Journal of Automotive Software Engineering
Vol. **1(1)**; 2019, *pp.* 1–19
DOI: https://doi.org/10.2991/jase.d.190131.001; ISSN: 2589-2258
https://www.atlantis-press.com/journals/jase/

Safely Entering the Deep: A Review of Verification and Validation for Machine Learning and a Challenge Elicitation in the Automotive Industry

Markus Borg[1,*], Cristofer Englund[1], Krzysztof Wnuk[2], Boris Duran[1], Christoffer Levandowski[3], Shenjian Gao[2], Yanwen Tan[2], Henrik Kaijser[4], Henrik Lönn[4], Jonas Törnqvist[3]

[1]*RISE Research Institutes of Sweden AB, Scheelevägen 17, SE-223 70 Lund, Sweden*
[2]*Blekinge Institute of Technology, Valhallavägen 1, SE-371 41 Karlskrona, Sweden*
[3]*QRTECH AB, Flöjelbergsgatan 1C, SE-431 35 Mölndal, Sweden*
[4]*AB Volvo, Volvo Group Trucks Technology, SE-405 08 Gothenburg, Sweden*

ARTICLE INFO

Article History
Received 28 May 2018
Accepted 13 Dec 2018

Keywords
Deep learning
Safety-critical systems
Machine learning
Verification and validation
ISO 26262

ABSTRACT

Deep neural networks (DNNs) will emerge as a cornerstone in automotive software engineering. However, developing systems with DNNs introduces novel challenges for safety assessments. This paper reviews the state-of-the-art in verification and validation of safety-critical systems that rely on machine learning. Furthermore, we report from a workshop series on DNNs for perception with automotive experts in Sweden, confirming that ISO 26262 largely contravenes the nature of DNNs. We recommend aerospace-to-automotive knowledge transfer and systems-based safety approaches, for example, safety cage architectures and simulated system test cases.

1. INTRODUCTION

As an enabling technology for autonomous driving, deep learning neural networks (DNNs) will emerge as a cornerstone in automotive software engineering. Automotive software solutions using DNNs is a hot topic, with new advances being reported almost weekly. Also in the academic context, several research communities study DNNs in the automotive domain from various perspectives, for example, applied machine learning (ML) [1], software engineering [2], safety engineering [3], and verification & validation (V&V) [4].

DNNs are used to enable *vehicle environmental perception*, that is, awareness of elements in the surrounding traffic. Successful perception is a prerequisite for autonomous features such as lane departure detection, path/trajectory planning, vehicle tracking, behavior analysis, and scene understanding [5]—and a prerequisite to reach levels three to five as defined by SAE International's levels of driving automation. A wide range of sensors have been used to collect input data from the environment, but the most common approach is to rely on front-facing cameras [6]. In recent years, DNNs have demonstrated their usefulness in classifying such camera data, which in turn has enabled both perception and subsequent breakthroughs toward autonomous driving [7].

From an ISO 26262 safety assurance perspective, however, developing systems based on DNNs constitutes a major paradigm shift compared to conventional systems[1] [2]. Andrej Karpathy, Director of AI at Tesla, boldly refers to the new era as "Software 2.0."[2] No longer do human engineers explicitly describe all system behavior in source code, instead DNNs are trained using enormous amounts of historical data.

DNNs have been reported to deliver superhuman classification accuracy for specific tasks [8], but inevitably they will occasionally fail to generalize [9]. Unfortunately, from a safety perspective, analyzing when this might happen is currently not possible due to the black-box nature of DNNs. A state-of-the-art DNN might be composed of hundreds of millions of parameter weights, thus the methods for V&V of DNN components must be different compared to approaches for human readable source code. Techniques enforced by ISO 26262 such as source code reviews and exhaustive coverage testing are not applicable [3].

The contribution of this review paper is twofold. First, we describe the state-of-the-art in V&V of safety-critical systems that rely on ML. We survey academic literature, partly through a reproducible

* *Corresponding author. Email: markus.borg@ri.se*

[1]by *conventional systems* we mean any system that does not have the ability to learn or improve from experience

[2]https://medium.com/@karpathy/software-2-0-a64152b37c35

snowballing review [10], that is, establishing a body of literature by tracing referencing and referenced papers. Second, we elicit the most pressing challenges when engineering safety-critical DNN components in the automotive domain. We report from workshops with automotive experts, and we validate findings from the literature review through an industrial survey. The research has been conducted as part of SMILE[3], a joint research project between RISE AB, Volvo AB, Volvo Cars, QRTech AB, and Semcon AB.

The rest of the paper is organized as follows: Section 2 presents safety engineering concepts within the automotive domain and introduces the fundamentals of DNNs. Section 3 describes the proposed research method, including four sources of empirical evidence, and Section 4 reports our findings. Section 5 presents a synthesis targeting our two objectives, and discusses implications for research and practice. Finally, Section 6 concludes the paper and outlines the most promising directions for future work. Throughout the paper, we use the notation **[PX]** to explicitly indicate publications that are part of the snowballing literature study.

2. BACKGROUND

This section first presents development of safety-critical software according to the ISO 26262 standard [11]. Second, we introduce fundamentals of DNNs, required to understand how it could allow vehicular perception. In the remainder of this paper, we adhere to the following three definitions related to safety-critical systems:

- **Safety** is "freedom from unacceptable risk of physical injury or of damage to the health of people" [12]
- **Robustness** is "the degree to which a component can function correctly in the presence of invalid inputs or stressful environmental conditions" [13]
- **Reliability** is "the probability that a component performs its required functions for a desired period of time without failure in specified environments with a desired confidence" [14]

2.1. Safety Engineering in the Automotive Domain: ISO 26262

Safety is not a property that can be added at the end of the design. Instead, it must be an integral part of the entire engineering process. To successfully engineer a safe system, a systematic safety analysis and a methodological approach to managing risks are required [15]. Safety analysis comprises identification of hazards, development of approaches to eliminate hazards or mitigate their consequences, and verification that the approaches are in place in the system. Risk assessment is used to determine how safe a system is, and to analyze alternatives to lower the risks in the system.

Safety has always been an important concern in engineering, and best practices have often been collected in governmental or industry *safety standards*. Common standards provide a common vocabulary as well as a way for both internal and external safety assessment, that is, work tasks for both engineers working in the development organization and for independent safety assessors from certification bodies. For software-intensive systems, the generic meta-standard IEC 61508 [12] introduces the fundamentals of functional safety for electrical/electronic/programmable electronic (E/E/PE) safety-related systems, that is, hazards caused by malfunctioning E/E/PE systems rather than nonfunctional considerations such as fire, radiation, and corrosion. Several different domains have their own adaptations of IEC 61508.

ISO 26262 [11] is the automotive derivative of IEC 61508, organized into 10 parts, constituting a comprehensive safety standard covering all aspects of automotive development, production, and maintenance of safety-related systems. V&V are core activities in safety-critical development and thus discussed in detail in ISO 26262, especially in Part 4: product development at the system level and Part 6: product development at the software level. The scope of the current ISO 26262 standard is series production passenger cars with a max gross weight of 3,500 kg. However, the second edition of the standard, expected in the beginning of 2019, will broaden the scope to cover also trucks, buses, and motorcycles.

The *automotive safety lifecycle* (ASL) is one key component of ISO 26262 [16], defining fundamental concepts such as safety manager, safety plan, and confirmation measures including safety review and audit. The ASL describes six phases: management, development, production, operation, service, and decommission. Assuming that a safety-critical DNN will be considered a software unit, especially the development phase on the software level (Part 6) mandates practices that will require special treatment. Examples include verification of software implementation using inspections (Part 6:8.4.5) and conventional structural code coverage metrics (Part 6:9.4.5). It is evident that certain ISO 26262 process requirements cannot apply to ML-based software units, in line with how model-based development is currently partially excluded.

Another key component of ISO 26262 is the *automotive safety integrity level* (ASIL). In the beginning of the ASL development phase, a safety analysis of all critical functions of the system is conducted, with a focus on hazards. Then a risk analysis combining 1) the probability of exposure, 2) the driver's possible controllability, and 3) the possible severity of the outcome, results in an ASIL between A and D. ISO 26262 enforces development and verification practices corresponding to the ASIL, with the most rigorous practices required for ASIL D. Functions that are not safety-critical, that is, below ASIL A, are referred to as "QM" as no more than the normal quality management process is enforced.

2.2. Deep Learning for Perception: Approaches and Challenges

While there currently is a deep learning hype, there is no doubt that the technique has produced ground breaking results in various fields—by clever utilization of the increased processing power in the last decade, nowadays available in inexpensive GPUs, combined with the ever-increasing availability of data.

Deep learning is enabled by DNNs, which are a kind of artificial neural networks (ANNs). To some extent inspired by biological connectomes, that is, mappings of neural connections such as in the human brain, ANNs composed of connected layers of neurons are designed to learn to perform classification tasks. While ANNs have

[3]The SMILE project: Safety analysis and verification/validation of MachIne LEarning based systems

been studied for decades, significant breakthroughs came when the increased processing power allowed adding more and more layers of neurons—which also increased the number of connections between neurons by orders of magnitude. The exact number of layers, however, needed for a DNN to qualify as deep is debatable.

A major advantage of DNNs is that the classifier is less dependent on *feature engineering*, that is, using domain knowledge to (perhaps manually) identify properties in data for ML to learn from—this is often difficult. Examples of operations used to extract features in computer vision include color analysis, edge extraction, shape matching, and texture analysis. What DNNs instead introduced was an ML solution that learned those features directly from input data, greatly decreasing the need for human feature engineering. DNNs have been particularly successful in speech recognition, computer vision, and text processing—areas in which ML results were limited by the tedious work required to extract effective features.

In computer vision, essential for vehicular perception, the state-of-the-art is represented by a special class of DNNs known as *convolutional neural networks* (CNNs) [17–20]. Since 2010, several approaches based on CNNs have been proposed—and in only five years of incremental research the best CNNs matched the image classification accuracy of humans. CNN-based image recognition is now reaching the masses, as companies like Nvidia, Intel, etc. are now commercializing specialized hardware with automotive applications in mind such as the Drive PX series. Success stories in the automotive domain include lane keeping applications for self-driving cars [21,22].

Generative adversarial networks (GANs) is another approach in deep learning research that is currently receiving considerable interest [23,24]. In contrast to discriminative networks (what has been discussed so far) that learn boundaries between classes in the data for the purpose of classification, a generative network can instead be used to learn the probability of features given a specific class. Thus, a GAN could be used to generate samples from a learned network—which could possibly be used to expand available training data with additional synthetic data. GANs can also be used to generate *adversarial examples*, that is, inputs to ML classifiers intentionally created to cause misclassification.

Finally, successful applications of DNNs rely on the availability of large labeled datasets from which to learn features. In many cases, such labels are limited or does not exist at all. To maximize the utility of the labeled data, truly hard currency for anyone engineering ML-based systems, techniques such as *transfer learning* are used to adapt knowledge learned from one dataset to another domain [25].

3. RESEARCH METHOD

The overarching goal of the SMILE project is to develop approaches to V&V of ML-based systems, more specifically automotive applications relying on DNNs. Our current paper is guided by two research questions:

RQ1 What is the state-of-the-art in V&V of ML-based safety-critical systems?

RQ2 What are the main challenges when engineering safety-critical systems with DNN components in the automotive domain?

Figure 1 shows an overview of the research, divided into three sequential parts (P1–P3). Each part concluded with a Milestone (I–III). In Figure 1, tasks driven by academia (or research institutes) are presented in the light gray area—primarily addressing RQ1. Tasks in the darker gray area above, are primarily geared toward collecting data in the light of RQ2, and mostly involve industry practitioners. The darkest gray areas denote involvement of practitioners that were active in safety-critical development but not part of the SMILE project.

