interface with the users are set out with justification. Image processing software and prediction algorithms used to identify parameters and to calculate drivers' fitness level are not discussed in detailed. It is proposed that future work be done by a psychology, or physiology, expert with software engineering background.

5.7 Interface and Safety Impact

Technical interfaces of the subsystems are traded out to understand the relationship between contributing elements.

It is not recommended that the new capability be integrated with the existing braking system and DSD at this stage. The consequences of failure of these safety critical systems could be catastrophic. Minimising modifications on the current systems allows the DSD to continue to operate if the new DVD fails. An ALARP argument will be required at option selection stage on whether the risk is acceptably controlled for the improved capability of the DSD to not interface with the existing systems.

Any changes in the vehicle needs to comply with the CSM-RA. Although novel, the likelihood of the change being significant is lower if implemented without integrating, but if it were integrated with the current system, it would interface with safety critical systems (the current braking system and DSD). A Quantitative Risk Assessment needs to be carried out to meet SIL allocation targets.

One of the foreseeable interface hazards is interference with the existing communication system to the signaller. Wireless transmission is used for sending event notification to wayside and checking of signals. It is essential to ensure that it does not interfere with the existing wireless transmission of the train control management system. This can be validated by conducting EMC tests in compliance with the BS EN 50121 series of standards on railway electromagnetic compatibility.

For systems that involve operational technology, security-informed safety needs to be addressed. Security hazards include leakage of confidential personal information through cyber incidents, such as hacking. This may have a detrimental effect on the company's reputation. The system architecture has to be robustly built according to the IEC 62443 series of standards regarding security for industrial automation and control systems. Both 'data at rest' and 'data in motion' need to be encrypted as part of maintaining cyber security and recovery throughout the whole life cycle.

To ensure distraction and confusion to drivers are minimised, the new DVD has to be designed ergonomically. Causes of distraction might be unfamiliar alarms and warnings (both audio and visible); discomfort from vibrating seats; and perception of the infrared beam. These can potentially lead to a driver being unaware of a critical alarm and taking inappropriate action. Sufficient driver training, human factors assessment, and trial exercises shall be conducted to collect user feedback and to ensure warnings are given clearly. As explained in MoE 7, the infrared beam wavelength has to be set in the NIR range above the safety threshold at 780nm.

To satisfy MoE 1, it is critical to ensure operation of the new system does not interfere with driving operations. A specific concern is the frequency and timing of when the old and new audio alarms would go off; it is shown in Figure 8 that for the new system, the audio alarm occurs within the first 15s, whereas for the current system, the buzzer sound becomes active after 30s of driver inactivity. Therefore, no overlap of audio alarms is foreseen that could cause driver confusion.

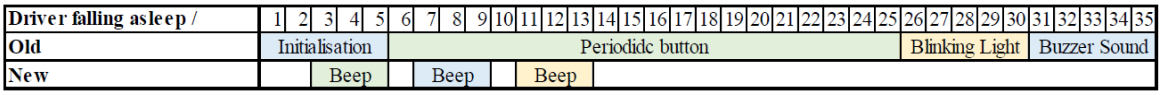

Figure 8 ~ Comparison of Frequency of Audio Alarm between Current and New Vigilance System

The users need to be assured that the system design meets their needs, and that this system is not intended for surveillance purposes by their managers. An acknowledgement button shall be available for the driver to reset the warning. Implementation should be voluntary, such that a deactivation switch shall be available for drivers to isolate the system, although this switch is intended for isolation of the device when it fails. The drivers should also be able to cancel the notification to the control centre. A driver's identity is only available upon request and is not shown on human-machine interface summary reports. Distraction events are not used for real-time intervention. Video footage stored locally is to be overwritten after 24 hours.

6 Discussion of Results

6.1 Results

The current DSD is to remain as it is still important for detecting drivers who become incapacitated or dead. It would also serve as a backstop if the new vigilance system were to fail. The new vigilance system is most effective for drivers asleep who cannot be woken up by audio alarms. Driver who experience microsleep can be woken by vibration, but not by the beeping sound. In such cases, the reaction time drop from 65s to 2s resulting in a 97% decrease. Table 6 below shows the extent of improved capability in the column 'Percentage of Reduction'.

Table 6 ~ Comparison Between Current and New Vigilance Systems on the Delay Time Between the Onset of Inattention and the Warning Being Reset

Level of Driver Readiness	**Reaction Time**		
	Current Vigilance System	**New Vigilance System**	**Percentage of Reduction**
Driver incapacitated/dead	65s	65s	No Change
Driver asleep but *cannot* be woken up by beeping sound	65s	2s	97%
Driver asleep but *can* be woken up by beeping sound	30s	2s	93%
Microsleep or driver who softly closes eyes	N/A	2s	N/A
Distracted driver (both intentional and unintentional)	N/A	2-3s	N/A
Drivers who experience Highway Hypnosis	N/A	N/A	N/A

New functionalities include detection of microsleep and distraction. Neither system is yet capable of detecting highway hypnosis.

6.2 Application

Vigilance monitoring is widely adopted in freight vehicles in USA and Australia (Seeing Machines 2019), and has now been transformed to be used for light rail drivers (RAIB 2019). The proposed system would provide enhancement of capability for current DSD and associated DVD for all TOCs and manufacturers in UK mainline railway who conform with EN standards. RSSB has recently published research (RSSB 2021) on understanding the functional requirements for train driver alertness and attention monitoring devices. The study went on to discuss the devices that have been chosen to be used in the railway industry, usually involving only one or two specific suppliers. This approach lacks a comprehensive analysis, considering all possible solutions, and has not identified measurable requirements that can be verified. However, it reflects the criticality and urgent need to improve the capability of existing vigilance device for national use.

If implemented successfully, this system is foreseen to reduce 20.7% of overall fatal train accidents in UK. This figure is calculated based on multiplying 44% of accidents caused

by SPAD or over-speeding (Evans 2017), and 49% of these accidents caused by the driver's alertness and distraction (RSSB 2021) with an additional 4% involving the driver's incapacitated or asleep, where:

- *44% × (49% - 5% + 3%) = 20.68%*;
- assuming 1% are from incapacitated drivers and 3% are from asleep drivers, out of the 4% of drivers being incapacitated or asleep; and
- assuming the software algorithm is 100% effective; and
- assuming the system is used on every journey; and
- excluding accidents caused by drivers experiencing highway hypnosis, which is assumed to be 5%. (Further analysis is required to understand the breakdown of accident causes to obtain a more accurate figure.)

Apart from passenger vehicles, this new capability can be explored for UK rail freight usage as well.

6.3 Limitations

There are caveats on the new capability regarding the validity of road data that need verifying for the rail environment. Most research and data found for fatigue, distraction and inertia detection are based on road driving. Working nature and shifts are however different from railway environment; controls and road regulations are more complex. A separate investigation is needed for the railway to redefine these values specifically.

Related work has been done on determination of fatigue and alertness based on gaze determination by pupil and glint tracking, as well as face orientation monitoring using properties of pupils (Haq and Hasan 2016). The fatigue determination threshold varies, depending on skin colour and colour of the iris (Haq and Hasan 2016). Due to the complexity of this algorithm, these factors are not included in the consideration of this paper.

Prediction algorithms (von Jan et al. 2010), and their on-board implementation falls under the software evaluation process, which has not been studied in detail in this paper. A major concern around Machine Learning technology is its inherent black box nature. Computer vision processing using non-traditional modelling also remains a controversial topic, and requires further development, particularly on how integration with safety-critical systems can be done in railway industry.

6.4 Challenges

Implementing changes to the driver's cab is challenging, because of all the stakeholders that need to be satisfied. The train drivers' union called for strikes relating to Croydon trams being fitted with the infrared sleep sensor. Health concerns such as headaches and blurred vision were claimed to be caused by the new device. Although TfL claimed that the system is safe, with many years' experience of it use in the road haulage industry (BBC 2017), the implementation of change is still yet to be achieved with full operation.

The video camera sensor is associated with facial recognition technology, which is a controversial topic. One wonders if this is a breach of human rights. The usage of the new vigilance system thus shall be rolled out on a voluntary basis. The driver can deactivate the function when wishing to be withdrawn from the programme.

There needs to be a clear agreement that employers shall not use the technology for performance management, or for disciplinary action. The technology should be discussed at length with drivers and their trade union representatives before introduction.

The new capability needs to assure the users of confidentiality of their personal data, as well as the benefits that can be brought to both the individual, the passenger, and the railway as a whole. If employers do the right thing, I believe the trade unions can help them convince the employees; to explain that the introduction of this technology is to make driving safer whilst not compromising their privacy.

6.5 Implementation Opportunities

The first phase of implementation should involve a defect liability period, focusing on the detection and alarm process of distraction and fatigue events within the drivers' cab. Data collected from the events should be analysed, and changes made according to feedback from the primary user as well as the output of the analysis.

Although detection of facial expressions has been under development for years, the application is novel to rail industry. Therefore, a period for identifying defects for specific application to train driving cab should be allowed before full operation.

Once any defects are resolved, the project can move to a second phase which implements the connection with the wayside system.

It is estimated that this system can be rolled out in one year from start of development. This include 2 months of planning, 4 months of design and build, 2 months of installation ,and 4 months of testing and commissioning. It took eight months for the Croydon tram to implement the Guardian device after the fatal accident investigation (Booth 2017), but the regulations and stakeholders are more complex for mainline rail, therefore, it is considered that one year is a reasonable timeframe.

The immediate benefit of the vigilance system is to reduce fatigue and distraction occurrences, and thus improve safety, if implemented successfully. In the longer term, a continual improvement programme can be rolled out to strategically plan work shift rosters and manage fatigue more effectively.

With the popularity of autonomous driving alongside upgrade of Grade of Automation, one may query the value of developing such a device for drivers. Autonomous driving for metros has been under rapid development in recent years.

