Use case application of PD ISO/PAS 8800 for machine learning (ML) components

January 2025

Your partner in progress

Centre for Connected & Autonomous Vehicles

About CCAV

The Centre for Connected and Autonomous Vehicles (CCAV) is a joint Department for Business and Trade (DBT) and Department for Transport (DfT) unit. Established in 2015, CCAV is an expert unit that is working with industry and academia to make every-day journeys greener, safer, more flexible and more reliable by shaping the safe and secure emergence of connected and self-driving vehicles in the UK and by leading the government's Future of Transport strategy.

Author

Richard Hawkins – University of York

Richard Hawkins is a senior lecturer in the Department of Computer Science at the University of York. Richard specialises in safety cases for autonomous systems and AI. He is also a member of BSI Technical Committee AUE/32/1, Safety and security of electrical and electronic components in road vehicles that is the mirror committee of ISO/TC 22/SC 32.

Disclaimer

This report has been prepared for general information purposes relating to its subject matter only.

For more information on its subject matter specifically, please contact CAV@bsigroup.com.

Contents

1 Introduction

This document provides a use case of how to apply the framework outlined in PD ISO/PAS 8800: Road vehicles - Safety and artificial intelligence to a machine learning (ML)-based object detection function as part of an automotive system. The use case illustrates how the different phases of the AI safety lifecycle fit together for a single example, enabling the development of an assurance argument for the AI system. Please note that not all the requirements in PD ISO/PAS 8800 are addressed by this example. The use case provided in this paper covers the sections of PD ISO/PAS 8800 highlighted in Figure 1, including:

- Refinement of AI safety requirements.
- AI safety lifecycle (see Figure 2).
- Development of the safety assurance argument.

This use case describes the work products that are generated for the ML component. In this use case, the ML component is used to identify traffic signs present within the operational design domain (ODD) and shows how these work products can be used as artefacts in a safety assurance argument. For this document, the term 'AI system' includes the ML component providing object detection functionality.

This use case focuses on addressing AI errors caused by functional insufficiencies of the ML component and assumes that risks associated with AI errors caused by random hardware failures or systematic faults in the execution platform are addressed elsewhere.