In the first part of the project (P1 in Figure 1), we initiated a systematic snowballing review of academic literature to map the state-of-the-art. In parallel, we organized a workshop series with domain experts from industry with monthly meetings to also assess the state-of-practice in the Swedish automotive industry. The literature review was seeded by discussions from the project definition phase (a). Later, we shared intermediate findings from the literature review at workshop #4 (b) and final results were brought up

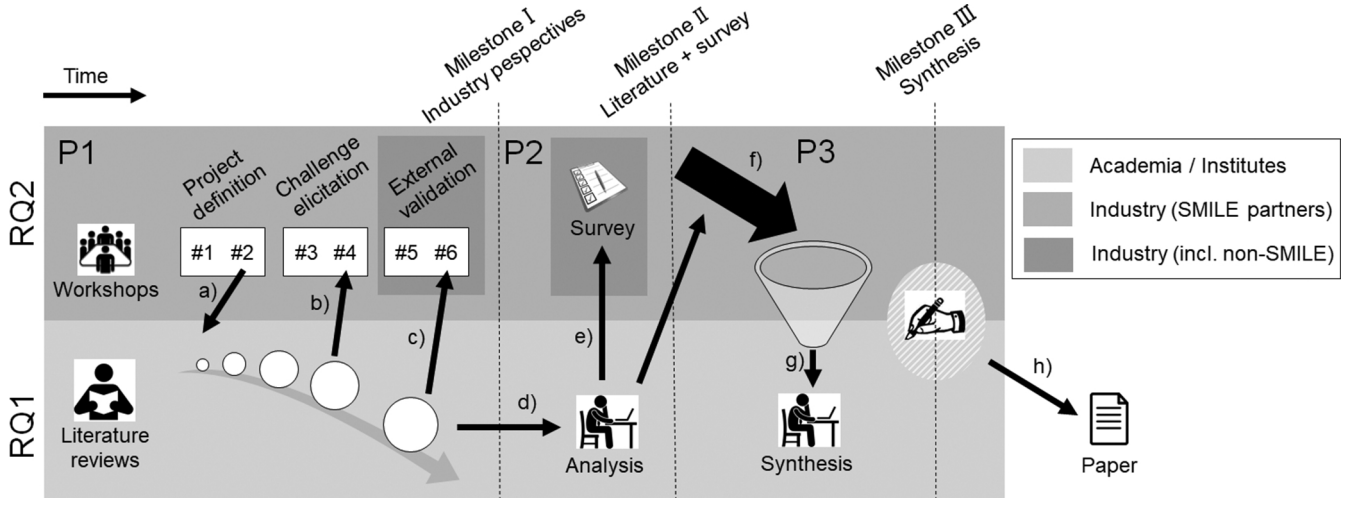

Figure 1 | Overview of the SMILE project and its three milestones. The figure illustrates the joint industry/academia nature of SMILE, indicated by light gray background for tasks driven by academia and darker gray for tasks conducted by practitioners.

to discussion at workshop #6 (c). The first part of the project concluded with Milestone I: a collection of industry perspectives.

The second part of the SMILE project (P2 in Figure 1) involved an analysis of the identified literature (d). We extracted challenges and solution proposals from the literature, and categorized them according to a structure that inductively emerged during the process (see Section 3.1). Subsequently, we created a questionnaire-based survey to validate our findings and to receive input from industry practitioners beyond SMILE (e). The second phase concluded with analyzing the survey data at Milestone II.

In the third part of the project (P3 in Figure 1), we collected all results (f), and performed a synthesis (g). Finally, writing this article concludes the research at Milestone III.

Figure 2 shows an overview of the SMILE project from an evidence perspective. The collection of empirical evidence was divided into two independent tracks resulting in four sets of evidence, reflecting the nature of the joint academia/industry project. Furthermore, the split enabled us to balance the trade-off between rigor and relevance that plagues applied research projects [26].

As shown in the upper part of Figure 2, the SMILE consortium performed (nonreplicable, from now on "*ad hoc*") searching for related work. An early set of papers was used to seed the systematic search described in the next paragraph. The findings in the body of related work (cf. A in Figure 2) were discussed at the workshops. The workshops served dual purposes, they collected empirical evidence of priorities and current needs in the Swedish automotive industry (cf. B in Figure 2), and they validated the relevance of the research identified through the *ad hoc* literature search. The upper part focused on *maximizing industrial relevance*, at the expense of rigor, that is, we are certain that the findings are relevant to the Swedish automotive industry, but the research was conducted in an *ad hoc* fashion with limited traceability and replicability. The right part of Figure 2 complements the practice-oriented research of the SMILE project by a systematic literature review, adhering to an established process [10]. The identified papers (cf. C in Figure 2) were systematized and the result was validated through a questionnaire-based survey. The survey also acted as a means to collect additional primary evidence, as we collected practitioners' opinions on V&V of ML-based systems in safety-critical domains (cf. D in Figure 2). Thus, the lower part focused on *maximizing academic rigor*.

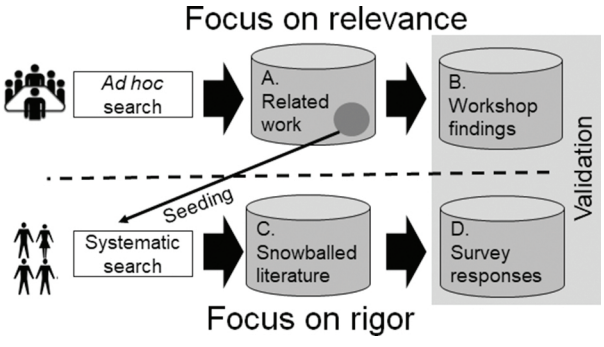

Figure 2 | Overview of the SMILE project from an evidence perspective. We treat the evidence as four different sets: A. Related work and C. Snowballed literature represent secondary evidence, whereas B. Workshop findings and D. Survey responses constitute primary evidence.

3.1. The Systematic Review

Inspired by evidence-based medicine, systematic literature reviews have become a popular software engineering research method to aggregate work in a research area. Snowballing literature reviews [10] is an alternative to more traditional database searches relying on carefully developed search strings, particularly suitable when the terminology used in the area is diverse, for example, in early stages of new research topics. This section describes the two main phases of the literature review: 1) paper selection and 2) data extraction and analysis.

3.1.1. Paper selection

As safety-critical applications of DNNs in the automotive sector is still a new research topic, we decided to broaden our literature review to encompass also other types of ML, and also to go beyond the automotive sector. We developed the following criteria: for a publication to be included in our literature review, it should describe 1) engineering of an ML-based system 2) in the context of autonomous cyber-physical systems, and 3) the paper should address V&V or safety analysis. Consequently, our criteria includes ML beyond neural networks and DNNs. Our focus on autonomous cyber-physical systems implicitly restricts our scope to safety-critical systems. Finally, we exclude papers that do not target V&V or safety analysis, but instead other engineering considerations, for example, requirements engineering, software architecture, or implementation issues.

First, we established a *start set* using exploratory searching in Google Scholar and applying our inclusion criteria. By combining various search terms related to ML, safety analysis, and V&V identified during the project definition phase of the workshop series (cf. a) in Figure 1, we identified 14 papers representing a diversity of authors, publishers, and publications venues, that is, adhering to recommendations for a feasible start set [10]. Still, the composition of the start set is a major threat to the validity of an snowballing literature review. Table 1 shows the papers in the start set.

Originating in the 14 papers in the start set, we iteratively conducted backward and forward snowballing. Backward snowballing means scanning the reference lists for additional papers to include. Forward snowballing from a paper involves adding related papers that cite the given paper. We refer to one combined effort of backward and forward snowballing as an *iteration*. In each iteration, two researchers collected candidates for inclusion and two other researchers validated the selection using the inclusion criteria. Despite our efforts to carefully process iterations, there is always a risk that relevant publications could not be identified by following references from our start set due to citation patterns in the body of scientific literature, for example, research cliques.

3.1.2. Data extraction and analysis

When the snowballing was completed, two authors extracted publication metadata according to a predefined extraction form, for example, publication venue and application domain. Second, the

Table 1 The start set and the four subsequent iterations of the snowballing literature review.

Start set	**[P1]** M. Clark *et al.* [27], **[P2]** D. Amodei *et al.* [28], **[P3]** G. Brat and A. Jonsson [29], **[P4]** A. Broggi *et al.* [30], **[P5]** B. Taylor *et al.* [31], **[P6]** J. Taylor *et al.* [32], **[P7]** A. Carvalho *et al.* [33], **[P8]** S. Ramos *et al.* [1], **[P9]** R. Alexander *et al.* [34], **[P10]** X. Zou *et al.* [35], **[P11]** X. Zou xtit *et al.* [36], **[P12]** J. Arnold and R. Alexander [37], **[P13]** S. Sivaraman and M. Trivedi [38], **[P14]** A. Mozaffari *et al.* [39]
Iteration 1	**[P15]** S. Seshia *et al.* [40], **[P16]** P. Helle *et al.* [41], **[P17]** L. Li *et al.*, [42], **[P18]** W. Shi *et al.* [43], **[P19]** K. Sullivan *et al.* [44], **[P20]** R. Broderick [45], **[P21]** N. Li *et al.* [46], **[P22]** S. Russell *et al.* [47], **[P23]** A. Broggi *et al.* [48], **[P24]** J. Schumann and S. Nelson [49], **[P25]** J. Hull *et al.* [50], **[P26]** L. Pulina and A. Tacchella [51], **[P27]** S. Lefevre *et al.* [52]
Iteration 2	**[P28]** X. Huang *et al.* [53], **[P29]** K. Sullivan *et al.* [44], **[P30]** P. Gupta P and J. Schumann [54], **[P31]** J. Schumann *et al.* [55], **[P32]** R. Broderick [56], **[P33]** Y. Liu *et al.* [57], **[P34]** S. Yerramalla *et al.* [58], **[P35]** R. Zakrzewski [59], **[P36]** S. Yerramalla tit *et al.* [58], **[P37]** G. Katz *et al.* [4], **[P38]** A. Akametalu *et al.* [60], **[P39]** S. Seshia *et al.* [61], **[P40]** A. Mili *et al.* [62], **[P41]** Z. Kurd *et al.* [63], **[P42]** L. Pulina and A. Tacchella [64], **[P43]** J. Schumann *et al.* [49], **[P44]** R. Zakrzewski [65], **[P45]** D. Mackall *et al.* [66]
Iteration 3	**[P46]** S. Jacklin *et al.* [67], **[P47]** N. Nguyen and S. Jacklin [68], **[P48]** J. Schumann and Y. Liu [69], **[P49]** N. Nguyen and S. Jacklin [70], **[P50]** S. Jacklin *et al.* [71], **[P51]** J. Taylor *et al.* [32], **[P52]** G. Li *et al.* [72], **[P53]** P. Gupta *et al.* [73], **[P54]** K. Scheibler *et al.* [74], **[P55]** P. Gupta *et al.* [75], **[P56]** S. Jacklin *et al.* [76], **[P57]** V. Cortellessa *et al.* [77], **[P58]** S. Yerramalla *et al.* [78]
Iteration 4	**[P58]** S. Jacklin, [79], **[P59]** F. Soares *et al.* [80], **[P60]** X. Zhang *et al.* [81], **[P61]** F. Soares and J. Burken [82], **[P62]** C. Torens *et al.* [83], **[P63]** J. Bosworth and P. Williams-Hayes [84], **[P64]** R. Zakrzewski [85]

same two authors conducted an assessment of rigor and relevance as recommended by Ivarsson and Gorschek [26]. Third, they addressed RQ1 using thematic analysis [86], that is, summarizing, integrating, combining, and comparing findings of primary studies to identify patterns.

Our initial plan was to classify challenges and solution proposals in previous work using classification schemes developed by Amodei *et al.* [**P2**] and Varshney [87], respectively. However, neither of the two proposed categorization schemes were successful in spanning the content of the selected papers. To better characterize the selected body of research, we inductively created new classification schemes for challenges and solution proposals according to a grounded theory approach. Table 2 defines the final categories used in our study, seven challenge categories and five solution proposal categories.

3.2. The Questionnaire-Based Survey

To validate the findings from the snowballed literature (cf. C. in Figure 2), we designed a web-based questionnaire to survey practitioners in safety-critical domains. Furthermore, reaching out to additional practitioners beyond the SMILE project enables us to collect more insights into challenges related to ML-based systems in additional safety-critical contexts (cf. D. in Figure 2). Moreover, we used the survey to let the practitioners rate the importance of the challenges reported in the academic literature, as well as the perceived feasibility of the published solutions proposals.

We designed the survey instrument using Google Forms, structured as 10 questions organized into two sections. The first section consisted of seven closed-end questions related to demographics of the respondents and their organizations and three Likert items concerning high-level statements on V&V of ML-based systems. The second section consisted of three questions: 1) rating the importance of the challenge categories, 2) rating how promising the solution proposal categories are, and 3) an open-end free-text answer requesting a comment on our main findings and possibly adding missing aspects.