However, there is actually a low chance of drivers becoming redundant in the near future on the mainline railway. According to Stagecoach Group plc, at the time of the research there were tens of thousands of railway drivers employed to run the UK train network. Compared to metro or light rail, the mainline railway has a slower pace of introducing automation due to its complexity and longer journeys. Even if automation is granted, drivers are still needed to be maintained with continuous driver training for dealing with emergency situations.

7 Conclusions

7.1 Summary

A capability study has been carried out on a drivers' vigilance system to understand the high-level requirements, should UK mainline railways plan to adopt this monitoring system. Throughout the systems engineering process, the functional and non-functional requirements were captured and verified once the whole capability design had been developed. The new vigilance system was found to improve reaction time of fatigue and distraction events detection by more than 90%, and under general assumptions, this would reduce fatal train accidents in UK by up to 20.7% once implemented.

7.2 Findings

It is believed that there is an urgent need for transforming current DSD and DVD system to address its inherent shortcomings. As discussed in Sub-section 2.2, the existing system has a long delay time, can be easily isolated and cannot detect microsleep. The improved capability of the vigilance system is able of detecting microsleep that takes place between 0.5s to 30s, which the traditional device is not able to spot. A significant improvement is shown compared to the current system with delay time of 60s. It is also able to alert the driver of distraction events if they looked away for more than 1.8s. This additional functionality enables the passengers to be "in safe hands", away from a driver's intentional distraction. The system is not intended for detecting a driver becoming incapacitated or dead, which is currently covered in the existing DSD.

The system needs to be designed in a way to deal with a wide range of scenarios. Eye detection technology needs to be sophisticated enough to produce accurate results for populations with a variety of iris colour and skin colour. Other situations include operational aspects when the system is being intentionally tricked, or if the camera is obscured. There should be design consideration and rules in place, in combination with driver training to avoid the occurrence of these behaviours.

Compared to previous research, one of the innovations of my project is the method of adopting the Capability System Engineering approach. This process allows a system-wide overview of the vigilance system tailor-made for UK mainline railway. Unlike other papers that merely compare available gadgets on the market from a supplier's or purchaser's point of view, this system analysis provides a holistic understanding of industrial needs, guided by stakeholders' wishes and requirements.

7.3 Recommendations

Future work involving research and trials on the mainline railway, of drivers fatigue and distraction events in correlation to their physiological status will be helpful to determine detection thresholds, because the available data are mainly from road haulage vehicles, in which the driving pattern and working nature are significantly different.

Further work on this topic can be done extending the scope to cover driver operations in abnormal scenarios, for example in an emergency situation. Software development needs to understand the associated drivers' tasks during these circumstances.

Separate assessment is recommended for future integration with current vigilance device, which involving safety-critical systems. Interfaces with braking control, DSD control, and

the train management system require consideration and a formal change management process.

Fatigue determination using computer vision algorithms has not been discussed in detail in this paper. Future work is recommended to research a comprehensive method for fatigue and distraction prediction, and to modify the software model accordingly. This would best be done by a physiology or psychology expert, ideally with image processing and evaluation experiences. This is an iterative process that relies on technical disciplinary knowledge, and behavioural and medical sciences.

Starting off with a simple version, then eventually migrating to a vigilance system that caters for all scenarios, would be the ideal approach. Over time, the specifications of the product will be refined to determinate the nature of the final vigilance system. The best way would be for RSSB, RDG or DfT to put in place a set of high-level requirements as a standard for the improved capability of the vigilance system, and to fund development of a prototype demonstrator, or even a full system. This will ensure a consistent implementation across the industry.

A Polish physicist[9] once said, "*Nothing in life is to be feared, it is only to be understood*". The intent of the system analysis conducted was to facilitate the decision making of railway undertakings, by identifying the benefits and trade-offs of implementing a driving monitoring system. Only by exploring the requirements that need to be achieved, and addressing concerns associated with all stakeholders can the railway industry continue to strive to be a better and safer transportation mode.

Acknowledgments

Thank you to the organisers of the Safety Critical Systems Symposium, SSS'24, held in February 2024 by the Safety-Critical Systems Club in Bristol, UK, at which a presentation based on this paper was given.

Thank you also to Dr Marcelo Blumenfeld of the University of Birmingham, UK, who was supervisor for the dissertation upon which this paper is based.

Images are the work of the author, unless otherwise acknowledged by reference.

References

Azizan A. and Ittianuwat R. (2016). *Effect Of Vibration On Occupant Driving Performances: Measured By Simulated Driving*. International Journal of Scientific & Technology Research, Volume 5, Issue 01, January 2016. https://www.ijstr.org/final-print/jan2016/Effect-Of-Vibration-On-Occupant-Driving-Performances-Measured-By-Simulated-Driving.pdf. Accessed 12th July 2024.

BBC. (2017). *Croydon Tram drivers to strike over 'sleep detectors'*. BBC News, 1 November 2017. https://www.bbc.co.uk/news/uk-england-london-41830688. Accessed 12th July 2024.

[9] Maria Skłodowska-Curie (1867 – 1934), also known as Marie Curie.

Booth S. (2017). *We take a look at the controversial new safety device installed on Croydon trams.* Croydon Advertiser, 8th November 2017. https://www.croydonadvertiser.co.uk/news/croydon-news/take-look-controversial-new-safety-742796. Accessed 12th July 2024.

BS EN 13452-1. (2003) *Railway applications. Braking. Mass transit brake systems. Part 1: Performance requirements*. British Standard Institution.

BS EN 14033-1. (2017). *Railway applications. Track. Railbound construction and maintenance machines. Part 1: Technical requirements for running.* British Standards Institution.

BS EN 15734-2. (2010). *Railway applications. Braking systems of high speed trains. Part 2: Test methods*. British Standards Institution.

Commissaris R. (2019). *Unpublished Interview with Randall Commissaris, Ph.D., Professor of Wayne State University* (2 January 2019).

Created Mode. (2020). STEER: Wearable Device That Will Not Let You Fall Asleep. Kickstarter, 28 November 2020.

D'Agostino A. (2016). *Big Data in Railways — Common Occurrence Reporting Programme*. Technical Document ERA-PRG-004-TD-003 V 1.0. European Union Agency for Railways.

Deayea. (2019). *China High Speed Rail Brain Wave Sensor*. Deayea Technology Ltd., Shanghai.

de Castella T. (2015). *InterCity 125 v Hitachi: What are the UK's new trains like?* BBC News, 12 March 2015. https://www.bbc.co.uk/news/magazine-31831603. Accessed 12th July 2024.

Deighton D. (2014). *Enterprise Architecture Framework*. University of Birmingham.

DEUTA-WERKE. (2019). *Safe+ Module Instruction Handbook*. DEUTA-WERKE GmbH

Dinges D. F., Mallis M. M., Maislin G., and Powell J. W. (1998). *Evaluation of Techniques for Ocular Measurement as an Index of Fatigue and the Basis for Alertness Management*. U.S. Department of Transportation, National Highway Traffic Safety Administration Report DOT HS 808 762.

Dorrian J. (2008). *The driver vigilance telemetric control system (DVTCS): Investigating sensitivity to experimentally induced sleep loss and fatigue*. University of South Australia, Adelaide.

Doudou M., Bouabdallah A., and Charfaoui V. (2018). *A Light on Physiological Sensors for Efficient Driver Drowsiness Detection System*. Sensors & Transducers Journal, International Frequency Sensor Association (IFSA), 2018, 224 (8), pp.39-50.

EKE-Electronics Ltd. (2017). *Vigilance Control System (VCS)*. EKE-Electronics Ltd. Espoo. https://www.eke-electronics.com/wp-content/uploads/2022/08/EKE-Trainnet-Brochure-full_20220826.pdf (page 34). Accessed 12th July 2024.

El Rashidy R. A, and Van Gulijk C. (2016). *Driver Competence Performance Indicators Using OTMR*. Institute of Railway Research, University of Huddersfield. https://eprints.hud.ac.uk/id/eprint/28600/1/3417-7130-2-SM.pdf. Accessed 12th July 2024.

(EU) No 402/2013. (2013). *Commission Implementing Regulation (EU) No 402/2013 of 30 April 2013 on the common safety method for risk evaluation and assessment.*

(EU) No 1302/2014. (2014). *Commission Regulation (EU) No 1302/2014 of 18 November 2014 concerning a technical specification for interoperability relating to the 'rolling stock — locomotives and passenger rolling stock' subsystem of the rail system in the European Union.*

ETSI. (2018). *Universal Mobile Telecommunications System (UMTS); LTE; 3G Security; Specification of the 3GPP confidentiality and integrity algorithms; Document 1: f8 and f9 specification*. European Telecommunications Standards Institute ETSI TS 135 201

Evans A. W. (2017). *Fatal Train Accidents on Europe's Railways: 1980-2016.* Imperial College London.

GMRT2185. (2001). *Train Safety Systems*. Railway Group Standard GMRT2185, Iss. 2, 2001. Rail Safety and Standards Board.

Graves T. (2009). *Enterprise Architecture: A Pocket Guide*. IT Governance Publishing, Ely.

Griffith C. (2017). *Artificial Intelligence to Predict Accident Risk of Bus Drivers*. The Australian, 20th March 2017.

Guo M., Hu L. and Ye L. (2019). *Cognition and driving safety: How does the high-speed railway drivers' cognitive ability affect safety performance?* Transportation Research Part F: Traffic Psychology and Behaviour. 65. 10-22.

Haq Z. A. and Hasan Z. (2016). Eye-Blink rate detection for fatigue determination. IICIP 2016, 1st India International Conference on Information Processing, 1-5.

Ji Q. and Yang X. (2002). *Real-Time Eye, Gaze, and Face Pose Tracking for Monitoring Driver Vigilance*. Real-Time Imaging. 8. 357-377.

Kemp D., and Daw A. (2014). *INCOSE UK Capability Systems Engineering Guide*. UK Chapter of the International Council on Systems Engineering.