Figure 1 – Summary AI safety lifecycle from PD ISO/PAS 8800

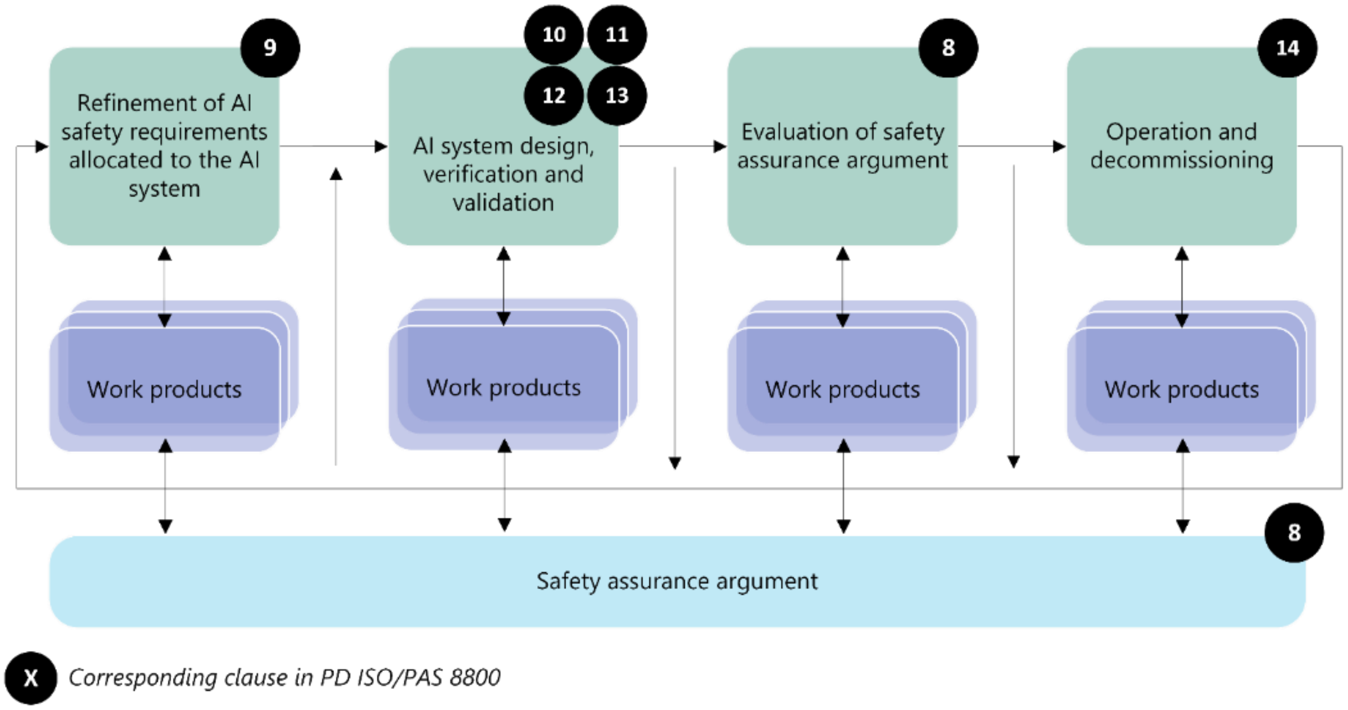
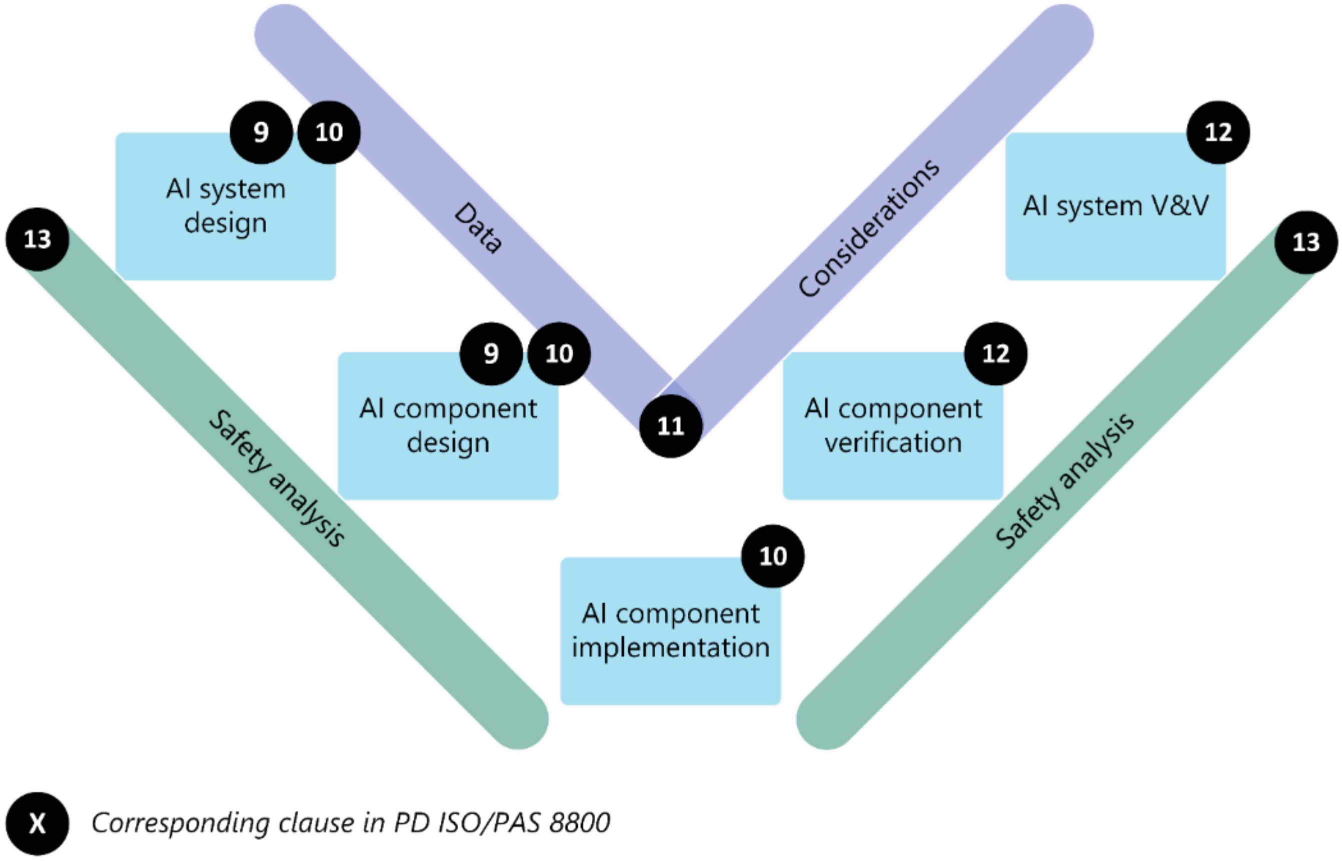