We opted for an inclusive approach and used convenience sampling to collect responses [88], that is, a nonprobabilistic sampling method. The target population was software and systems engineering practitioners working in safety-critical contexts, including both engineering and managerial roles, for example, test managers, developers, architects, safety engineers, and product managers. The main recruitment strategy was to invite the extended SMILE network (cf. workshops #5 and #6 in Figure 1) and to advertise

Table 2 Definition of categories of challenges and solution proposals for V&V of ML-based systems.

Challenge Categories	**Definitions**
State-space explosion	Challenges related to the very large size of the input space.
Robustness	Issues related to operation in the presence of invalid inputs or stressful environmental conditions.
Systems engineering	Challenges related to integration or co-engineering of ML-based and conventional components.
Transparency	Challenges originating in the black-box nature of the ML system.
Requirements specification	Problems related to specifying expectations on the learning behavior.
Test specification	Issues related to designing test cases for ML-based systems, e.g., nondeterministic output.
Adversarial attacks	Threats related to antagonistic attacks on ML-based systems, e.g., adversarial examples.
Solution Proposal Categories	**Definitions**
Formal methods	Approaches to mathematically prove that some specification holds.
Control theory	Verification of learning behavior based on automatic control and self-adaptive systems.
Probabilistic methods	Statistical approaches such as uncertainty calculation, Bayesian analysis, and confidence intervals.
Test case design	Approaches to create effective test cases, e.g., using genetic algorithms or procedural generation.
Process guidelines	Guidelines supporting work processes, e.g., covering training data collection or testing strategies.

V&V, verification and validation; ML, machine learning.

the survey invitation in LinkedIn groups related to development of safety-critical systems. We collected answers in 2017, from July 1 to August 31.

As a first step of the response analysis, we performed a content sanity check to identify invalid answers, for example, nonsense or careless responses. Subsequently, we collected summary statistics of the responses and visualized it with bar charts to get a quick overview of the data. We calculated Spearman rank correlation (ρ) between all ordinal scale responses, interpreting correlations as weak, moderate, and strong for $\rho > 0.3$, $\rho > 0.5$, and $\rho > 0.7$, respectively. Finally, the two open-ended questions were coded, summarized, and validated by four of the coauthors.

4. RESULTS AND DISCUSSION

This section is organized according to the evidence perspective provided in Figure 2: A. Related work, B. Workshop findings, C. Snowballed literature, and D) Survey responses. As reported in Section 3, A. and B. focus on industrial relevance, whereas C. and D. aim at academic rigor.

4.1. Related Work

The related work section (cf. A. in Figure 2) presents an overview of literature that was identified during the SMILE project. Fourteen of the papers were selected early to seed the (independent) snowballing literature review described in Section 3.1. In this section, we first describe the start set **[P1]**–**[P14]**, and then papers that were subsequently identified by SMILE members or the anonymous reviewers of the manuscript—but not through the snowballing process (as these are reported separately in Section 4.3).

4.1.1. The snowballing start set

The following 14 papers were selected as the snowballing start set, representing a diverse set of authors, publication venues, and publication years. We briefly describe them below, and motivate their inclusion in the start set.

[P1] Clark *et al.* reported from a US Air Force research project on challenges in V&V of autonomous systems. This work is highly related to the SMILE project.

[P2] Amodei *et al.* listed five challenges to artificial intelligence (AI) safety according to Google Brain: 1) avoiding negative side effects, 2) avoiding reward hacking, 3) scalable oversight, 4) safe exploration, and 5) robustness to distributional shift.

[P3] Brat and Jonsson discussed challenges in V&V of autonomous systems engineered for space exploration. Included to cover the space domain.

[P4] Broggi *et al.* presented extensive testing of the BRAiVE autonomous vehicle prototype by driving from Italy to China. Included as it is different, that is, reporting experiences from a practical trip.

[P5] Taylor *et al.* sampled research in progress (in 2003) on V&V of neural networks, aimed at NASA applications. Included to snowball research conducted in the beginning of the millennium.

[P6] Taylor *et al.* with the Machine Intelligence Research Institute surveyed design principles that could ensure that systems behave in line with the interests of their operators—which they refer to as "AI alignment." Included to bring in a more philosophical perspective on safety.

[P7] Carvalho *et al.* presented a decade of research on control design methods for systematic handling of uncertain forecasts for autonomous vehicles. Included to cover robotics.

[P8] Ramos *et al.* proposed a DNN-based obstacle detection framework, providing sensor fusion for detection of small road hazards. Included as the work closely resembles the use case discussed at the workshops (see Section 4.2).

[P9] Alexander *et al.* suggested "situation coverage methods" for autonomous robots to support testing of all environmental circumstances. Included to cover coverage.

[P10] Zou *et al.* discussed safety assessments of probabilistic airborne collision avoidance systems and proposes a genetic algorithm to search for undesired situations. Included to cover probabilistic approaches.

[P11] Zou *et al.* presented a safety validation approach for avoidance systems in unmanned aerial vehicles, using evolutionary search to guide simulations to potential conflict situations in large state spaces. Although the authorsoverlap, included to snowball research on simulation.

[P12] Arnold and Alexander proposed using procedural content generation to create challenging environmental situations when testing autonomous robot control algorithms in simulations. Included to cover synthetic test data.

[P13] Sivaraman and Trivedi compared three active learning approaches for on-road vehicle detection. Included to add a semi-supervised ML approach.

[P14] Mozaffari *et al.* developed a robust safety-oriented autonomous cruise controller based on the model predictive control technique. Included to identify approaches based on control theory.

In the start set, we consider **[P1]** to be the research endeavor closest to our current study. While we target the automotive domain rather than aerospace, both studies address highly similar research objectives—and also the method used to explore the topic is close to our approach. **[P1]** describes a year-long study aimed at 1) understanding the unique challenges to the certification of safety-critical autonomous systems and 2) identifying the V&V approaches needed to overcome them. To accomplish this, the US Air Force organized three workshops with representatives from industry, academia, and governmental agencies, respectively. **[P1]** concludes that that there are four enduring problems that must be addressed:

- State-Space Explosion—In an autonomous system, the decision space is nondeterministic and the system might be continuously learning. Thus, over time, there may be several output signals for each input signal. This in turn makes it inherently challenging to exhaustively search, examine, and test the entire decision space.
- Unpredictable Environments—Conventional systems have limited ability to adapt to unanticipated events, but an autonomous systems should respond to situations that were not programmed at design time. However, there is a trade-off

between performance and correct behavior, which exacerbates the state-space explosion problem.

- Emergent Behavior—Nondeterministic and adaptive systems may induce behavior that result in unintended consequences. Challenges comprise how to understand all intended and unintended behavior and how to design experiments and test vectors that are applicable to adaptive decision-making in an unpredictable environment.
- Human-Machine Communication—Hand-off, communication, and cooperation between the operator and the autonomous system play an important role to create mutual trust between the human and the system. It is not known how to address these issues when the behavior is not known at design time.

With these enduring challenges in mind, **[P1]** calls for research to pursue five goals in future technology development. First, approaches to *cumulatively build safety evidence* through the phases of Research & Development (R&D), Test & Evaluation (T&E), and Operational Tests. The US Air Force calls for effective methods to reuse safety evidence throughout the entire product development lifecycle. Second, **[P1]** argues that *formal methods*, embedded during R&D, could provide safety assurance. This approach could reduce the need for T&E and operational tests. Third, novel techniques to *specify requirements based on formalism, mathematics, and rigorous natural language* could bring clarity and allow automatic testcase generation and automated traceability to low-level designs. Fourth, *run-time decision assurance* may allow restraining the behavior of the system, thus shifting focus from off-line verification to instead performing online testing at run-time. Fifth, **[P1]** calls for research on *compositional case generation*, that is, better approaches to combine different pieces of evidence into one compelling safety case.

4.1.2. Non-snowballed-related work

This subsection reports the related work that stirred up the most interesting discussions in the SMILE project. In contrast to the snowballing literature review, we do not provide steps to replicate the identification of the following papers:

Knauss *et al.* conducted an exploratory interview study to elicit challenges when engineering autonomous cars [89]. Based on interviews and focus groups with 26 domain experts in five countries, the authors report in particular challenges in testing automated vehicles. Major challenges are related to 1) virtual testing and simulation, 2) safety, reliability, and quality, 3) sensors and their models 4) complexity of, and amount of, test cases, and 5) hand-off between driver and vehicle.

Spanfelner *et al.* conducted research on safety and autonomy in the ISO 26262 context [9]. Their conclusion is that driver assistance systems need models to be able to interpret the surrounding environment, that is, to enable vehicular perception. Since models, by definition, are simplifications of the real world, they will be subject to functional insufficiencies. By accepting that such insufficiencies may fail to reach the functional safety goals, it is possible to design additional measures that in turn can meet the safety goals.

Heckemann *et al.* identified two primary challenges in developing autonomous vehicles adhering to ISO 26262 [90]. First, the driver is today considered to be part of the safety concept, but future vehicles will make driving maneuvers without interventions by a human driver. Second, the system complexity of modern vehicle systems is continuously growing as new functionality is added. This obstructs safety assessment, as increased complexity makes it harder to verify freedom of faults.

Varshney discussed concepts related to engineering safety for ML systems from the perspective of minimizing risk and epistemic uncertainty [87], that is, uncertainty due to gaps in knowledge as opposed to intrinsic variability in the products. More specifically, he analyzed how four general strategies for promoting safety [91] apply to systems with ML components. First, *inherently safe design* means excluding a potential hazard from the system instead of controlling it. A prerequisite for assuring such a design is to improve the interpretability of the typically opaque ML models. Second, *safety reserves* means the factor of safety, that is, the ratio of absolute structural capacity to actual applied load in structural engineering. In ML, interpretations include a focus on a the maximum error of classifiers instead of the average error, or training models to be robust to adversarial examples. Third, *safe fail* implies that a system remains safe even when it fails in its intended operation, traditionally by relying on constructs such as electrical fuses and safety valves. In ML, a concept of run-time monitoring must be accomplished, for example, by continuously monitoring how certain a DNN model is performingin its classification task. Fourth, *procedural safeguards* covers any safety measures that are not designed into the system, for example, mandatory safety audits, training of personnel, and user manuals describing how to define the training set.

Seshia *et al.* identified five major challenges to achieve formally-verified AI-based systems [40]. First, a methodology to provide a model of the environment even in the presence of uncertainty. Second, a precise mathematical formulation of what the system is supposed to do, i.e., a formal specification. Third, the need to come up with new techniques to formally model the different components that will use ML. Fourth, systematically generating training and testing data for ML-based components. Finally, developing computationally scalable engines that are able to verify quantitatively the requirements of a system.

One approach to tackle the opaqueness of DNNs is to use visualization. Bojarski *et al.* [92] developed a tool for visualizing the parts of an image that are used for decision-making in vehicular perception. Their tool demonstrated an end-to-end driving application where the input is images and the output is the steering angle. Mhamdi *et al.* also studied the black-box aspects of neural networks, and show that the robustness of a complete DNN can be assessed by an analysis focused on individual neurons as units of failure [93]—a much more reasonable approach given the state-space explosion.

In a paper on ensemble learning, Varshney *et al.* describes a reject option for classifiers [94]. Such a classifier could, instead of presenting a highly uncertain classification, request that a human operator must intervene. A common assumption is that the classifier is the least confident in the vicinity of the decision boundary, that is, that there is an inverse relationship between distance and confidence. While this might be true in some parts of the feature space, it is not a reliable measure in parts that contain too few training examples.

For a reject option to provide a "safe fail" strategy, it must trigger both 1) near the decision boundary in parts of the feature space with many training examples and 2) in any decision represented by too few training examples.

Heckemann *et al.* proposed using the concept of *adaptive safety cage architectures* to support future autonomy in the automotive domain [90], that is, an independent safety mechanism that continuously monitors sensor input. The authors separated two areas of operation: a valid area (that is considered safe) and an invalid area that can lead to hazardous situations. If the function is about to enter the invalid area, the safety cage will invoke an appropriate safe action, such as a minimum risk emergency stopping maneuver or a graceful degradation. Heckemann *et al.* argued that a safety cage can be used in an ASIL decomposition by acting as a functionally redundant system to the actual control system. The highly complex control function could then be developed according to the quality management standard, whereas the comparably simple safety cage could adhere to a higher ASIL level.

Adler *et al.* presented a similar run-time monitoring mechanism for detecting malfunctions, referred to as a *safety supervisor* [95]. Their safety supervisor is part of an overall safety approach for autonomous vehicles, consisting of a structured four-step method to identify the most critical combinations of behaviors and situations. Once the critical combinations have been specified, the authors propose implementing tailored safety supervisors to safeguard against related malfunctions.