Kyriakidis M. (2013). *Developing a Human Performance Railway Operational Index to enhance safety of railway operations*. Imperial College London.

Langer C. (2016). *Analysing the root causes of Spain's high-speed derailment*. Rail Technology Magazine. https://www.railtechnologymagazine.com/Comment/analysing-the-root-causes-of-spains-high-speed-derailment. Accessed 12th July 2024.

LBC. (2018). *Train Safety Bosses Want To Put Spy Devices In Drivers' Cabs To Alert Them If They Doze*. Leading Britain's Conversation, 29 November 2018. https://www.lbc.co.uk/news/london/train-chiefs-putting-spy-devices-in-drivers-cabs. Accessed 12th July 2024.

Magazine Monitor. (2013). *Who, What, Why: What is 'highway hypnosis'?* BBC News, 4 December 2013. https://www.bbc.co.uk/news/magazine-25216601. Accessed 12th July 2024.

Masood S., Rai A., Aggarwal A., Doja M., and Ahmad M. (2018). *Detecting Distraction of drivers using Convolutional Neural Network*. Pattern Recognition Letters V.139 pp.79-85.

Meiring G. A. M. and Myburgh H. C. (2015). *A Review of Intelligent Driving Style Analysis Systems and Related Artificial Intelligence Algorithms*. Sensors 2015, 15(12), 30653-30682.

Microsoft. (2019). *Artificial Intelligence and road safety: A new eye on the highway — Facial recognition technology watches out for risks and monitors driver behavior*. Microsoft Asia News Center. https://news.microsoft.com/apac/features/artificial-intelligence-and-road-safety-a-new-eye-on-the-highway/. Accessed 12th July 2024.

Murray D. (2017). *Croydon tram bosses install infra-red beams to shine on drivers' faces to check they are awake — drivers to strike in protest*. Evening Standard, 1st November 2017. https://www.standard.co.uk/news/transport/croydon-tram-bosses-install-infrared-beams-to-check-drivers-are-awake-a3673341.html. Accessed 12th July 2024.

NEC. (2014). *NEC to implement solutions for SMRT Corporation to enhance bus service excellence — First telematics system in Singapore to monitor bus drivers' driving behavior*. https://www.nec.com/en/press/201408/global_20140805_02.html. Accessed 12th July 2024.

Neurocom. (2019). *Engine Driver Vigilance Telemetric Control System EDVTCS*. Neurocom, Moscow.

Nicol J, and Seglins D. (2014). *Freight train drivers report falling asleep on the job*. CBC News: October 7, 2014. CBC/Radio-Canada

NRIL. (2017). *Network Rail (Infrastructure) Ltd (NRIL) Health & Safety Management System*. Issue 4.0, May 2017. Network Rail (Infrastructure) Ltd.

NXP. (2020). Driver Monitoring Systems (DMS) and Occupant Monitoring Systems. NXP Semiconductors N.V. https://www.nxp.com/applications/solutions/automotive/adas-and-highly-automated-driving/driver-monitoring-systems-dms-and-occupant-monitoring-systems-:DRIVER-MONITORING-SYSTEMS. Accessed 12th July 2024.

ORR. (2010). *Guidance to Inspectors regarding the use of mobile phones / hand held electronic communication devices by train drivers*. RSD Internal Guidance RIG-2009-06, July 2010. Office of Rail and Road.

Poudel G. R., Innes C. R., Bones P. J., Watts R., and Jones R. D. (2014). *Losing the struggle to stay awake: divergent thalamic and cortical activity during microsleeps*. Hum Brain Mapp. 2014 Jan;35(1):257-69.

Progress Rail (2015). *Fatigue and distraction monitoring*. https://www.progressrail.com/

Reason J. T. (1990). *The Contribution of Latent Human Failures to the Breakdown of Complex Systems.* Philosophical Transactions of the Royal Society of London. Series B, Biological Sciences. 327 (1241): 475–84. London

RAIB. (2016). *Rail Accident Report: Overturning of a tram at Sandilands junction, Croydon*. Report 18/2017. Rail Accident Investigation Branch.

RAIB. (2019). *Rail Accident Investigation Branch Annual Report 2018.* Department for Transport. Derby, UK.

RSSSG. (2018). *Long Term Passenger Rolling Stock Strategy for the Rail Industry*. Sixth Edition, March 2018. Rolling Stock Strategy Steering Group.

RSSB. (2005a). *Human Factors Study of Fatigue and Shift Work*. Research Report T059, Rail Safety and Standards Board.

RSSB. (2005b). T*rain Driver Mental Workload: The Train Driver Workload Principles Guidance Note*. In: *MWL Workload*. Ergonomics, I.F.O. and C.D.E. Ltd., p15. Rail Safety and Standards Board.

RSSB. (2008). *Understanding Human Factors — a guide for the railway industry*. Rail Safety and Standards Board.

RSSB. (2014). *Project S184 — Driver alertness monitoring systems*. Rail Safety and Standards Board.

RSSB. (2018). *TW5: Preparation and movement of trains: Defective or isolated vehicles and on-train equipment*. Rule Book Modules: Train Working, GERT8000-TW5 Iss. 9. Rail Safety and Standards Board.

RSSB. (2019). *The Machine Learning validation challenge: what it is and how to overcome the 'black box' criticism*. Rail Safety and Standards Board Featured story 24 January 2019. https://www.rssb.co.uk/about-rssb/insights-and-news/blogs/machine-learning-validation-challenge-how-to-overcome-the-black-box-criticism. Accessed 12th July 2024.

RSSB. (2021). *Project T1193 — T1193 Understanding the Functional Requirements for Train Driver Attention and Alertness Monitoring Devices*. Rail Safety and Standards Board.

Seeing Machines. (2019). *Real-time Driver Fatigue, Distraction and Accident Prevention Technology*. Seeing Machines, Fyshwick ACT, Australia.

Seglins, J. N. a. D., 2014. Freight train drivers report falling asleep on the job, s.l.: CBC.

Stagecoach Group PLC, 2011. UK Rail Industry - FAQs, s.l.: s.n.

Stopsleep. (2020). *Anti Sleep Alarm for Drivers*. PPS Diagnostika, model number S200.

SMI. (2014). *SMI Eye Tracking Glasses 2 Wireless*. SensoMotoric Instruments GmbH., Germany. https://www.mindmetriks.com/uploads/4/4/6/0/44607631/final_smi_etg2w_naturalgaze.pdf. Accessed 26th June 2024.

Tango F., and Botta M. (2013). *Real-Time Detection System of Driver Distraction Using Machine Learning. Intelligent Transportation Systems*. IEEE Transactions on Intelligent Transportation Systems 14(2):894-905.

The Train Guard. (2014). *Train and Railway Safety Systems*. March 9, 2014. https://thetrainguard.wordpress.com/2014/03/09/train-and-railway-safety-systems. Accessed 12th July 2024.

Toto S. (2012). *Sleep Buster: Japanese Company Develops Anti-Sleep Driver Seat Sheet.* TechCrunch.com, January 2, 2012.

Townsend J. (2019). *Tram Operations Ltd Presentation to Public Transport Liaison Panel.* Tram Operations Limited at Croydon Council Meeting, 27 February 2019. https://democracy.croydon.gov.uk/documents/s13859/Tram%20Operations%20LTD.pdf. Accessed 26th June 2024.

Tufuor T. (2017). *Using Active IR for eye detection and tracking*. Tufts University, ECE Senior Capstone Project.

von Jan T., Karnahl T., Seifert K., Hilgenstock J., and Zobel R. (2010). Don't Sleep and Drive — VW's Fatigue Detection Technology.

Whitlock A. (2002). *Driver Vigilance Devices: Systems Review*. Quintec Associates Limited, Hazelmere. August 30, 2002

Wi-Tronix. (2016). *Distracted Driving Detection for Train Safety*. [Video] Wi-Tronix LLC. https://www.youtube.com/watch?v=gaS8MQ3nt14. Accessed 12th July 2024.

Zen. (2019). *The Minimum Internet Speeds for Streaming Video in 2019*. Zen Internet Ltd. https://www.zen.co.uk/blog/posts/zen-blog/2019/04/18/the-minimum-internet-speeds-for-streaming-video-in-2019/. Accessed 12th July 2024.

Zhang X., Li J., Liu Y., Zhang Z., Wang Z., Luo D., Zhou X., Zhu M., Salman W., Hu G., and Wang, C. (2017). *Design of a Fatigue Detection System for High-Speed Trains Based on Driver Vigilance Using a Wireless Wearable EEG*. Sensors (Basel, Switzerland), 17(3), 486.

Appendix A. Requirements of Current Vigilance System

A.1 European Norm EN14033-1

Below is an extract of standard EN14033-1, “Railway applications — Track — Railbound construction and maintenance machines, Part 1: Technical requirements for running” (BS EN 14033-1:2017 Clause 14.11 on Driver's vigilance device):

The driver’s cab shall be equipped with a means to monitor the driver’s activity, and to automatically stop the machine when a lack of driver’s activity is detected.

The driver’s activity shall be monitored when the machine is in running mode and is moving. This monitoring is permitted to be done by noting the action of the driver on dedicated devices (pedal, push buttons, sensitive touches) and/or the driver's action on the Train Control and Monitoring System and/or the driver's vigilance by indirect means. When no action is monitored for more than 5 s, the vigilance monitoring shall start.

After a time not longer than 60 s without detecting driver’s activity the lack of driver's activity shall be triggered.

Before triggering a lack of driver’s activity, a warning shall be given to the driver, in order for him to have the possibility to react and reset the system.

The system should have the information “lack of driver’s activity triggered” available for being interfaced to other systems (e.g. the radio system).

The detection of the lack of the driver’s activity is a function that shall be subject to a reliability study considering the failure mode of components, redundancies, software, periodic checks and other provisions, and the estimated failure rate of the function (lack of driver’s activity as specified above not detected) shall be provided in the technical documentation. When existing systems with known service experience are used, it is permissible that this study is not necessary.