9
Refinement of AI safety requirements allocated to the AI system
10
11
12
13
AI system design, verification and validation
8
Evaluation of safety assurance argument
14
Operation and decommissioning
Work products
Work products
Work products
Work products
8
Safety assurance argument
X
Corresponding clause in PD ISO/PAS 8800

Figure 2 – Detailed view of the AI system design and V&V phase of the reference AI safety lifecycle adapted from PD ISO/PAS 8800



2 Definition of safety requirements allocated to the AI system

2.1 Objectives and requirements from PD ISO/PAS 8800

The objectives of this phase of the AI safety lifecycle are to:

- Specify a complete and consistent set of AI safety requirements that are sufficient to ensure AI safety.
- Refine AI safety requirements based on learnings from development, verification and validation.
- Specify the limitations of an AI system over its input space to be escalated to its encompassing system development process.

This involves fulfilling the following requirements.

- The input space definition of the AI system shall be refined to the degree suitable for initiating the AI safety lifecycle.
- To provide a connection between each AI safety requirement and the addressed problem, the refined AI safety requirements shall either:
    - Trace to the safety requirements allocated to the AI system (from external sources), assumptions or critical scenarios; or
    - Address and trace to the potential influencing factors or root causes of functional insufficiencies and triggering conditions.
- A justification shall be provided that the refined AI safety requirements are reasonable to either ensure the achievement of the safety requirements allocated to the AI system (from external sources) or prevent or mitigate the functional insufficiencies at the AI system level.
- To argue for the absence of unreasonable risk due to random hardware faults and classic systematic faults, the requirements of the BS ISO 26262 series[1] shall be fulfilled.
- The following cases shall be identified and reported to the encompassing system development process:
    - The AI system does not fully conform to the AI safety requirements; and
    - The AI safety requirements are only fulfilled for a limited part of the input space.
- AI safety requirements shall be identified to support the measures ensuring AI safety during operation.

2.2 Work products

This phase of the AI safety lifecycle results in the following work products:

- Input space definition (**2.5.1**).
- AI safety requirements (**2.5.2**).
- Known insufficiencies of the AI system and the corresponding subdomains of the input space.