Finally, a technical report prepared by Bhattacharyya *et al.* for the NASA Langley Research Center discussed certification considerations of adaptive systems in the aerospace domain [96]. The report separates adaptive control algorithms and AI algorithms, and the latter is closely related to our study since it covers ML and ANN. Their certification challenges for adaptive systems are organized in four categories:

- Comprehensive requirements—Specifying a set of requirements that completely describe the behavior, as mandated by current safety standards, is presented as the most difficult challenge to tackle.
- Verifiable requirements—Specifying pass criteria for test cases at design-time might be hard. Also, current aerospace V&V relies heavily on coverage testing of source code in imperative languages, but how to interpret that for AI algorithms is unclear.
- Documented design—Certification requires detailed documentation, but components realizing adaptive algorithms were rarely developed with this in mind. Especially AI algorithms are often distributively developed by open source communities, which makes it hard to reverse engineer documentation and traceability.
- Transparent design—Regulators expect a transparent design and a conventional implementation to be presented for evaluation. Increasing system complexity by introducing novel adaptive algorithms challenges comprehensibility and trust. On top of that, adaptive systems are often nondeterministic, which makes it harder to demonstrate absence of unintended functionality.

4.2. The Workshop Series

During the six workshops with industry partners (cf. #1–#6 in Figure 1), we discussed key questions that must be explored to enable engineering of safety-critical automotive systems with DNNs. Three subareas emerged during the workshops: 1) robustness, 2) interplay between DNN components and conventional software, and 3) V&V of DNN components.

4.2.1. Robustness of DNN components

The concept of robustness permeated most discussions during the workshops. While robustness is technically well defined, in the workshops it often remained a rather elusive quality attribute—typically translated to "something you can trust."

To bring the workshop participants to the same page, we found it useful to base the discussions on a simple ML case: a confusion matrix for a one-class classifier for camera-based animal detection. For each input image, the result of the classifier is limited to one of the four options: 1) an animal is present and correctly classified (true positive), 2) no animal is present and the classifier does not signal animal detection (true negative), 3) the classifier reports animal presence, but there is none (false positive), and 4) an animal is present, but the classifier misses it (false negative).

For the classifier to be considered robust, the participants stressed the importance of not generating false positives and false negatives despite occasional low quality input or changes in the environmental conditions, for example, dusk, rain, or sun glare. A robust ML system should neither miss present animals, risking collisions, nor suggest emergency braking that risk rear-end collisions. As the importance of robustness in the example is obvious, we see a need for future research both on how to specify and verify acceptable levels of ML robustness.

During the workshops, we also discussed more technical aspects engineering robust DNN components. First, our industry practitioners brought up the issue of DNN architectures to be problem-specific. While there are some approaches to automatically generating neural network architectures [97,98], typically designing the DNN architecture is an *ad hoc* process of trial and error. Often a well-known architecture is used as a baseline and then it is tuned to fit the problem at hand. Our workshops recognized the challenge of engineering robust DNN-based systems, in part due to their highly problem-specific architectures.

Second, once the DNN architecture is set, training commences to assign weights to the trainable parameters of the network. The selection of training data must be representative for the task, in our discussions animal detection, and for the environment that the system will operate in. The workshops agreed that robustness of DNN components can never be achieved without careful selection of training data. Not only must the amount and quality of sensors (in our case cameras) acquiring the different stimuli for the training data be sufficient, also other factors such as positioning, orientation, aperture, and even geographical location like city and country must match the animal detection example. At the workshops, we emphasized the issue of camera positions as both car and truck manufacturers were part of SMILE—to what extent can training data from a car's

perspective be reused for a truck? Or should a truck rather benefit from its size and collect dedicated training data from its elevated camera position?

Third, also related to training data, the workshops discussed working with synthetic data. While such data always can be used to complement training data, there are several open questions on how to best come up with the best mix during the training stage. As reported in Section 2.2, GANs [23,24] could be a good tool for synthesizing data. Sixt *et al.* [99] proposed a framework called RenderGAN that could generate large amounts of realistic labeled data for training. In transfer learning, training efficiency improves by combining data from different data sets [25,100]. One possible approach could be to first train the DNN component using synthetic data from, for example, simulators like TORCS[4], then data from some publicly available database could be used to continue the training, for example, the KITTI data[5] or CityScape[6], and finally, data from the geographical region where the vehicle should operate could be added. For any attempts at transfer learning, the workshops identified the need to measure to what extent training data matches the planned operational environment.

4.2.2. Complementing DNNs with conventional components

During the workshops, we repeatedly reminded the participants to consider DNNs from a systems perspective. DNN components will always be part of an automotive system consisting of also conventional hardware and software components.

Several researchers claim that that DNN components is a prerequisite for autonomous driving [2,22,101]. However, how to integrate such components in a system is an open question. Safety is a systems issue, rather than a component specific issue. All hazards introduced by both DNNs and conventional software must be analyzed within the context of systems engineering principles. On the other hand, the hazards can also be addressed on a system level.

One approach to achieve DNN safety is to introduce complementary components, that is, when a DNN model fails to generalize, a conventional software or hardware component might step in to maintain safe operation. During the workshops, particular attention was given to introducing a *safety cage concept*. Our discussions orbited a solution in which the DNN component was encapsulated by a supervisor, or a safety cage, that continuously monitors the input to the DNN component. The envisioned safety cage should perform novelty detection [102] and alert when input does not belong within the training region of the DNN component, that is, if the risk of failed generalization was too high, the safety cage should redirect the execution to a *safe-track*. The safe-track should then operate without any ML components involved, enabling traditional approaches to safety-critical software engineering.

The concept of an ML safety cage is in line with Varshney's discussions of "safe fail" [87]. Different options to implement an ML safety cage include adaptations of fail-silent systems [103], plausibility checks [104], and arbitration. However, Adler *et al.* [95] indicated that the *no free lunch* theorem might apply for safety cages, by stating that if tailored safety safety cages are to be developed to safeguard against domain-specific malfunctions, thus, different safety cages may be required for different systems.

Introducing redundancy in the ML system is an approach related to the safe track. One method is to use ensemble methods in computer vision applications [105], that is, employing multiple learning algorithms to improve predictive performance. Redundancy can also be introduced in an ML-based system using hardware component, for example, using an array of sensors of the same, or different, kind. Increasing the amount of input data should increase the probability of finding patterns closer to the training data set. Combining data from various input sources, referred to as sensor fusion, also helps overcoming the potential deficiencies of individual sensors.

4.2.3. V&V approaches for systems with DNN components

Developing approaches to engineer robust systems with DNN components is not enough, the automotive industry must also develop novel approaches to V&V. V&V is a cornerstone in safety certification, but it still remains unclear how to develop a safety case around applications with DNNs.

As pointed out in previous work, the current ISO 26262 standard is not applicable when developing autonomous systems that rely on DNNs [90]. Our workshops corroborate this view, by identifying several open questions that need to be better understood:

- How is a DNN component classified in ISO 26262? Should it be regarded as an individual software unit or a component?
- From a safety perspective, is it possible to treat DNN misclassifications as "hardware failures"? If yes, are the hardware failure target values defined in ISO 26262 applicable?
- ISO 26262 mandates complete test coverage of the software, but what does this imply for a DNN? What is sufficient coverage for a DNN?
- What metrics should be used to specify the DNN accuracy? Should quality targets using such metrics be used in the DNN requirements specifications, and subsequently as targets for verification activities?

Apart from the open questions, our workshop participants identified several aspects that would support V&V. First, as requirements engineering is fundamental to high-quality V&V [106], some workshop participants requested a formal, or semiformal, notation for requirements related to functional safety in the DNN context. Defining low-level requirements that would be verifiable appears to be one of the greatest challenges in this area. Second, there is a need for a tool-chain and framework tailored to lifecycle management of systems with DNN components—current solutions tailored for human-readable source code are not feasible and must be complemented with too many immature internal tools. Third, methods for test case generation for DNN will be critical, as manual creation of test data does not scale.

[4] http://torcs.sourceforge.net

[5] http://www.cvlibs.net/datasets/kitti/

[6] https://www.cityscapes-dataset.com/

Table 3 | Distribution of challenge and solution proposal categories.

Challenge Category	#	Paper IDs
State-space explosion	6	[P3], [P15], [P16], [P47]
Robustness	4	[P1], [P2], [P15], [P55]
Systems engineering	2	[P1], [P55]
Transparency	2	[P1], [P55]
Requirements specification	3	[P15], [P55]
Test specification	3	[P16], [P46], [P55]
Adversarial attacks	1	[P15]
Solution Proposal Category	**#**	**Paper IDs**
Formal methods	8	[P3], [P26], [P42], [P28], [P37], P[40], [P44], [P53]
Control theory	7	[P7], [P20], [P25], [P64], [P36], [P47], [P57], [P60]
Probabilistic methods	7	[P18], [P30], [P31], [P32], [P33], [P35], [P50], [P52], [P54]
Test case design	5	[P9], [P10], [P12], [P17], [P21]
Process guidelines	4	[P23], [P51], [P56], [P59]

Finally, a major theme during the workshops was how to best use simulation as a means to support V&V. We believe that the future will require massive use of simulation to ensure safe DNN components. Consequently, there is a need to develop simulation strategies to cover both normal circumstances as well as rare, but dangerous, traffic situations. Furthermore, simulation might also be used to assess the sensitivity to adversarial examples.

4.3. The Systematic Snowballing

Table 1 shows the results from the five iterations of the snowballing. In total, the snowballing procedure identified 64 papers including the start set. We notice two publication peaks: 29 papers were published between 2002 and 2007 and 25 papers were published between 2013 and 2016. The former set of papers were dominated by research on using neural networks for adaptive flight controllers, whereas the latter set predominantly addresses the automotive domain. This finding suggests that organizations currently developing ML-based systems for self-driving cars could learn from similar endeavors in the aerospace domain roughly a decade ago—while DNN was not available then, several aspects of V&V enforced by aerospace safety standards are similar to ISO 26262. Note, however, that 19 of the papers do not target any specific domain, but rather discusses ML-based systems in general.

Table 3 shows the distribution of challenge and solution proposal categories identified in the papers; "#" indicates the number of unique challenges or solution proposals matching a specific category. As each paper can report more than one challenge or solution proposal, and the same challenge or solution proposal can occur in more than one paper, the number of paper IDs in the third column does not necessarily match the "#." The challenges most frequently mentioned in the papers relate to state-space explosion and robustness, whereas the most commonly proposed solutions constitute approaches that belong to formal methods, control theory, or probabilistic methods.

Regarding the publication years, we notice that the discussion on state-space explosion primarily has been active in recent years, possibly explained by the increasing application of DNNs. Looking at solution proposals, we see that probabilistic methods was particularly popular during the first publication peak, and that research specifically addressing test case design for ML-based systems has appeared first after 2012.

Figure 4 shows a mapping between solution proposals categories and challenge categories. Some of the papers propose a solution to address challenges belonging to a specific category. For each such instance, we connect solution proposals (to the left) and challenges (to the right), that is, the width of the connection illustrates the number of instances. Note that we did put the solution proposal in **[P4]** (deployment in real operational setting) in its own "Other" category. None of the proposed solutions address challenges related to the categories "Requirements specification" or "Systems engineering," indicating a research gap. Furthermore, "Transparency" is the challenge category that has been addressed the most in the papers, followed by "State-space explosion."

Two books summarize most findings from the aerospace domain identified through our systematic snowballing. Taylor edited a book in 2006 that collected experiences for V&V of ANN technology [107] in a project sponsored by the NASA Goddard Space Flight Center. Taylor concluded that the V&V techniques available at the time must evolve to tackle ANNs. Taylor's book reports five areas that need to be augmented to allow V&V of ANN-based systems:[7]

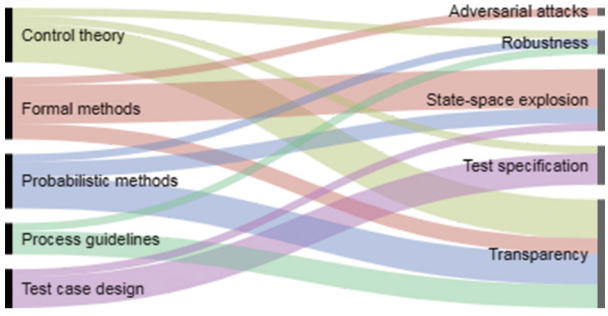

Figure 4 | Mapping between categories of solution proposals (to the left) and challenges (to the right).