Specification of actions triggered at machine level when a lack of driver’s activity is detected:

- *a lack of driver’s activity when the machine is in running mode and is moving (criterion for movement detection is at a low speed threshold) shall lead to a full service brake or an emergency brake application on the machine;*
- *in case of application of a full service brake, its effective application shall be automatically controlled and in case of non-application, it shall be followed by an emergency brake.*

It is permitted to have the function described in this clause fulfilled by the in-cab signalling and control systems.

A.2 Commission Regulation (EU) No 1302/2014

Below is an extract of Commission Regulation (EU) No 1302/2014 of 18 November 2014 concerning a technical specification for interoperability relating to the rolling stock — locomotives and passenger rolling stock subsystem of the rail system in the European Union (Clause 4.2.9.3.1 on Driver's Activity Control Function):

(1) The driver's cab shall be equipped with a means to monitor the driver's activity, and to automatically stop the train when a lack of driver's activity is detected.

(2) Specification of the means to monitor (and detect a lack of) the driver's activity:

The driver's activity shall be monitored when the train is in driving configuration and is moving (criterion for movement detection is at a low speed threshold); this monitoring shall be done by controlling the action of the driver on dedicated devices (pedal, push buttons, sensitive touches...) and/or his action on the Train Control and Monitoring System and/or his vigilance by indirect means.

When no action is monitored during more than a time of X seconds, a lack of driver's activity shall be triggered.

The system shall allow for the adjustment (at workshop, as a maintenance activity) of the time X within the range of 5 seconds to 60 seconds.

When the same action is monitored continuously for more than a time not higher than 60 seconds, a lack of driver's activity shall also be triggered.

Before triggering a lack of driver's activity, a warning shall be given to the driver, in order for him to have the possibility to react and reset the system.

The system shall have the information ―lack of driver's activity triggered‖ available for being interfaced to other systems (i.e. the radio system).

(3) Additional requirement:

The detection of the lack of the driver's activity is a function that shall be subject to a reliability study considering the failure mode of components, redundancies, software, periodic checks and other provisions, and the estimated failure rate of the function (lack of driver's activity as specified above not detected) shall be provided in the technical documentation.

(4) Specification of actions triggered at train level when a lack of driver's activity is detected:

A lack of driver's activity when the train is in driving configuration and is moving (criterion for movement detection is at a low speed threshold) shall lead to a full service brake or an emergency brake application on the train. In case of application of a full service brake, its effective application shall be automatically controlled and in case of non application, it shall be followed by an emergency brake.

Notes:

— It is allowed to have the function described in this clause fulfilled by the CCS Subsystem.

— As a transitional measure, it is also allowed to install a system of a fix time X (no adjustment possible) provided that the time X is within the range of 5 seconds to 60 seconds.

— A Member State may ask for a maximum fix time for safety reasons, but in any case it cannot prevent the access to a railway undertaking that using a higher time Z (within the range specified), unless that Member State is able to demonstrate that the national safety level is endangered.…

Appendix B. Railway Industry Acronyms

Acronym	Expansion
AWS	Automatic Warning System
CSM-RA	Common Safety Method on Risk Evaluation and Assessment
DAS	Driver Advisory System
DCO	Driver Controlled Operation
DRM	Driver Resource Manager
DSD	Driver Safety Device
DVD	Driver's Vigilance Device
ERTMS	European Railway Traffic Management System
ETCS	European Train Control System
GSM-R	Global System for Mobile Communications-Railway
LOC&PAS TSI	Locomotive & Passenger Rolling Stock Technical Specification for Interoperability
MoE	Measure of Effectiveness
MoP	Measure of Performance
ORR	Office of Rail Regulation
OTCM	On-Train Camera/Monitors
OTMR	On-Train Monitoring Recorders
RAIB	Rail Accident Investigation Branch
RDG	Rail Delivery Group
RSSB	Rail Safety and Standards Board Ltd
SPAD	Signal Passed at Danger
TBC	Traction/Brake Controller
TCMS	Train Control and Monitoring System
TOCs	Train Operating Companies
TPWS	Train Protection and Warning System
TfL	Transport for London

ISSN 2754-1118 (Online)
ISSN 2753-6599 (Print)

Towards Defect-based Testing for Safety-critical ML Components

Amit Sahu and Carmen Carlan

Edge Case Research GmbH, Munich, Germany

Abstract

The input space of Machine Learning (ML) components used in safety-critical applications is complex. Testing such components on exponentially large datasets that cover all potential real-world situations to meaningfully measure performance metrics, such as the false negative rate, is infeasible due to practical restrictions. Consequently, we must limit the test data while adequately covering critical input data points. Inspired by defect-based software testing, a method for specifying adequate test cases for software components, we propose a process for collecting adequate test data for ML components. Concretely, we systematically employ different existing ML data quality metrics, and methods for enhancing the test data, to uncover critical scenarios where the ML component may be less performant. We exemplify the usage of our process in two case studies, each involving an ML component implementing different functionalities, i.e. stop sign recognition, and railway track segmentation.

1 Introduction

Automated Driving (AD) systems have an unpredictable and complex operational domain. Therefore, AD functions are (partially) implemented by Machine Learning (ML) components, which use training data to approximate the target function rather than using the specific requirements.

ML components that are used to implement safety-critical tasks must demonstrate that they are sufficiently performant. To measure the performance of ML components, different metrics like accuracy or false negative rate are measured on test datasets. However, it is difficult to extrapolate the performance results obtained while exercising ML components with test datasets, to the entire operational domain. For example, a test dataset may be biased (Pagano et al. 2023), imbalanced (Ashmore et al. 2021), or may not be representative of the operational input domain (Zendel et al. 2015). Also, critical scenarios may not be well represented (Cheng et al. 2018b). Here, critical scenarios are scenarios in which the poor performance of an ML system leads to invalidation of system safety goals. Further, test datasets may not have good coverage of the data distribution in the feature subspace (Mani et al. 2019). Consequently, there is little confidence in the performance measured while exercising the ML component with test datasets when used as evidence in safety arguments.

While current standards and guidelines, such as ISO 21448:2022, require that the training, testing, or validation datasets used are good approximations of the real world, they do not provide practical guidance for how to collect such datasets.

In traditional software testing, to measure the performance of components, good test cases are defined and created. For example, in the context of defect-based testing, a "good" test case is defined as a test case that, when executed, has the potential of revealing system defects (Pretschner 2015). Defect-based software testing builds defect models to capture potential defects. These are defined and used to search for good test cases. In the ML literature, with similar objectives, metrics for measuring the quality of ML data, such as scenario coverage or domain shift, have been proposed. However, there is still no indication of how using these ML data quality metrics supports the detection or reduction of ML component defects.

In this paper, inspired by concepts from defect-based software testing, we propose a novel method for collecting, generating, or selecting ML test data. New ML test data may be collected from sensors and new data points may be selected from newly collected ML test data. Further, specific syntactic test data points may be generated, for example, by setting parameters in a simulator. Further, we provide requirements on new good test data points. These can be used either to filter (data selection) or search for (data collection) test data points. We first define what an adequate test dataset is and then specify a systematic process to collect adequate test datasets. To ensure that different types of ML defects can be identified during testing, our method recommends that different ML performance and data quality metrics are meaningfully combined for collecting the test dataset. To this end, we propose investigating which ML defects may be identified, eliminated, or mitigated using a specific ML metric. We showcase the execution of the process given an exemplary set of ML data quality metrics available in the Neural Network Dependability Kit (NNDK) , an open-source toolbox supporting safety engineering of neural networks.

Next, in Section 2, we discuss concepts upon which our work builds, such as software defect-based testing, ML-specific defects, and ML metrics. Then, in Section 3, we elaborate on our process for collecting adequate test data. In Section 4, we exemplify how a set of selected ML data and performance metrics can be complementary used to collect test data. We continue in Section 5 by exemplifying the usage of our method on two case studies, whereas in Section 6, we discuss the open challenges for using ML metrics as evidence in safety arguments. Section 7 positions the contributions of our work in the current state of the art. Finally, in Section 8, we summarize the contributions of this work and discuss the next steps.

2 Background

2.1 Defect-based Testing

Our approach for collecting adequate test data for ML safety-critical components is inspired by defect-based testing from the software domain. "*A good test case reveals a potential defect with good cost-effectiveness*" (Pretschner 2015). Defect-based testing argues this concept in a systematic manner. Defect-based testing should be applied when random testing does not support the identification of sufficient system defects, being most efficient when a set of recurring defects can be defined. A software defect is an error, a fault, or a failure within the source code or within the operating environment of the software, which causes a deviation from the desired behaviour of the system.

In defect-based testing, defect classes need first to be defined. A defect class specifies a deviation from the system specification in a set of behaviour descriptions. Examples of software defect classes are division-by-zero, null pointer dereferencing, stuck-at-one, and

array index overflow. Whereas these are examples of domain-independent defect classes, domain-specific defect classes may also be specified. For example, out of distribution (but within the Operational Design Domain (ODD)) is a defect specific to data used for training ML components. The ODD, as defined by SAE J3016_202104, describes the "*operating conditions under which a given driving automation system, or feature thereof, is specifically designed to function, including, but not limited to, environmental, geographical, and time-of-day restrictions, and/or the requisite presence or absence of certain traffic or roadway characteristics*".

Second, based on the defined defect classes, defect models for these classes shall be specified. Defect models are single instances of defect classes. They may be used to search for test cases, as they partition the input domain of the system into inputs that lead to incorrect behaviour, i.e. the defect domain, and inputs that lead to correct behaviour.

Third, to determine the "goodness"/adequacy of a test case, fitness functions may be used (Kolb et al. 2021). A fitness function assigns to a test case a fitness value, indicating which test cases challenge the system, i.e. can detect system defects. The fitness function must be specific to the use case. In the software domain, the fitness function could be running the code in multiple scenarios and counting the number of errors generated by a specific input. For example, inputting "null" values into a calculator program and doing addition, subtraction, division, and multiplication scenarios. The more errors are generated, the better the test case, and thus, the fitness function should assign a high quality to the respective input. In ML domain, a fitness function may measure accuracy, false negative rate (FNR), mean Intersection over Union (mIoU), precision, or recall.

2.2 Defects Specific to ML Components and ML Metrics

Padilla et al. (2020) identify a set of defects related to the performance of ML components. First, a False Positive (FP) specifies the incorrect detection of a non-existent object or the misplaced detection of an existing object. Second, a False Negative (FN) indicates an undetected ground-truth bounding box or class. Other ML performance defect classes can be derived from ML performance metrics. Examples of performance metrics include precision, recall, per class accuracy, per class confidence score, and robustness against Out of Distribution (OOD) samples.

ML data defect classes can be derived from ML data metrics. An ML data defect class is the fact that the considered ML data does not meet the target value defined for a specific ML data metric. There are several ML metrics proposed by the ML Engineering community (Ashmore et al. 2021) (Tambon et al. 2022) (Willers et al. 2020) scoping the detection and/or the reduction of FPs and FNs. Examples of data quality metrics are k-projection coverage, neuron activation pattern, perturbation loss, scenario coverage, data augmentation (Cheng et al. 2018b), unknown behaviour in rare critical situations, adversarial attacks (Goodfellow et al. 2015), layer-wise relevance propagation (Montavon et al. 2019) for transparency and explainability in neural networks, and sanity checks during model development.

3 Proposed Process for Collecting Adequate Test Data

We use the concepts from defect-based software testing to develop a systematic technique for collecting, generating, or selecting adequate test data for ML-based components that

implement safety-critical functionality. The technique synthesizes test data samples from ML defect classes.

As there is no generally accepted definition for adequate test data for ML-based components, in the same line of thought as defect-based testing, like good test cases, we define a "good test dataset" as a dataset that will exercise the system with problematic inputs, scoping at uncovering defects of ML components. As a result, the ML performance values measured during testing will most probably initially suffer a decrease. Whereas a good test dataset uncovers possible defects in the system, an adequate test dataset will uncover defects belonging to all identified safety-critical defect classes. Safety engineers may argue about which defect classes are safety-critical using safety arguments. Such safety arguments shall show how component-level performance measurements indicate the satisfaction of system-level safety goals.

Next, we specify a four-step process for collecting test data by combining different available ML metrics and methods. The scope of this is to maximize the number of ML defect classes addressed by the test data. ML metrics and methods are used to measure the quality of data, or to add new test data points to reach the targeted values (as per requirements) for the metrics.

Step 1: Select an ML data metric/method and identify the addressed ML defect class(es);

Step 2: Measure the metric/execute the method. If the target value is not met, collect new data points;

Step 3: To measure the quality of the newly added data points, apply one or more fitness functions. A fitness function assigns to the test dataset a value obtained from the measurement of an ML performance metric, such as a false negative rate. If the newly collected data points are effective test data points, then the measured performance value will initially drop, meaning that system defect models are detected during the testing of the system given the collected test data points. This step will allow the engineers to control the size and quality of the test dataset by acting as a filtering mechanism with a minimum threshold on the fitness function;

Step 4: Optionally, to increase the probability of uncovering defects belonging to different defect classes, select a new data metric/ method. Then, execute again Step 1 to Step 3 for the newly selected metric/method, and, if needed, also Step 4.

Even after the addition of data points collected while using the ML metrics, there will still be a considerable number of data points that will not be tested, and for which there is no guarantee for how the system will behave. To compensate for the data points in the input space that are not covered by the test dataset, a filtering mechanism to restrict ML output in such scenarios is needed. A filtering mechanism would ensure that defect models belonging to unknown, or known but not addressed, defect classes are identified during system deployment. For example, one such filtering mechanism could be implemented by runtime monitors, which enable the detection and handling of defects at runtime.

4 Application

A pre-requirement of our technique is to have a pre-selected set of ML data quality and ML performance metrics. While the adequacy of the data depends on the selected metrics, the selection of the metrics is out of the scope of this work, and needs to be further explored in the future. In practice, system developers use ML metrics for which they have tool support or about which they have the most expertise. Further, safety engineers should

reason about how the selected metrics provide sufficient and trustworthy evidence about the data quality in the system safety case. In Section 7, we discuss state-of-the-art ML metrics that, with the help of our technique, could be combined to collect good test data.

Other pre-requirements are the existence of an initial test dataset and system-level safety requirements. The system-level safety requirements are needed to derive the target values for the selected ML metrics.

Next, we explain how we combined a concrete set of selected ML metrics to evaluate and enhance test datasets. In this example, we consider the ML metrics proposed in the Neural Network Dependability Kit (NNDK) (Cheng et al. 2018a), an open-source toolbox to support data-driven engineering of neural networks for safety-critical domains. First, to check whether the initial test dataset sufficiently covers the ODD, NNDK provides tool support for measuring the k-projection scenario coverage during the data management development stage. To this end, an ODD of the system shall be specified. Second, during the ML model learning and validation development stages, NNDK can be used to measure the perturbation loss metric, which assesses the vulnerability of the ML component to noise. Third, the neuron activation pattern coverage can be used during the model verification (testing) stage for detecting missing feature combinations. Fourth, NNDK makes available runtime monitors that can be used during model deployment (system operation) to detect new system defects based on data from the operating context.

In Figure 1, we illustrate how to argue about test data adequacy based on evidence generated while executing our proposed four-step method given the NNDK metrics. As proposed by Hawkins et al. (2021), such an argument shall be part of any argument about ML safety assurance. The argument is based on a combination of design-time and runtime evidence. Each tree branch uses a metric used in one of the development stages. In this work, the considered development stages are data management, ML model learning and validation, model verification (testing), and model deployment. In the figure, we depict the data points within the test dataset as red circles and the data points in the rest of the input space as green triangles. The figure shows that green triangle points shift to red circle points as new test data points are created while measuring and optimizing different ML metrics. For example, applying adversarial attacks results in new red circle points.

First, in Step 1 of the technique proposed in Section 3, we identify the ML defect class addressed by the NNDK k-projection scenario coverage metric. Given a test dataset and the ODD of a system, the k-projection scenario coverage metric identifies if the dataset fails to cover the ODD. The metric uses the meta-labels (other than the prediction class) of the data points to check the coverage of different combinations of scenarios and situations.

Then, in Step 2, the metric is to be measured for a given ML model and a test dataset, and the measured value is to be compared against the target value defined based on the system requirements. If the measured value does not meet the target value, the test dataset must be enhanced. Namely, to reach the target value for k-projection coverage, new data points with new combinations are to be added to the test dataset. The metric can suggest ideal data points with the most potential in increasing the metric for example, sunny with snow and a rainbow. However, these suggestions need to be filtered by the probability of real-life occurrences and acquiring difficulty. In Figure 1, we show how new data points are added to meet the target value.

In Step 3, we measure the quality of the new test data points using a fitness function (performance metric) to further filter the high-quality test cases as good test cases cause defects. If the measured values of the selected fitness function are less when the system is exercised with the new test data than when the system was exercised with the initial test

data, then critical scenarios in which the system does not perform well (high-quality) are present in the new data points.

In Step 4, we iteratively consider other ML data metrics that can be measured with NNDK, and for each of these metrics, we execute again Steps 1 to 3. The analysis of the newly uncovered critical scenarios may also be used to improve the system's performance in the next development iteration.

To check another ML defect class, namely the vulnerability of the ML component to noise, similarly to fault injection software testing methods, in the domain of ML verification, it is commonly accepted to use data augmentation and the addition of noise (Adversarial (Goodfellow et al. 2015), Bayesian, etc.). Since the datasets are only supplemented and not reduced, the domain coverage of the dataset is not affected.

Inspired by software testing, where structural and requirements coverages are computed, neuron activation patterns can be used for ML components to detect (neuron) features not contained in the dataset. More complex approaches of coverage testing like Mani et al. (2019) could also be used. The NNDK neuron coverage, like k-projection coverage, can suggest characteristics of new samples that will increase the test coverage.

Since these steps can generate different numbers of new test cases, a balancing based on the total number of all defect classes should be considered. As the final step, the NNDK runtime monitor can be added to the system architecture with the scope of detecting OOD data points. The ML component should be restricted from computing these points, as the output is unknown. Data points (during operation) still passing the runtime monitor, but failing the system (high FNR), could be considered as a measure of the leftover risk of using the ML component inside the system. This is the final filter to build an adequate test dataset (of good test cases) for the next iteration.

5 Evaluation

5.1 Preamble

Next, we demonstrate that the usage of NNDK metrics for enhancing test datasets, as discussed in Section 3, indeed adds value to the test datasets, meaning that the enhanced datasets uncover more defects. To this end, we apply the process proposed in Section 4 in two small systems. The first is a stop sign recognition component inside an autonomous vehicle system, and the second is a railway track segmentation component for an autonomous train.

The Stop Sign Recognition component is tasked with recognizing stop signs on the road for an autonomous vehicle. The basic ODD is all weather conditions, lighting, and size. In a real environment, the ODD will be more complex and include combinations of scenarios, for example, sign age, and condition. We are limited by the dataset only to images of stop signs in different scenarios.

The Track Segmentation component is responsible for identifying the pixels of the railway track in images with cityscapes. Hence, in addition to weather, lighting, and size the ODD will require different combinations of the scenarios happening in the environment. We considered labels like 'road', 'sidewalk', 'construction', 'tram_track', 'fence', 'pole', 'traffic_light', 'traffic_sign', 'vegetation', 'terrain', 'sky', 'human', 'rail_track', 'car', 'truck', 'trackbed', 'on_rails', 'rail_raised', 'rail_embedded', 'background'.