[7]The best practices were also laterdistilled into a guidance document intended for practitioners [73]

- *Configuration management* must track all additional design elements, for example, the training data, the network architecture, and the learning algorithms. Any V&V activity must carefully specify the configuration under test.
- *Requirements* need to specify novel adaptive behavior, including control requirements (how to acquire and act on knowledge) and knowledge requirements (what knowledge should be acquired).
- *Design specifications* must capture design choices related to novel design elements such as training data, network architecture, and activation functions. V&V of the ANN design should ensure that the choices are appropriate.
- *Development lifecycles* for ANNs are highly iterative and last until some quantitative goal has been reached. Traditional waterfall software development is not feasible, and V&V must be an integral part rather than an add-on.
- *Testing* needs to evolve to address novel requirements. Structure testing should determine whether the network architecture is better at learning according to the control requirements than alternative architectures. Knowledge testing should verify that the ANN has learned what was specified in the knowledge requirements.

The second book that has collected experiences on V&V of (mostly aerospace) ANNs, also funded by NASA, was edited by Schumann and Liu and published in 2010 [108]. While the book primarily surveys the use of ANNs in high-assurance systems, parts of the discussion is focused on V&V—and the overall conclusion that V&V must evolve to handle ANNs is corroborated. In contrast to the organization we report in Table 3, the book suggests grouping solution proposals into approaches that 1) separate ANN algorithms from conventional source code, 2) analyze the network architecture, 3) consider ANNs as function approximators, 4) tackle the opaqueness of ANNs, 5) assess the characteristics of the learning algorithm, 6) analyze the selection and quality of training data, and 7) provides means for online monitoring of ANNs. We believe that our organization is largely orthogonal to the list above, thus both could be used in a complementary fashion.

4.4. The Survey

This section organizes the findings from the survey into closed questions, correlation analysis, and open questions, respectively.

4.4.1. Closed questions

Forty-nine practitioners answered our survey, most of them primarily working in Europe (38 out of 49, 77.6%). Twenty respondents (40.8%) work primarily in the automotive domain, followed by 14 in aerospace (28.6%). Other represented domains include process industry (5 respondents), railway (5 respondents), and government/military (3 respondents). The respondents represent a variety of roles, from system architects (17 out of 49, 34.7%) to product developers (10 out of 49, 20.4%), and managerial roles (7 out of 49, 14.3%). Most respondents primarily work in Europe (38 out of 49, 77.6%) or North America (7 out of 49, 14.3%).

Most respondents have some proficiency in ML. Twenty-five respondents (51.0%) report having fundamental awareness of ML concepts and practical ML concerns. Sixteen respondents (32.7%) have higher proficiency, that is, can implement ML solutions independently or with guidance—but no respondents consider themselves ML experts. On the other side of the spectrum, eight respondents report possessing no ML knowledge.

We used three Likert items to assess the respondents' general thoughts about ML and functional safety, reported as a) to c) in Figure 4. Most respondents agree (or strongly agree) that applying ML in safety-critical applications will be important in their organizations in the future (29 out of 49, 59.2%), whereas eight (16.3%) disagree. At the same time, 29 out of 49 (59.2%) of the respondents report that V&V of ML-based features is considered particularly difficult by their organizations—20 respondents even strongly agrees with the statement. It is clear to our respondents that more attention is needed regarding V&V of ML-based systems, as only 10 out of 49 (20.4%) believe that their organizations are well prepared for the emerging paradigm.

Robustness (cf. e) in Figure 4 stands out as the particularly important challenge, reported as "extremely important" by 29 out of 49 (59.2%). However, all challenges covered in the questionnaire were considered important by the respondents. The only challenge that appears less urgent to the respondents is adversarial attacks, but the difference is minor.

The respondents consider simulated test cases as the most promising solution proposal to tackle challenges in V&V of ML-based systems, reported as extremely promising by 18 out of 49 respondents (36.7%) and moderately promising by 12 respondents (24.5%). Probabilistic methods is the least promising solution proposal according to the respondents, followed by process guidelines.

4.4.2. Correlation analysis

We identified some noteworthy correlations in the responses. The respondents' ML proficiency (Q4) is moderately correlated ($\rho = 0.53$) with the perception of ML importance (Q5)—an expected finding as respondents with a personal investment are likely to be biased. More interestingly, we found that ML proficiency was also moderately correlated to two of the seven challenge categories: transparency ($\rho = 0.61$) and state-space explosion ($\rho = 0.54$). This suggests that these two challenges are particularly difficult to comprehend for nonexperts. Perceiving the organization as well prepared for introducing ML-based solutions (Q4) is moderately correlated ($\rho = 0.57$) with considering systems engineering challenges (Q7) as particularly important and weakly correlated regarding process guidelines (Q16) as a promising solution ($\rho = 0.37$). As these are the only correlations with Q4, it indicates that organizations that have reached a certain ML maturity have progressed beyond specific issues and instead focus on the bigger picture, that is, how to incorporate ML in systems and how to adapt internal processes in the new ML era.

There are more correlations within the categories of challenges (Q5–Q11) and solution proposals (Q12–Q16) than between the two groups. The only strong correlation between groups is test specification (Q11) and formal methods (Q12) ($\rho = 0.71$). Within the challenges, the correlation between the two challenges state-space

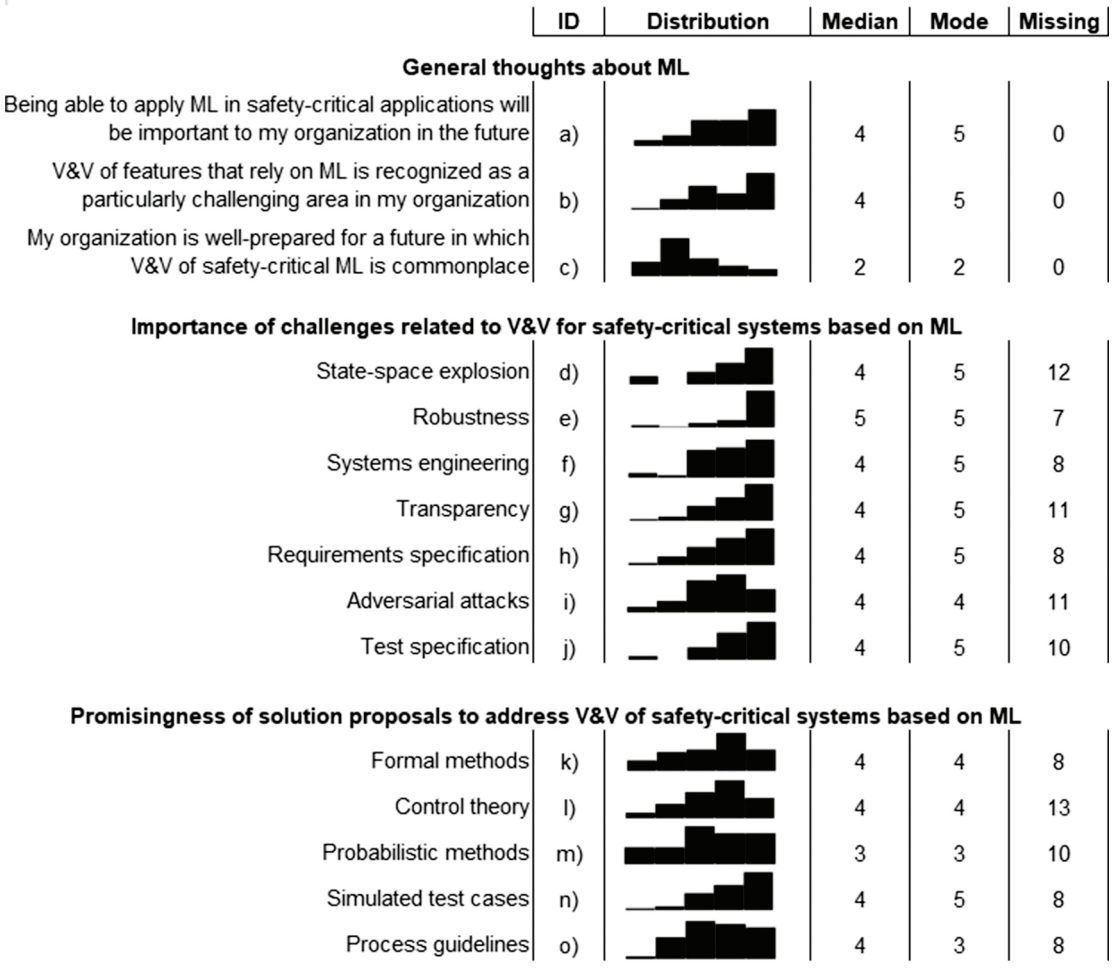

Figure 4 Answers to the closed questions of the survey. a)–c) show three Likert items, ranging from strongly disagree (1) to strongly agree (5). d)–o) Reports on importance/promisingness using the following ordinal scale: not at all, slightly, somewhat, moderately, and extremely. The "Missing" column includes both "I don't know" answers and missing answers.

explosion (Q5) and transparency (Q8) stands out as particularly strong ($\rho = 0.91$), illustrating the close connection between these two issues with large DNN architectures. Also the two challenge categories requirements specifications (Q9) and test specifications (Q11) are strongly correlated ($\rho = 0.71$), in line with a large body of previous work on aligning the two concepts [106].

4.4.3. Open questions

The end of the questionnaire contained an open-ended question (Q17), requesting a comment on Figure 4 and the accompanying findings: "although few individual V&V challenges related to ML transparency are highlighted in the literature, it is the challenge most often addressed by the previous publications' solution proposals. We also find that the second most addressed challenge in previous work is related to state-space explosion."

Sixteen out of 49 respondents (32.7%) provided a free text answer to Q17, representing highly contrasting viewpoints. Eight respondents reported that the findings were not in line with their expectations, whether seven respondents agreed—one respondent neither agreed nor disagreed. Examples of more important challenges emphasized by the respondents include both other listed challenges, that is, robustness and requirements specification, and other challenges, for example, uncertainty of sensor data (in automotive) and the knowledge gap between industry and regulatory bodies (in the process industry). Three respondents answer in general terms that the main challenge of ML-based systems is the intrinsic nondeterminism.

On the other hand, the agreeing respondents motivate that state-space explosion is indeed the most pressing challenge due to the huge input space of the operational environment (both in automotive and railway applications). One automotive researcher stresses that the state-space explosion impedes rigid testing but raises the transparency challenge as well—a lack thereof greatly limits analyzability, which is a key requirement for safety-critical systems. One automotive developer argues that the bigger state-space of the input domain, the bigger the attack surface becomes—possibly referring to both adversarial attacks and other antagonistic cyber attacks. Finally, two respondents provide answers that encourage us to continue work along to paths in the SMILE project: 1) a tester in the railway domain explains that the traceability during root cause analyses in ML-applications will be critical, in line with our argumentation at a recent traceability conference [109] and 2) one automotive architect argues that the state-space explosion will not be the main

challenge as any autonomous driving will have to be within "guard rails," that is, a solution similar to the safety cage architectures we intend to develop in the next phase of the project.

Seven respondents complemented the survey answers with concluding thoughts in Q18. One experienced manager in the aerospace domain explained: "What is now called ML was called neural nets (but less sophisticated) 30 years ago," a statement that supports our recommendation that the automotive industry should aim for a cross-domain knowledge transfer regarding V&V of ML-based systems. The manager followed by stating "it (ML) introduces a new element in safety engineering. Or at least it moves the emphasis to more resilience. If the classifier is wrong, then it becomes a hazard and the system must be prepared for it." We agree with the respondent that actions needed in the hazardous situation must be well specified. Two respondents comment that conservatism is fundamental in functional safety, one of them elaborates that the "end of predictability" introduced by ML is a disruptive change that requires a paradigm shift.

5. REVISITING THE RQs

This section first discusses the RQs in a larger context, and then aggregates the four sources of evidence presented in Figure 2. Finally, we discuss implications for research and practice, including automotive manufacturers and regulatory bodies, and conclude by reporting the main threats to validity. Table 4 summarizes our findings.

5.1. RQ1: State-of-the-Art in V&V of Safety-Critical ML

There is no doubt that deep learning research currently has incredible momentum. New applications and success stories are reported every month—and many applications come from the automotive domain. The rapid movement of the field is reflected by the many papers our study has identified on preprint archives, in particular the arXiv.org e-Print archive. It is evident that researchers are eager to claim novelty, and thus struggle to publish results as fast as possible.