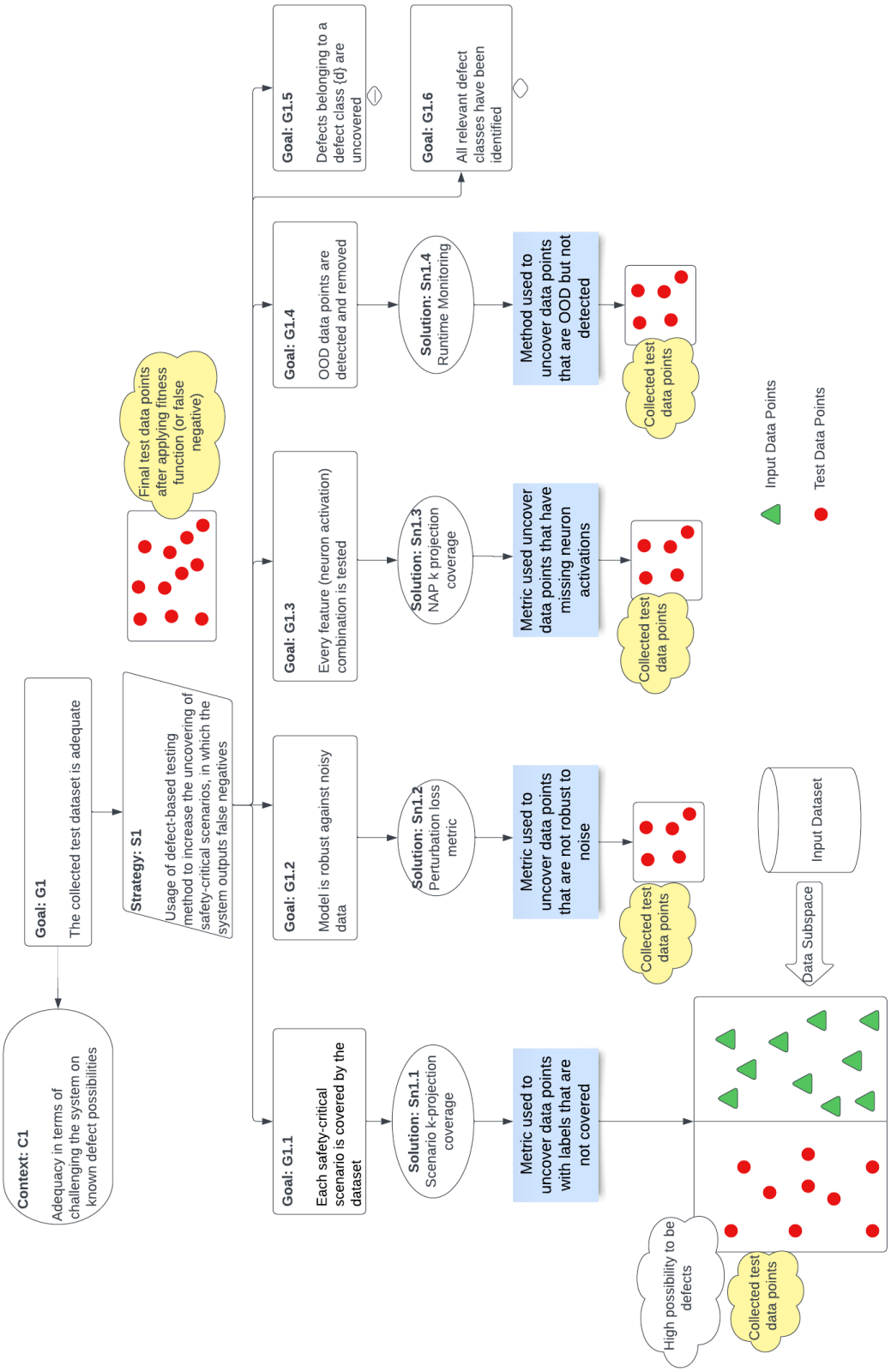

Figure 1~ Argument Explaining the Collection of Test Data Using NNDK Metrics

One pre-requisite of our process is the existence of an initial test dataset. Consequently, for each of the two case studies we consider in this work we first selected a testing dataset, which we used to measure selected ML performance metrics. Then, we applied our

process for collecting good test data points, and we measured again the selected ML performance metrics. The values of the ML performance metrics on the new enhanced test dataset suffered a decrease (Table 1 and Table 2). This means that the enhanced test dataset contains critical data points, that uncover the component's functional insufficiencies.

5.2 Initial Test Datasets

Whereas for the Stop Sign Recognition component we used as initial test dataset the German Traffic Sign Recognition Benchmark (GTSRB) (Stallkamp 2021), for the railway track segmentation component we used the RailSem19 dataset (Zendel et al. 2019).

The GTSRB dataset consists of 43 classes of traffic signs, and the images have varying light conditions and rich backgrounds.

The RailSem19 dataset for the Track Segmentation model consists of 8500 unique images from the ego-perspective of rail vehicles (trains and trams). The dataset provides extensive semantic annotations: geometry-based (rail-relevant polygons, all rails as polylines), and dense label maps with many road labels in Cityscapes.

5.3 Considered ML Performance Metrics

To measure the performance of the stop sign recognition component, we used the FNR performance metric. In the case of stop sign recognition, false negatives may cause more severe accidents than false positives.

For railway track segmentation, we used mIoU to measure the performance of the ML component. Since it is a segmentation model that assigns each pixel a classification, intersection and union with the ground truth is the standard performance metric.

5.4 Experiments

Table 1 and Table 2 show the values measured with NNDK for each of the considered case studies. Note: In Table 1, FGSM is the Fast Gradient Sign Method.

Table 1 ~ NNDK Results Measuring the Quality of the GTSRB Dataset

Data QualityMetric	**Basic Iteration (Performance metric)**	**The 5-step Process (Performancemetric)**
Perturbation Loss Metric (Snow)	3.33% (FNR)	32.6% (FNR with snow)
Adversarial Attack (FGSM)	N/A	25.82% (FPR)
Neuron k-projection coverage	3.33% (FNR)	98.33% (FNR with complete synthetic coverage)
Runtime Monitor	N/A	7.9% (FNR with Snow) and 22.5% (FPR with FGSM)

In Table 2, GN is Gaussian Noise.

Table 2 ~ NNDK Results Measuring the Quality of the RailSem19 Dataset

Data QualityMetric	**Basic Iteration (Performance metric)**	**The 5-step Process (Performance metric)**
Scenario k-projection coverage	57.68% (mIoU)	43.87% (mIoU with Fog)
Perturbation loss (GN)	57.89% (mIoU)	4.83% (mIoU with GN)

Initially, the FNR on the GTSRB dataset was 3.33%. We then applied snow perturbation, which was a missing label in the covered scenarios, to increase k-projection coverage. The FNR of the validation dataset dropped to 32.6%, indicating that snow is a limiting factor for the performance of the ML component. Further, we enhanced the test dataset with data points including iterative Fast Gradient Sign Method (FGSM) adversarial attack. While exercising the ML component with the new data points, the false positive rate was 25.82%. In other words, images in the dataset were attacked with noise to make them falsely classify as a stop sign and succeeded in 25.82% of the cases.

Modifying the dataset images synthetically to increase the Neuron Activation Pattern (NAP) metric resulted in an FNR of 98.33% on the synthetically created dataset. The synthetic images had neuron activation patterns that were missing in the test dataset. Hence, augmenting the test dataset with the synthetic images resulted in complete 2-projection coverage. The complete synthetic image creation process is explained in the NNDK paper (Cheng et al. 2018a).

Finally, we applied a runtime monitor based on the NAP monitoring technique from NNDK. After applying the runtime monitor, the FNR of the leftover images was reduced to 7.9% for snow perturbation and 22.4% for iterative FGSM. These results indicate the potential of the NNDK ML metrics to uncover critical scenarios. Given only these metrics, collecting the test data points using all the metrics resulted in an improved dataset, which helped with finding issues with the ML model. However, the adequacy of the test dataset is dependent on the data quality metrics as evidence. In the future, we plan on building an argument about the adequacy of a test data using the NNDK metrics.

For our second experiment, we trained a modified DeepLabV3Plus model (Chen et al. 2018) from github2 on the RailSem19 dataset. We modified the model parameters to keep the model within a limited GPU memory of 8 GB. The model was trained on 4200 images, and resnet34 encoder with ImageNet weights. This resulted in a mIoU performance score of 57.68% on 1000 test images.

On the RailSem19 dataset, the k-projection coverage was computed using the labels mentioned in the ODD. As an initial assessment k=2 was taken as an input for the metric. The 2-projection coverage using ODD labels gives us a high value of 99.71% (712 out of 714 two-label combinations). However, these labels were manually selected. Instead, to get a value confidently reflecting the current performance of the ML component, all relevant labels for the operational domain should be added. For example, the weather labels in test images included sunny, rainy, and cloudy weather but missed fog. Consequently, the synthetic noise of fog was added to all the test images, and then the mIoU was computed on the same dataset. The score dropped from 57.68% (without fog) to 43.87% (with fog).

To argue about the performance of the considered railway track segmentation model under sensor noise, the performance under Gaussian noise (GN) with *std = 0.1* was also

computed. The performance dropped from 57.89% (without GN) to 4.93% (with GN). As in the case of the first use case, these results show the potential of the NNDK ML metrics in uncovering critical scenarios that need to be tested to make the system more performant.

As data collection is a tedious process, for exemplification, in the two considered case studies, the data collection process was addressed as a data augmentation process. This has been done due to the reduced resources we had while conducting the case studies. Still, when implemented in a real-life project, our process assumes the collection of data by testing on a test track or operating in a real-world environment. Test data may also be collected from simulations or standard benchmark datasets.

6 Discussion

Continuous identification of critical test data points. Finding all defects during system development is difficult, given the uncertainties due to the complexity of the input space, the lack of interpretability of the ML components, and the complexity of the operational environment. Each step in the lifecycle of an ML component aims at identifying and mitigating ML defects to reach the required performance. Consequently, test data collection is continuous throughout the system lifecycle, meaning that the different considered ML metrics may be applied at different development stages of an ML component. Further, at runtime, given operating scenarios that were not considered during design time, system defects may be uncovered. In this work, we assume a continuous system development process, during which data samples are continuously added to the test dataset, covering newly identified operating scenarios.

Focus on test data. Defect models provide a systematic process to uncover new good test cases. Hence, our process is to augment the original test dataset. We consider adequacy based on knowledge provided by the metrics. Other aspects of the test dataset are not considered. The method can also be applied for validation or training datasets, but the intention of the process is to test the system with more difficult scenarios. Hence, the learning and validation benefits must be subjectively analysed as per the use case.

The importance of using adequate ML metrics. ML metrics are used during the development of the ML model to measure different properties of ML components, such as robustness against noise. However, these metrics do not guarantee that the model satisfies such properties. Since our process is dependent on the used ML metrics, it is as good as the used metrics. For example, in the second case study, replacing neuron k-projection metric with coverage testing (Mani et al. 2019) would have resulted in a higher-quality test dataset. Trusting the metric and the threshold defined as the target value acquired by experimentation on a variety of datasets is currently the only choice. While the usage of formal methods (Beyene and Sahu 2020) (Katz et al. 2017) has been researched to reach some guarantees, the application of such methods is only feasible when the ML models are small (a few thousand neurons), whereas ML models implementing safety-critical functions are usually considerably big, with millions of neurons and parameters (Redmon and Farhadi 2018). Safety arguments could be used to reason about the adequacy of the measured ML metrics to be used as safety evidence.

Additional general metrics. While specific ML methods may not be related to data directly, some methods like data augmentation, or adversarial attacks, are useful for identifying the defect models, as we show in our use cases. Adversarial defects point us to differences in features used by ML models as compared to humans. Since they are

artificially created, the possibility of their occurrence in the ODD should be validated and then used accordingly (Dilmegani 2024).

Data synthesis instead of data collection. While our process recommends the collection of new test data, this data collection process is usually tedious. Consequently, while conducting our case studies, we replaced data collection with data synthesis (namely, data augmentation). Further research is to be done on whether synthesized test data could also be used.

7 Related Work

7.1 Defining Test Data Adequacy

Having a good quality test dataset is an important prerequisite for measuring the performance of an ML component. Kim et al. (2023) define the adequacy of test datasets in terms of the degree of "out-of-distribution-ness" of a given input. The more the model gets "surprised" from the test input (compared to training input), the more adequate the test dataset. In contrast, Deepgauge (Ma et al. 2018a) uses minimum and maximum neuron activation values and k-multisection neuron coverage for gauging the adequacy of test datasets. Sun et al. (2018a) use Modified Condition/Decision Coverage (MC/DC) (Kelly et al. 2001) to define four novel criteria to structure features and semantics of deep neural networks, which are used to measure the adequacy of the test dataset. The main idea behind this method is that all the conditions that contribute to a decision must be tested.

These state-of-the-art approaches do not contradict each other, but are complementary. Whereas these approaches address single characteristics of the test dataset, we propose a higher-level definition of test data adequacy, stating that an adequate test dataset can detect all possible types of system defects. To enable the creation of adequate test datasets, we propose a four-step data collection process employing different ML methods and metrics. While the approach proposed by Kim et al. (2023) can be seen as an ML data metric to be used in our process to evaluate and enhance the collected data, the approach proposed by Sun et al. (2018a) and Ma et al. (2018a) propose ML performance metrics that can be used as fitness functions.

7.2 Systematically Searching Through the Input Space

In the literature, different approaches for generating test cases have been proposed. Such approaches may be used to collect ML test data.

CV-HAZOP (Zendel et al. 2015) combines guide words and parameters to systematically investigate the critical scenarios at every location in the system. The parameters refer to the physical and operational aspects of the sub-component configuration, whereas a guide word is a short expression to express the deviation from the intent of the design or process. For example, light source intensity (parameter) and "More" (guide word) will result in "Overexposure of lit objects" as a suggested hazard.

Zhao et al. (2021) introduced a Reliability Assessment Model (RAM) for ML classifiers. First, system-level safety targets are broken down into component-level requirements and claims supported by reliability metrics (Claims Arguments Evidence structure). Second, the input domain space is split into cells like a geometrical grid. Third, the operational profile, i.e. the possible output map of the component, and the robustness of these input

cells are measured, and the measurements are then combined to estimate the reliability of the claim.

Mani et al. (2019) proposed a method for test case generation using the quality aspects of the feature space: equivalence partitioning, centroid positioning, boundary conditioning, and pair-wise boundary conditioning are also provided.

Approaches modified from traditional software testing to accommodate them for ML models like mutation testing (Ma et al. 2018b), concolic testing (Sun et al. 2018b), illumination search (Zohdinasab et al. 2021) are different defect classes.

We propose a systematic process for using well-known methods to uncover critical scenarios and test cases. There are a number of these methods and new ones are constantly published.

Dola et al. (2024) applied combinatorial interaction testing to the latent space of generative models and can generate rare and fault-revealing test inputs. If applicable, this would be a good replacement for the NNDK neuron activation pattern. Hirschle et al. (2023) proposed a workflow to generate high-level scenario descriptions that are parameterized with the operational envelope data of the ML system. Synthetic data can then be simulated based on these parameters. This can be a replacement for NNDK k-projection coverage. Hartjen (2023) proposes a method for scenario management and their semantic classification.

In Alshareef et al. (2023), feature importance weights, that are weighed as per the contribution on the model output, are used to measure the coverage of the test set. This also facilitates the generation of additional test cases. This approach is a good replacement for the NNDK neuron activation pattern.

Whereas the above approaches propose test cases using weights, latent space, or scenario classification methods, our paper proposes an approach for combining (many) well-known methods to build an adequate test dataset. Individually all these approaches supplement our approach as individual metrics.

8 Summary and Future Work

Our work was motivated by the problem of collecting adequate test datasets for ML components implementing safety-critical functionality. To solve this problem, we defined a systematic, four-step data collection process, which uses state-of-the-art ML data and performance metrics and concepts behind defect-based testing to operationalize the acquisition of high-quality test data points. The proposed approach aims to be complementary to other test approaches, such as unguided and guided random testing. We showcased the usage of the process in two small case studies implementing different functionalities, i.e. stop sign recognition and railway track segmentation.

Next, we plan to apply the proposed technique in a more complex case study, employing a higher number of ML metrics.

Another line of future work is to provide technical solutions for executing each of the steps in the proposed process. To execute Step 2, novel techniques for generating new inputs and their corresponding labels are needed. To support the execution of Step 3 and 4, we plan on proposing a selection mechanism as an aid for choosing the ML metrics and methods used in the process. Here, one way to combine more systematically different ML metrics to collect test data points is to cover the known limitations of one ML metric with

other metrics. For example, one limitation of the k-projection coverage metric is that it does not ensure that the dataset is balanced. Namely, the metric does not provide information about a specific scenario's number of data points, i.e. even one data point can satisfy k-projection coverage for a critical scenario. This could be easily remedied by checking the proportion or representation of each scenario in the number of input samples. After that, data balancing techniques like oversampling, under-sampling, or class weight can be used.

Further, designing run-time monitors that can detect scenarios not covered by the dataset is a challenging task, which can be further investigated.

Acknowledgments

This research received funding from the Federal Ministry for Economic Affairs and Climate Action (BMWK) and the European Union under grant agreement 19I21039A.

References

Alshareef A., Berthier N., Schewe S., and Huang X. (2023). *Weight-based Semantic Testing Approach for Deep Neural Networks*. In: The IJCAI-2023 AISafety and SafeRL Joint Workshop.

Ashmore R., Calinescu R., and Paterson C. (2021). *Assuring the machine learning lifecycle: Desiderata, methods, and challenges*. ACM Computing Surveys (CSUR), 54(5), 1-39.

Beyene T. A., and Sahu A. (2020). *Rule-based safety evidence for neural networks*. In: Computer Safety, Reliability, and Security. SAFECOMP 2020 Workshops: DECSoS 2020, DepDevOps 2020, USDAI 2020, and WAISE 2020, Lisbon, Portugal, September 15, 2020, Proceedings 39 (pp. 328-335). Springer International Publishing.

Chen L. C., Zhu Y., Papandreou G., Schroff F., and Adam H. (2018). *Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation*. In: Proceedings of the European Conference on Computer Vision (ECCV) (pp. 801-818).

Cheng C. H., Huang C. H., and Nührenberg G. (2018a). *nn-dependability-kit: Engineering neural networks for safety-critical systems*. arXiv preprint arXiv:1811.06746.

Cheng C. H., Huang C. H., and Yasuoka H. (2018b). *Quantitative projection coverage for testing ML-enabled autonomous systems*. In: *Automated Technology for Verification and Analysis: 16th International Symposium*, ATVA 2018, Los Angeles, CA, USA, October 7-10, 2018, Proceedings 16 (pp. 126-142). Springer International Publishing.

Dilmegani C. (2024). *Synthetic Data vs Real Data: Benefits, Challenges in 2024*. https://research.aimultiple.com/synthetic-data-vs-real-data/#some-challenges-with-using-synthetic-data-against-real-data. Accessed 20th July 2024.

Dola S., McDaniel R., Dwyer M. B., and Soffa M. L. (2024). *CIT4DNN: Generating Diverse and Rare Inputs for Neural Networks Using Latent Space Combinatorial Testing*. In: Proceedings of the IEEE/ACM 46th International Conference on Software Engineering (pp. 1-13).

Goodfellow I. J., Shlens J., and Szegedy C. (2015). *Explaining and Harnessing Adversarial Examples*. arXiv preprint arXiv:1412.6572.

Hartjen L. (2023). *Semantic Classification of Urban Traffic Scenarios for the Validation of Automated Driving Systems*. Doctoral Dissertation, Technische Universität Carolo-Wilhelmina zu Braunschweig, “TU Braunschweig”.