While DNNs have enabled amazing breakthroughs, there is much less published work on engineering safety for DNNs. On the other hand, we observe a growing interest as several researchers call for more research on DNN safety, as well as ML safety in general. However, there is no agreement on how to best develop safety-critical DNNs, and several different approaches have been proposed. Contemporary research endeavors often address the opaqueness of DNNs, to support analyzability and interpretability of systems with DNN components.

Deep learning research is in its infancy, and the tangible pioneering spirit sometimes brings the mind to the Wild West. Anything goes, and there is a potential for great academic recognition for groundbreaking papers. There is certainly more fame in showcasing impressive applications than updating engineering practices and processes.

Safety engineering stands as a stark contrast to the pioneering spirit. On the contrary, safety is permeated by conservatism. When a safety

Table 4 Condensed findings in relation to the research questions, and implications for research and practice.

RQ1. What is the state-of-the-art in V&V of ML-based safety-critical systems?	• Most ML research showcases applications, while development on ML V&V is lagging behind. • Considerable gap between V&V mandated by safety standards and nature of contemporary ML-based systems. • The aerospace domain has collected experiences from V&V of adaptive flight controllers based on neural networks. • Support for V&V of ML-based systems can be organized into: 1) formal methods, 2) control theory, 3) probabilistic methods, 4) process guidelines, and 5) simulated test cases. • Academia has focused mostly on 1) to 3), whereas industry perceives 5) as the most promising.
RQ2. What are the main challenges when engineering safety-critical systems with DNN components in the automotive domain?	• How to certify safety-critical systems with DNNs for use on public roads is unclear. • Industry stresses robustness, whereas academia most often addresses state-space explosion and the lack of ML transparency. • Challenges elicited corroborate work on V&V by NASA and USAF, covering neural networks, autonomous systems, and adaptive systems.
Implications for research and practice	• Gap between ML practice and ISO 26262 requires novel standards rather than incremental updates. • Cross-domain knowledge transfer from the aerospace V&V engineers to the automotive domain appears promising. • Need for empirical studies to clarify what robustness means in the context of DNN-based autonomous vehicles. • Systems-based safety approaches encouraged by industry, including safety cage architectures and simulated test cases.

DNN, deep neural network; ML, machine learning; NASA, National Aeronautics and Space Administration; USAF, United States Air Force; V&V, verification and validation.

standard is developed, it captures the best available practices to engineer safe systems. This approach inevitably results in standards that lag behind the research front—safety first! In the automotive domain, ISO 26262 was developed long before DNNs for vehicles was an issue. Without question, DNNs constitute a paradigm shift in how to approach functional safety certification for automotive software, and we do not believe in any quick fixes to patch ISO 26262 for this new era. As recognized by researchers before us, for example, Salay *et al.* [3], there is a considerable gap between ML and ISO 26262—a gap that probably needs to be bridged by new standards rather than incremental updates of previous work.

Broadening the discussion from DNNs to ML in general, our systematic snowballing of previous research on safety-critical shows a peak of aerospace research between 2002 and 2007 and automotive research dominating from 2013 and onwards. We notice that the aerospace domain allocated significant resources to research on neural networks for adaptive flight controllers roughly a decade before DNNs became popular in automotive research. We hypothesize that considerable knowledge transfer between the domains is possible now, and plan to proceed such work in the near future.

The academic literature on challenges in ML-based safety engineering has most frequently addressed state-space explosion and robustness (see Table 2 for definitions). On the other hand, the most commonly proposed solutions to overcome challenges of ML-based safety engineering are approaches that belong to formal methods, control theory, or probabilistic methods—but these appear to be only moderately promising by industry practitioners, who would rather see research on simulated test cases. As discussed in relation to RQ2, academia and industry share a common view on what challenges are important, but the level of agreement on what is the best way forward appears to be less clear.

5.2. RQ2: Main Challenges for Safe Automotive DNNs

Industry practice is far from certifying DNNs for use in driverless safety-critical applications on public roads. Both the workshop series and the survey show that industry practitioners across organizations do not know how to tackle the challenge of approaching regulatory bodies and certification agencies with DNN-based systems. Most likely, both automotive manufacturers and safety standards need to largely adapt to fit the new ML paradigm—the current gap appears not to be bridgeable in the foreseeable future through incremental science alone.

On the other hand, although the current safety standards do not encompass ML yet, several automotive manufacturers are highly active in engineering autonomous vehicles. Tesla has received significant media coverage through pioneering demonstrations and self-confident statements. Volvo Cars is also highly active through the Drive Me initiative, and has announced a long-lasting partnership with Uber toward autonomous taxis.

Several other partnerships have recently been announced among automotive manufacturers, chipmakers, and ML-intensive companies. For example, Nvidia has partnered with Uber, Volkswagen, and Audi to support engineering self-driving cars using their GPU computing technology for ML development. Nvidia has also partnered with the Internet company Baidu, a company that has a highly competitive ML research group. Similarly, the chipmaker Intel has partnered with Fiat Chrysler Automobiles and the BMW Group to develop autonomy around their Mobileye solution. Moreover, large players such as Google, Apple, Ford, and Bosch are active in the area, as well as startups such as nuTonomy and FiveAI—no one wants to miss the boat to the lucrative future.

While there are impressive achievements both from spearheading research, and some features are already available on the consumer market, they all have in common that the safety case argumentation relies on a human-in-the-loop. In case there is a critical situation, the human driver is expected to be present and take control over the vehicle. There are joint initiatives to formulate regulations for autonomous vehicles, but, analogously, there is a need for initiatives paving the way for new standards addressing functional safety of systems that rely on ML and DNNs.

We elicited the most pressing issues concerning engineering of DNN-based systems through a workshop series and a survey with practitioners. Many discussions during the workshops were dominated by robustness of DNN components, including detailed considerations about robust DNN architectures and the requirements on training data to learn a robust DNN model. Also the survey shows the importance of ML robustness, which motivates the attention it has received in academic publications (cf. RQ1). On the other hand, while there is an agreement on the importance of ML robustness between academia and industry, how to tackle the phenomenon is still an open question—and thus a potential avenue for future research. Nonetheless, the problem of training a robust DNN component corresponding to the complexity of public traffic conforms with several of the "enduring problems" highlighted by the US Air Force in their technical report on V&V of autonomous systems **[P1]**, for example, state-space explosion and unpredictable environments.

While robustness is stressed by practitioners, academic publications have instead to a larger extent highlighted challenges related to the limited transparency of ML-based systems (e.g., Bhattacharyya *et al.* [96]) and the inevitable state-space explosion. The survey respondents confirm these challenges as well, but we recommend future studies to meet the expectations from industry regarding robustness research. Note, however, that the concept of robustness might have different interpretations despite having a formal IEEE definition [13]. Consequently, we call for an empirical study to capture what industry means by ML and DNN robustness in the automotive context.

The workshop participants perceived two possible approaches to pave the way for safety-critical DNNs as especially promising. First, continuous monitoring of DNN input using a safety cage architecture, a concept that has been proposed for example by Adler *et al.* [95]. Monitoring safe operation, and re-directing execution to a "safe track" without DNN involvement when uncertainties grow too large, is an example of the safety strategy safe fail [87]. Another approach to engineering ML safety, considered promising by the workshops and the survey respondents alike, is to simulate test cases.

6. CONCLUSION AND FUTURE WORK

DNNs is key to enable the vehicular perception required for autonomous driving. However, the behavior of DNN components

cannot be guaranteed by traditional software and system engineering approaches. On top of that, crucial parts of the automotive safety standard ISO 26262 are not well defined for certifying autonomous systems [3,110]—certain process requirements contravene the nature of developing ML-based systems, especially in relation to V&V.

Roughly a decade ago, using artificial neural networks (ANNs) in flight controllers was an active research topic, and also how to adhere to the strict aerospace safety standards. Now, in the advent of autonomous driving, we recommend the automotive industry to learn from guidelines [111] and lessons learned [107] from V&V of ANN-based components developed to conform with the DO-178B software safety standard for airborne systems. In particular, automotive software developers need to evolve practices for *configuration management* and *architecture specifications* to encompass fundamental DNN design elements. Also, *requirements specifications* and the corresponding *software testing* must be augmented to address the adaptive behavior of DNNs. Finally, the highly *iterative development lifecycle of DNNs* should be aligned with the traditional automotive V-model for systems development. A recent NASA report on safety certification of adaptive aerospace systems [96] confirms the challenges of requirements specification and software testing. Moreover, related to ML, the report adds the *lack of documentation and traceability* in many open-source libraries, and the issue of an *expertise gap between regulators and engineers*—conventional source code in C/C++ is very different from an opaque ML model trained on a massive dataset.

The work most similar to ours also originated in the aerospace domain, that is, a project initiated by the US Air Force to describe enduring problems (and future possibilities) in relation to safety certification of autonomous systems **[P1]**. The project highlighted four primary challenges: 1) state-space explosion, 2) unpredictable environments, 3) emergent behavior, and 4) human-machine communication. While not explicitly discussing ML, the first two findings match the most pressing needs elicited in our work, that is, *state-space explosion as stressed by the academic literature* (in combination with limited transparency) and *robustness as emphasized by the workshop participants as well as the survey respondents* (referred to as unpredictable environments in **[P1]**).

After having reviewed the state-of-the-art and state-of-practice, the SMILE project will now embark on a solution-oriented journey. Based on the workshops, and motivated by the survey respondents, we conclude that *pursuing a solution based on safety cage architectures* [90,95] encompassing DNN components is a promising direction. Our rationale is three-fold. First, the results from the workshops with automotive experts from industry clearly motivates us, that is, the participants strongly encouraged us to explore such a solution as the next step. Second, we believe it would be feasible to develop a *safety case* around a safety cage architecture, since the automotive industry already uses the concept in the physical vehicles. Third, we believe the DNN technology is ready to provide what is needed in terms of novelty detection. The safety cage architecture we envision will continuously monitor input data from the operational environment to redirect executionto a non-ML safe track when uncertainties grow too large. Consequently, we advocate *DNN safety strategies using a systems-based approach* rather than techniques that focus on the internals of DNNs. Finally, also motivated by both the workshops and the survey respondents, we propose an approach to V&V that makes heavy use of *simulation*—in line with previous recommendations by other researchers [21,112,113].

Future work will also study how *transfer learning* could be used to incorporate training data from different contexts or manufacturers, or even include synthetic data from simulators, into DNNs for real-world automotive perception. So far we have mostly limited the discussion to fixed DNN-based systems, that is, systems trained only prior to deployment. An obvious direction for future work is to explore how dynamic DNNs would influence our findings, that is, DNNs that adapt by continued learning either in batches or through online learning. Furthermore, research on V&V of ML-based systems is more complex than pure technology in isolation. Thus, we recognize the need to explore both ethical and legal aspects involved in safety certification of ML-based systems. Finally, there is a new automotive standard under development that will address autonomous safety: ISO/PAS 21448 Road vehicles—safety of the intended functionality. We are not aware of its contents at the time of this writing, but once published, we will use it as an important reference point for our future solution proposals.

ACKNOWLEDGMENTS

Thanks go to all participants in the SMILE workshops, in particular Carl Zandén, Michaël Simoen, and Konstantin Lindström. This work was carried out within the SMILE and SMILE II projects financed by Vinnova, FFI, Fordonsstrategisk forskning och innovation under the grant numbers: 2016-04255 and 2017-03066. We would like to acknowledge that this work was supported by the KKS foundation through the S.E.R.T. Research Profile project at Blekinge Institute of Technology.

References

[1] Ramos, S, Gehrig, S, Pinggera, P, Franke, U, Rother, C. Detecting unexpected obstacles for self-driving cars: fusing deep learning and geometric modeling. In: 2017 IEEE Intelligent Vehicles Symposium (IV), Los Angeles, CA; 2017, p. 1025–1032.

[2] Falcini, F Lami, G Costanza, A. Deep learning in automotive software. IEEE Softw 2017;34(3);56–63.

[3] Salay, R, and Queiroz, R, Czarnecki, K. An Analysis of ISO 26262: Machine Learning and Safety in Automotive Software. SAE Technical Paper 2018-01-1075, 2018.

[4] Katz, G, Barrett, C, Dill, DL, Julian, K, Kochenderfer, MJ. Reluplex: an efficient SMT solver for verifying deep neural networks. In: Majumdar, R, Kunčak, V, editors. Computer aided Verification. CAV 2017. Lecture Notes in Computer Science, Cham: Springer; 2017, vol. 10426.