Hawkins R., Paterson C., Picardi C., Jia Y., Calinescu R., and Habli, I. (2021). *Guidance on the Assurance of Machine Learning in Autonomous Systems (AMLAS*). arXiv preprint arXiv:2102.01564.

Hirschle M., Kirov D., Aievola R., Sinisi S., Iovino S., and Adamy J. (2023). *Scenario-Based Methods for Machine Learning Assurance.* In: 2023 IEEE/AIAA 42nd Digital Avionics Systems Conference (DASC) (pp. 1-10). IEEE.

ISO 21448. (2022). *Road vehicles — Safety of the intended functionality*. ISO 21448:2022, 1st Edition, International Standards Organization, Geneva.

Katz G., Barrett C., Dill D. L., Julian K., and Kochenderfer M. J. (2017). *Reluplex: An efficient SMT solver for verifying deep neural networks*. In: Computer Aided Verification: 29th International Conference, CAV 2017, Heidelberg, Germany, July 24-28, 2017, Proceedings, Part I 30 (pp. 97-117). Springer International Publishing.

Kelly J, H., Dan S, V., John J, C., & Leanna K, R. (2001). A Practical Tutorial on Modified Condition/Decision Coverage. Technical Report TM-2001-210876. NASA Langley Technical y Research Center. https://dl.acm.org/doi/pdf/10.5555/886632. Accessed 20th July 2024.

Kim J., Feldt R., and Yoo S. (2023). *Evaluating surprise adequacy for deep learning system testing*. ACM Transactions on Software Engineering and Methodology, 32(2), 1-29.

Kolb N., Hauer F., and Pretschner A. (2021). *Fitness function templates for testing automated and autonomous driving systems in intersection scenarios*. In: 2021 IEEE International Intelligent Transportation Systems Conference (ITSC) (pp. 217-222). IEEE.

Ma L., Juefei-Xu F., Zhang F., Sun J., Xue M., Li B., Chen C., Su T., Li L., Liu Y., Zhao J., and Wang Y. (2018a). *Deepgauge: Multi-granularity testing criteria for deep learning systems*. In: Proceedings of the 33rd ACM/IEEE International Conference on Automated Software Engineering (pp. 120-131).

Ma, L., Zhang, F., Sun, J., Xue, M., Li, B., Juefei-Xu, F., Xie, C., Li, L., Liu, Y., Zhao, J., Wang, Y. (2018b). Deepmutation: Mutation testing of deep learning systems. In 2018 IEEE 29th International Symposium on Software Reliability Engineering (ISSRE) (pp. 100-111). IEEE.

Mani S., Sankaran A., Tamilselvam S., and Sethi A. (2019). *Coverage Testing of Deep Learning Models using Dataset Characterization*. arXiv preprint arXiv:1911.07309.

Montavon G., Binder A., Lapuschkin S., Samek W., and Müller K. R. (2019). *Layer-Wise Relevance Propagation: An Overview*. In: Samek W., Montavon G., Vedaldi A., Hansen L., and Müller K. R. (eds). *Explainable AI: Interpreting, Explaining and Visualizing Deep Learning*. Lecture Notes in Computer Science, Vol 11700, pp 193-209. Springer, Cham. https://doi.org/10.1007/978-3-030-28954-6_10. Accessed 20th July 2024.

Padilla R., Netto S. L., and Da Silva E. A. (2020). *A survey on performance metrics for object-detection algorithms*. In: 2020 International Conference on Systems, Signals and Image Processing (IWSSIP) (pp. 237-242). IEEE.

Pagano T. P., Loureiro R. B., Lisboa F. V., Peixoto R. M., Guimarães G. A., Cruz G. O., R., Araujo M. M., Santos L. L., Cruz M. A. S., Oliveira E. L. S., Winkler I., and Nascimento E. G. (2023). *Bias and unfairness in machine learning models: a systematic review on datasets, tools, fairness metrics, and identification and mitigation methods*. Big Data and Cognitive Computing, 7(1), 15.

Pretschner A. (2015). *Defect-Based Testing*. Dependable Software Systems Engineering, 84.

Redmon J., and Farhadi A. (2018). *Yolov3: An incremental improvement*. arXiv preprint arXiv:1804.02767.

SAE. J3016. (2021). *Taxonomy and Definitions for Terms Related to Driving Automation Systems for On-Road Motor Vehicles*. Standard J3016_202104, 30th April 2021. SAE International, Warrendale, PA.

Stallkamp J., Schlipsing M., Salmen J., and Igel C. (2011). *The German traffic sign recognition benchmark: a multi-class classification competition*. In: The 2011 International Joint Conference on Neural Networks (pp. 1453-1460). IEEE.

Sun Y., Huang X., Kroening D., Sharp J., Hill M., and Ashmore R. (2018a). *Testing Deep Neural Networks*. arXiv preprint arXiv:1803.04792.

Sun, Y., Wu, M., Ruan, W., Huang, X., Kwiatkowska, M., & Kroening, D. (2018b). Concolic testing for deep neural networks. In Proceedings of the 33rd ACM/IEEE International Conference on Automated Software Engineering (pp. 109-119).

Tambon F., Laberge G., An L., Nikanjam A., Mindom P. S. N., Pequignot Y., Khomh F., Antoniol G., Merlo E., and Laviolette F. (2022). *How to Certify Machine Learning Based Safety-critical Systems? A Systematic Literature Review*. Automated Software Engineering, 29(2), 38.

Willers O., Sudholt S., Raafatnia S., and Abrecht S. (2020). *Safety concerns and mitigation approaches regarding the use of deep learning in safety-critical perception tasks*. In: Computer Safety, Reliability, and Security. SAFECOMP 2020 Workshops: DECSoS 2020, DepDevOps 2020, USDAI 2020, and WAISE 2020, Lisbon, Portugal, September 15, 2020, Proceedings 39 (pp. 336-350). Springer International Publishing.

Zendel O., Murschitz M., Humenberger M., and Herzner W. (2015). *CV-HAZOP: Introducing Test Data Validation for Computer Vision*. In: Proceedings of the IEEE International Conference on Computer Vision (pp. 2066-2074).

Zendel O., Murschitz M., Zeilinger M., Steininger D., Abbasi S., and Beleznai C. (2019). *RailSem19: A Dataset for Semantic Rail Scene Understanding*. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops.

Zhao X., Huang W., Bharti V., Dong Y., Cox V., Banks A., Wang S., Schewe S., and Huang X. (2021*). Reliability Assessment and Safety Arguments for Machine Learning Components in System Assurance*. arXiv e-prints, arXiv-2112.

Zohdinasab T., Riccio V., Gambi A., and Tonella P. (2021). *Deephyperion: Exploring the Feature Space of Deep Learning-Based Systems through Illumination Search*. In: Proceedings of the 30th ACM SIGSOFT International Symposium on Software Testing and Analysis (pp. 79-90).

This collation page left blank intentionally.

ISSN 2754-1118 (Online)
ISSN 2753-6599 (Print)

About the Safety-Critical Systems eJournal

Purpose and Scope

This is the Journal of the Safety-Critical Systems Club CIC (SCSC), ISSN 2754-1118 (Online), ISSN 2753-6599 (Print). Its mission is to publish high-quality, peer-reviewed articles on the subject of systems safety.

When we talk of systems, we mean not only the platforms, but also the people and their procedures that make up the whole. Systems Safety addresses those systems, their components, and the services they are used to provide. This is not a narrow view of system safety, our scope is wide and also includes safety-related topics such as resilience, security, public health and environmental impact.

Background

When the Safety-Critical Systems Club (SCSC) was set up over thirty years ago, its objectives were to raise awareness of safety issues and to facilitate safety technology transfer. To achieve these objectives, the club organised events, such as Seminars and an annual Symposium, and published a newsletter, Safety Systems, three times a year.

The Newsletter, in addition to news, opinion, correspondence, book reviews, and the like, also carries articles discussing current and emerging practices and standards. The length of such articles is limited to about two and a half thousand words, which does not allow an in-depth treatment. It was therefore decided to add a third string to our bow and supplement the events and newsletter with a journal containing longer papers. That journal is published here, as the Safety-Critical Systems eJournal, and comprises at least two issues a year.

Content Sources

Sources include the outputs of SCSC working groups; solicited technical articles and topic reviews; submitted articles on new analysis techniques, discussion of standards, and industrial practice; and guidelines and lessons learned. If you wish to contribute, please see, "Information for Authors".

Types of paper include, but are not limited to:

Technical Articles: Written by practitioners and describing practical safety engineering and assurance techniques, and their industrial applications.

Integration Studies: Written by practitioners reporting upon successful (or otherwise) synergies achieved in practice with other assurance domains, e.g. systems engineering, reliability/availability/maintainability engineering, resilience, human factors, security, and environment.

Position Papers: Written by, or on behalf of, Regulators, Standardisation Organisations, or other official bodies, setting out their position on a topic, e.g. the interpretation of a particular standard or regulation.

Review Articles: Papers highlighting recent developments and trends in some aspect of safety-critical systems or of their use in a particular industrial sector.

Historical Articles: Papers describing the development of safety assurance in an industrial sector; how we got to where we are today.

ISSN 2754-1118 (Online)
ISSN 2753-6599 (Print)

Perspectives: The authors' personal opinions on a subject, e.g. whether to use statistical methods in particular scenarios.

Reports: The lessons learned from incidents or the outcomes of trials with a description of scenarios, or methods, and a discussion of the results obtained.

Working Group Outputs: Written by, or on behalf of, Safety-Critical Systems Club Working Groups to include discussions, underpinning theory, or guidelines.

Copyright and Disclaimer

Letters to the Editor

The editorial to the first issue of this journal said, "*You may find some of this material controversial, or you may think that it does not go far enough. Subsequent issues of this journal will have provision for readers' letters to the Editor responding to individual papers.*" Such a letter should be no more than 1000 words in length (not counting title, attribution, or references). That would take up no more than two pages of the journal. Note that a letter should ideally address a single concern with few, if any, external references.