[5] Zhu, H, Yuen, K, Mihaylova, L, Leung, H. Overview of environment perception for intelligent vehicles. IEEE Trans Intell Trans Syst 2017;18(10);2584–601.

[6] Gurghian, A, Koduri, T, Bailur, SV, Carey, KJ, Murali, VN. Deeplanes: end-to-end lane position estimation using deep neural networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, Las Vegas, NV; 2016, p. 38–45.

[7] LeCun, Y, Bengio, Y, Hinton, G. Deep learning. Nature 2015; 521(7553);436–44.
[8] He, K, Zhang, X, Ren, R, Sun, J. Delving deep into rectifiers: surpassing human-level performance on ImageNet classification. In: Proceedings of the International Conference on Computer Vision, Santiago, Chile; 2015.
[9] Spanfelner, B, Richter, D, Ebel, S, Wilhelm, U, Branz, W, Patz C. Challenges in applying the ISO 26262 for driver assistance. In: Proceedings of the Schwerpunkt Vernetzung, 5. Tagung Fahrerassistenz, Munich, 2012.
[10] Wohlin, C. Guidelines for snowballing in systematic literature studies and a replication in software engineering. In: Proceedings of the 18th International Conference on Evaluation and Assessment in Software Engineering, London, UK; 2014, p. 38.
[11] International Organization for Standardization. ISO 26262 Road vehicles—Functional safety, 2011.
[12] International Electrotechnical Commission. IEC 61508 ed 1.0, Electrical/electronic/programmable electronic safety-related systems, 2010.
[13] IEEE Computer Society. 610.12-1990 IEEE standard glossary of software engineering terminology. Technical report, 1990.
[14] Billinton, R, Allan, R, editors. Reliability evaluation of engineering systems: concepts and techniques. Berlin/Heidelberg, Germany: Springer Science & Business Media; 2013.
[15] Bahr, NJ. System safety engineering and risk assessment: a practical approach. 2nd ed., Boca Raton, FL: CRC Press; 2014.
[16] National Instruments. What is the ISO 26262 Functional Safety Standard?
[17] He, K, Zhang, X, Ren, S, Sun, J. Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV; 2016, p. 770–78.
[18] Szegedy, C, Liu, W, Jia, Y, Sermanet, P, Reed, S, Anguelov, D, Erhan, D, Vanhoucke, V, Rabinovich, A. Going deeper with convolutions. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA; 2015, p. 1–9.
[19] Simonyan, K, Zisserman, A. Very deep convolutional networks for large-scale image recognition. arXiv: 1409.1556, 2014.
[20] Donahue, J, Jia, Y, Vinyals, O, Hoffman, J, Zhang, N, Tzeng, E, Darrell, T. DeCAF: a deep convolutional activation feature for generic visual recognition. In: PMLR, 2014, p. 647–55.
[21] Bojarski, M, Testa, DD, Dworakowski, D, Firner, B, Flepp, B, Goyal, P, Jackel, LD, Monfort, M, Muller, U, Zhang, J, Zhang, X, Zhao, J, Zieba, K. End to end learning for self-driving cars. arXiv: 1604.07316, 2016.
[22] Sallab, A, Abdou, M, Perot, E, Yogamani, S. End-to-end deep reinforcement learning for lane keeping assist. In: Proceedings of the Machine Learning for Intelligent Transportation Systems Workshop, NIPS 2016, Barcelona, Spain; 2016.
[23] Goodfellow, I. NIPS 2016 tutorial: generative adversarial networks. arXiv: 1701.00160, 2016.
[24] Radford, A, Metz, L, Chintala, S. Unsupervised representation learning with deep convolutional generative adversarial networks. arXiv: 1511.06434, 2015.
[25] Glorot, X, Bordes, A, Bengio, Y. Domain adaptation for large-scale sentiment classification: a deep learning approach. In: Proceedings of the 28th International Conference on International Conference on Machine Learning, Bellevue, Washington; 2011, p. 513–20.
[26] Ivarsson, M, Gorschek, T. A method for evaluating rigor and industrial relevance of technology evaluations. Empir Softw Eng 2011;16(3);365–95.
[27] Clark, M, Kearns, K, Overholt, J, Gross, K, Barthelemy, B, Reed, C. Air force research laboratory test and evaluation, verification and validation of autonomous systems challenge exploration. Technical report, Air Force Research Lab Wright-Patterson, 2014.
[28] Amodei, D, Olah, J, Steinhardt, J, Christiano, P, Schulman, J, Mane, D. Concrete problems in AI safety. arXiv: 1606.06565, 2016.
[29] Brat, G, Jonsson, A. Challenges in verification and validation of autonomous systems for space exploration. In: Proceedings of the IEEE International Joint Conference on Neural Networks, Montreal, Canada; 2005, vol. 5, p. 2909–14.
[30] Broggi, A, Buzzoni, M, Debattisti, S, Grisleri, P, Laghi, MC, Medici, P, Versari, P. Extensive tests of autonomous driving technologies. IEEE Trans Intell Trans Syst 2013;14(3);1403–15.
[31] Taylor, BJ, Darrah, MA, Moats, CD. Verification and validation of neural networks: a sampling of research in progress. In: Proceedings Volume 5103, AEROSENSE 2003, Intelligent Computing: Theory and Applications, Orlando, FL; 2003, vol. 5103, p. 8–16.
[32] Taylor, J, Yudkowsky, E, LaVictoire, P, Critch, A. Alignment for advanced machine learning systems. Technical report, Machine Intelligence Research Institute, 2016.
[33] Carvalho, A, Lefevre, S, Schildbach, G, Kong, J, Borelli, F. Automated driving: the role of forecasts and uncertainty—a control perspective. Eur J Control 2015; 24;14–32.
[34] Alexander, R, Hawkins, HR, JohnRae, A. Situation coverage—a coverage criterion for testing autonomous robots. Technical report, University of York, 2015.
[35] Zou, X, Alexander, R, McDermid, J. On the validation of a UAV collision avoidance system developed by model-based optimization: challenges and a tentative partial solution. In: Proceedings of the 46th Annual IEEE/IFIP International Conference on Dependable Systems and Networks Workshop, Toulouse, France; 2016, p. 192–99.
[36] Zou, X, Alexander, R, McDermid, J. Safety validation of sense and avoid algorithms using simulation and evolutionary search. In: Bondavalli, A, Di Giandomenico, F, editors. Computer safety, reliability, and security. SAFECOMP 2014. Lecture notes in computer science, Cham: Springer; 2014, vol. 8666, p. 33–48.
[37] Arnold, J, Alexander, R. Testing autonomous robot control software using procedural content generation. In: Bitsch, F, Guiochet, J, Kaâniche, M, editors. Computer safety, reliability, and security. SAFECOMP 2013. Lecture notes in computer science, Berlin, Heidelberg: Springer; 2013, vol. 8153.
[38] Sivaraman, S, Trivedi, MM. Active learning for on-road vehicle detection: a comparative study. Mach Vision Appl 2014;25(3);599–611.
[39] Mozaffari, A, Vajedi, M, Azad, NL. A robust safety-oriented autonomous cruise control scheme for electric vehicles based on model predictive control and online sequential extreme learning machine with a hyper-level fault tolerance-based supervisor. Neurocomputing 2015;151;845–56.
[40] Seshia, SA, Sadigh, D, ShankarSastry, S. Towards verified artificial intelligence. arXiv: 1606.08514, 2016.
[41] Helle, P, Schamai, W, Strobel, C. Testing of autonomous systems—challenges and current state-of-the-art. In: Proceedings

of the 26th Annutal INCOSE International Symposium, Edinburgh, UK; 2016, p. 571–84.
[42] Li, L, Huang, WL, Liu, Y, Zheng, NN, Wang, FY. Intelligence testing for autonomous vehicles: a new approach. IEEE Trans Intell Veh 2016;1(2);158–66.
[43] Shi, W, BakerAlawieh, M, Li, X, Yu, H, Arechiga, N, Tomatsu, N. Efficient statistical validation of machine learning systems for autonomous driving. In: Proceedings of the 35th International Conference on Computer-Aided Design, Austin, Texas; 2016, vol. 8, p. 1–36.
[44] Sullivan, KB, Feigh, KM, Durso, FT, Fischer, U, Pop, VL, Mosier, K, Blosch, J, Morrow, D. Using neural networks to assess human-automation interaction. In: IEEE/AIAA 30th Digital Avionics Systems Conference, Seattle, Washington; 2011, p. 6A4–1–6A4–10.
[45] Broderick, RL. Adaptive verification for an on-line learning neural-based flight control system. In: Proceedings of the 24th Digital Avionics Systems Conference, Washington, DC; 2005, vol. 1, p. 6.C.2–61–10.
[46] Li, N, Oyler, D, Zhang, M, Yildiz, Y, Kolmanovsky, I, Girard, A. Game-theoretic modeling of driver and vehicle interactions for verification and validation of autonomous vehicle control systems. arXiv: 1608.08589, 2016.
[47] Russell, S, Dewey, D, Tegmark, M. Research priorities for robust and beneficial artificial intelligence. arXiv: 1602.03506, 2016.
[48] Broggi, A, Cerri, P, Medici, P, Porta, PP, Ghisio, G. Real time road signs recognition. In: IEEE Intelligent Vehicles Symposium, Istanbul, Turkey; 2007, p. 981–86.
[49] Schumann, J, Nelson, S. Toward V&V of neural network based controllers. In: Proceedings of the 1st Workshop on Self-healing Systems, Charleston, SC; 2002, p. 67–72.
[50] Hull, J, Ward, D, Zakrzewski, RR. Verification and validation of neural networks for safety-critical applications. In: Proceedings of the 2002 American Control Conference, Anchorage, AK; 2002, vol. 6, p. 4789–94.
[51] Pulina, L, Tacchella, A. An abstraction-refinement approach to verification of artificial neural networks. In: Touili, T, Cook, B, Jackson, P, editors. Computer aided verification. CAV 2010. Lecture notes in computer science, Berlin, Heidelberg: Springer; 2010, vol. 6174, p. 243–57.
[52] Lefevre, S, Vasquez, D, Laugier, C. A survey on motion prediction and risk assessment for intelligent vehicles. ROBOMECH J 2014;1(1);1–14.
[53] Huang, X, Kwiatkowska, M, Wang, S, Wu, M. Safety verification of deep neural networks. In: Majumdar, R, Kunčak, V, editors. Computer aided verification. CAV 2017. Lecture notes in computer science, Cham: Springer; 2017, vol. 10426.
[54] Gupta, P, Schumann, J. A tool for verification and validation of neural network based adaptive controllers for high assurance systems. In: Proceedings of the 8th IEEE International Symposium on High Assurance Systems Engineering, Tampa, FL; 2004, p. 277–78.
[55] Schumann, J, Gupta, P, Jacklin, S. Toward verification and validation of adaptive aircraft controllers. In: Proceedings of the IEEE Aerospace Conference, Big Sky, MT; 2005, p. 1–6.
[56] Broderick, RL. Statistical and adaptive approach for verification of a neural-based flight control system. In: Proceedings of the 23rd Digital Avionics Systems Conference, Salt Lake City, UT; 2004, vol. 2, p. 6.E.1–61–10.
[57] Liu, Y, Cukic, B, Gururajan, S. Validating neural network-based online adaptive systems: a case study. Softw Qual J 2007;15(3);309–26.
[58] Yerramalla, S, Liu, Y, Fuller, E, Cukic, B, Gururajan, S. An approach to V&V of embedded adaptive systems. In: Hinchey, MG, Rash, JL, Truszkowski, WF, Rouff, CA, editors. Formal approaches to agent-based systems, Berlin, Heidelberg: Springer Berlin Heidelberg; 2005, p. 173–88.
[59] Zakrzewski, RR. Randomized approach to verification of neural networks. In: Proceedings of the IEEE International Joint Conference on Neural Networks, Budapest, Hungary; 2004, vol. 4, p. 2819–24.
[60] Akametalu, AK, Fisac, JF, Gillula, JH, Kaynama, S, Zeilinger, MN, Tomlin, CJ. Reachability-based safe learning with Gaussian processes. In: Proceedings of the 53rd IEEE Conference on Decision and Control, Los Angeles, CA; 2014, p. 1424–31.
[61] Seshia, SA, Sadigh, D, Shankar Sastry, S. Formal methods for semi-autonomous driving. In: Proceedings of the 52nd Annual Design Automation Conference, San Francisco, CA; 2015, vol. 5, p. 1–148.
[62] Mili, A, Jiang, P, Cukic, B, Liu, Y, Ayed, RB. Towards the verification and validation of online learning systems: general framework and applications. In: Proceedings of the 37th Annual Hawaii International Conference on System Sciences, Waikoloa Village, HI; 2004.
[63] Kurd, Z, Kelly, T, Austin, J. Developing artificial neural networks for safety critical systems. Neural Comput Appl 2007; 16(1);11–19.
[64] Pulina, L, Tacchella, A. NeVer: a tool for artificial neural networks verification. Ann Math Artif Intell 2011;62(3);403–25.
[65] Zakrzewski, RR. Verification of a trained neural network accuracy. In: Proceedings of the International Joint Conference on Neural Networks, Washington, DC; 2001, vol. 3, p. 1657–62.
[66] Mackall, D, Nelson, S, Schumman, J. Verification and validation of neural networks for aerospace systems. Technical report, 2002.
[67] Jacklin, S, Schumann, J, Gupta, P, Richard, M, Guenther, K, Soares, F. Development of advanced verification and validation procedures and tools for the certification of learning systems in aerospace applications. In: Proceedings of Infotech@Aerospace, American Institute of Aeronautics and Astronautics, Arlington, Virginia; 2005.
[68] Nguyen, NT, Jacklin, SA. Neural net adaptive flight control stability. In: Verification and Validation Challenges, and Future Research, IJCNN Conference, 2007.
[69] Schumann, J, Liu, Y. Tools and methods for the verification and validation of adaptive aircraft control systems. In: Proceedings of the IEEE Aerospace Conference, Big Sky, MT; 2007, p. 1–8.
[70] Nguyen, NT, Jacklin, SA. Stability, convergence, and verification and validation challenges of neural net adaptive flight control. In: Schumann, J, Liu, Y, editors. Applications of neural networks in high assurance systems, Berlin, Heidelberg: Springer Berlin Heidelberg; 2010, p. 77–110.
[71] Jacklin, SA, Schumann, J, Bosworth, JT, Williams-Hayes, PS, Larson, RS. Case study: test results of a tool and method for in-flight, adaptive control system verification on a NASA F-15 flight research aircraft. In: Proceedings of the 7th World Congress on Computational Mechanics Minisymposium: Accomplishments and Challenges in Verification and Validation, Los Angeles, CA; 2006.

[72] Li, G, Lu, M, Liu, B. A scenario-based method for safety certification of artificial intelligent software. In: Proceedings of the 2010 International Conference on Artificial Intelligence and Computational Intelligence, Sanya, China; 2010, vol. 3, p. 481–83.

[73] Gupta, P, Guenther, K, Hodgkinson, J, Jacklin, S, Richard, M, Schumann, J, Soares, F. Performance monitoring and assessment of neuro-adaptive controllers for aerospace applications using a Bayesian approach. In: AIAA Guidance, Navigation, and Control Conference and Exhibit. San Francisco, CA: American Institute of Aeronautics and Astronautics; 2005.

[74] Scheibler, K, Winterer, L, Wimmer, R, Becker, B. Towards verification of artificial neural networks. In: Proceedings of MBMV, Chemnitz, Germany; 2015.

[75] Gupta, P, Loparo, KA, Mackall, D, Schumann, J, Soares, FR. Verification and validation methodology of real-time adaptive neural networks for aerospace applications. Technical report, NASA, 2004.

[76] Jacklin, S, Schumann, J, Gupta, P, Lowry, M, Bosworth, J, Zavala, E, Hayhurst, K, Belcastro, C, Belcastro, C. Verification, validation, and certification challenges for adaptive flight-critical control system software. In: AIAA Guidance, Navigation, and Control and Exhibit, Providence, RI; 2004.

[77] Cortellessa, V, Cukic, B, DelGobbo, B, Mili, A, Napolitano, M, Shereshevsky, M, Sandhu, H. Certifying adaptive flight control software. In: Proceedings of the Software Risk Management Conference, Limerick, Ireland; 2000.

[78] Yerramalla, S, Fuller, E, Mladenovski, M, Cukic, B. Lyapunov analysis of neural network stability in an adaptive flight control system. In: Proceedings of the 6th International Conference on Self-stabilizing Systems, San Francisco, CA; 2003, p. 77–92.

[79] Jacklin, S. Closing the certification gaps in adaptive flight control software. In: AIAA guidance, Navigation and Control Conference and Exhibit. Honolulu, HI: American Institute of Aeronautics and Astronautics; 2008.

[80] Soares, F, Burken, J, Marwala, T. Neural network applications in advanced aircraft flight control system, a hybrid system, a flight test demonstration. In: King, I, Wang, J, Chan, L-W, Wang, D, editors. Neural information processing, Berlin, Heidelberg: Springer Berlin Heidelberg; 2006, p. 684–91.

[81] Zhang, X, Clark, M, Rattan, K, Muse, J. Controller verification in adaptive learning systems towards trusted autonomy. In: Proceedings of the ACM/IEEE 6th International Conference on Cyber-Physical Systems, Seattle, WA; 2015, p. 31–40.

[82] Soares, F, Burken, J. A Flight Test Demonstration of On-line Neural Network Applications in Advanced Aircraft Flight Control System. In: Proceedings of the 2006 International Conference on Computational Inteligence for Modelling Control and Automation and International Conference on Intelligent Agents Web Technologies and International Commerce, Sydney, Australia; 2006, p. 136.

[83] Torens, C, Adolf, F-N, Goormann, L. Certification and software verification considerations for autonomous unmanned aircraft. J Aerosp Info Sys 2014;11(10);649–64.

[84] Bosworth, J, Williams-Hayes, P. Flight test results from the NF-15b intelligent flight control system project with adaptation to a simulated stabilitor failure. Technical report, NASA TM-2007-214629, 2007.

[85] Zakrzewski, RR. Verification of performance of a neural network estimator. In: Proceedings of the 2002 International Joint Conference on Neural Networks, Honolulu, HI; 2002, vol. 3, p. 2632–37.

[86] Cruzes, DS, Dyba, T. Recommended steps for thematic synthesis in software engineering. In: Proceedings of the international symposium on empirical software engineering and measurement, Banff, Canada; 2011, p. 275–84.

[87] Varshney, K. Engineering safety in machine learning. In: Proceedings of the 2016 Information Theory and Applications Workshop, San Diego, CA; Jan. 2016, p. 1–5.

[88] Rea, LM, Parker, RA. Designing and conducting survey research: A comprehensive guide. 4th ed., Hoboken, NJ: John Wiley & Sons; 2014.

[89] Knauss, A, Schroeder, J, Berger, C, Eriksson, H. Software-related challenges of testing automated vehicles. In: Proceedings of the 39th International Conference on Software Engineering Companion, Buenos Aires, Argentina; 2017, p. 328–30.

[90] Heckemann, K, Gesell, M, Pfister, T, Berns, K, Schneider, K, Trapp, M. Safe Automotive Software. In: König, A, Dengel, A, Hinkelmann, K, Kise, K, Howlett, RJ, Jain, LC, editors. Knowledge-Based and Intelligent Information and Engineering Systems. KES 2011. Lecture notes in computer science, Berlin, Heidelberg: Springer; 2011 vol. 6884, p. 167–76.

[91] Moller, N, OveHansson, S. Principles of engineering safety: risk and uncertainty reduction. Rel Eng Syst Safety 2008;93(6); 798–805.

[92] Bojarski, M, Choromanska, A, Choromanski, K, Firner, B, Jackel, L, Muller, U, Zieba, K. VisualBackProp: efficient visualization of CNNs. arXiv: 1611.05418, 2016.

[93] Mhamdi, EME, Guerraoui, R, Rouault, S. On the robustness of a neural network. In: 2017 IEEE 36th Symposium on Reliable Distributed Systems (SRDS), Hong Kong; 2017, 84–93.

[94] Varshney, K, Prenger, R, Marlatt, T, Chen, B, Hanley, W. Practical ensemble classification error bounds for different operating points. IEEE Trans Knowl Data Eng 2013;25(11);2590–601.

[95] Adler, R, Feth, P, Schneider, D. Safety engineering for autonomous vehicles. In: Proceedings of the 46th Annual IEEE/IFIP International Conference on Dependable Systems and Networks Workshops, Toulouse, France; 2016, p. 200–205.

[96] Bhattacharyya, S, Cofer, D, Musliner, D, Mueller, J, Engstrom, E. Certification considerations for adaptive systems. Technical report CR2015-218702, NASA, 2015.

[97] Angeline, PJ, Saunders, GM, Pollack, JB. An evolutionary algorithm that constructs recurrent neural networks. IEEE Trans Neural Netw 1994;5(1);54–65.

[98] Yao, X, Liu, Y. A new evolutionary system for evolving artificial neural networks. IEEE Trans Neural Netw 1997;8(3); 694–713.

[99] Sixt, L, Wild, B, Landgraf, T. Rendergan: generating realistic labeled data. Front Robot AI 2018;5;66.

[100] Oquab, M, Bottou, L, Laptev, I, Sivic, J. Learning and transferring mid-level image representations using convolutional neural networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH; 2014, p. 1717–24.

[101] Agrawal, A, Chen, C-Y, Choi, J, Gopalakrishnan, K, Oh, J, Shukla, S, Srinivasan, V, Venkataramani, S, Zhang, W. Accelerator design for deep learning training: extended abstract: invited. In: Proceedings of the 54th Annual Design Automation Conference, Austin, TX; 2017, vol. 2, p. 1–57.

[102] Pimentel, MAF, Clifton, DA, Clifton, L, Tarassenko, L. A review of novelty detection. Signal Process 2014;99;215–49.

[103] Brasileiro, FV, Ezhilchelvan, PD, Shrivastava, SK, Speirs, NA, Tao, S. Implementing fail-silent nodes for distributed systems. IEEE Trans Comput 1996;45(11);1226–38.

[104] Korte, M, Holzmann, F, Kaiser, G, Scheuch, V, Roth, H. Design of a robust plausibility check for an adaptive vehicle observer in an electric vehicle. In: Meyer, G, editor. Advanced microsystems for automotive applications 2012, Berlin, Heidelberg: Springer; 2012, p. 109–19.

[105] Maji, D, Santara, A, Mitra, P, Sheet, D. Ensemble of deep convolutional neural networks for learning to detect retinal vessels in fundus images. arXiv: 1603.04833, 2016.

[106] Bjarnason, E, Runeson, P, Borg, M, Unterkalmsteiner, M, Engstrom, E, Regnell, B, Sabaliauskaite, G, Loconsole, A, Gorschek, T, Feldt, R. Challenges and practices in aligning requirements with verification and validation: a case study of six companies. Empirical Softw Eng 2014;19(6);1809–55.

[107] Taylor, BJ. Methods and procedures for the verification and validation of artificial neural networks. Berlin/Heidelberg, Germany: Springer Science & Business Media; 2006. Google-Books-ID: ax3Q_YBuXFEC.

[108] Schumann, J, Gupta, P, Liu, Y. Application of neural networks in high assurance systems: a survey. In: Applications of neural networks in high assurance systems, studies in computational intelligence, Berlin, Heidelberg: Springer; 2010, p. 1–19.

[109] Borg, M, Englund, C, Duran, B. Traceability and deep learning—safety-critical systems with traces ending in deep neural networks. In: Proceedings of the Grand Challenges of Traceability: The Next Ten Years, Lexington, Kentucky; 2017, p. 48–49.

[110] Henriksson, J, Borg, M, Englund, C. Automotive safety and machine learning: initial results from a study on how to adapt the ISO 26262 safety standard. In: Proceedings of the 1st International Workshop on Software Engineering for AI in Autonomous Systems, IEEE, Gothenburg, Sweden; 2018, p. 47–49.

[111] Pullum, L, Taylor, B, Darrah, M. Guidance for the verification and validation of neural networks. Hoboken, NJ: John Wiley & Sons, Inc.; 2007.

[112] Abdessalem, RB, Nejati, S, Briand, LC, Stifter, T. Testing advanced driver assistance systems using multi-objective search and neural networks. In: Proceedings of the 31st International Conference on Automated Software Engineering, Singapore: ACM; 2016, p. 63–74.

[113] Tian, Y, Pei, K, Jana, S, Ray, B. Deeptest: automated testing of deep-neural-network-driven autonomous cars. In: Proceedings of the 40th International Conference on Software Engineering, Gothenburg, Sweden: ACM; 2018, p. 303–14.