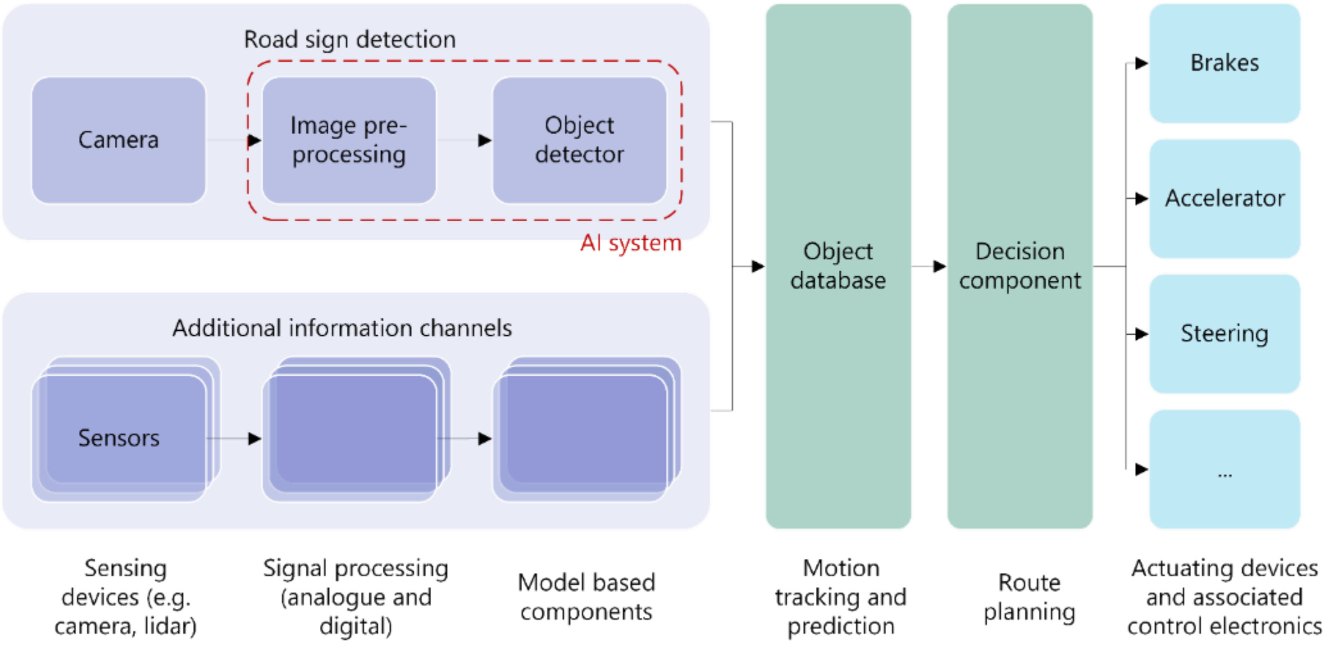
[1] BSI ISO 26262 (all parts). *Road vehicles – Functional safety*

2.3 Description of the encompassing system context

A prerequisite to the development of the AI safety requirements is a definition of the scope of the AI system and its context. Figure 3 shows an abstract representation of the autonomous driving platform that is considered in this use case which is designed to fulfil the defined safety requirements. This is referred to in this document as the 'encompassing system'. The architecture of the platform includes multiple parallel information processing channels to control the behaviour of the vehicle. The decision-making component is reliant on an accurate understanding of the operating domain within which the vehicle is operating and the state of objects within that environment. This in turn relies on the ability of the perception functions to identify the types of objects in the space (including road signs) as well as their location with respect to the ego vehicle.

In the case of road signs, the primary information channel for road sign detection processes raw data gathered from a video camera which is pre-processed before being passed to the object detection algorithm.

Figure 3 – Representation of the autonomous driving platform considered in the use case showing the AI System as part of the encompassing system


Road sign detection
Camera
Image pre-processing
Object detector
AI system
Additional information channels
Sensors
Object database
Decision component
Brakes
Accelerator
Steering
...
Sensing devices (e.g. camera, lidar)
Signal processing (analogue and digital)
Model based components
Motion tracking and prediction
Route planning
Actuating devices and associated control electronics

2.4 Definition of the AI system scope

The road sign detection functionality can be considered, conceptually, as having three functions as shown in Figure 4.

1. **Image pre-processing**

   The raw images received from the camera are pre-processed to create input amenable for analysis by the ML component. This may include normalization of light levels, removal of sensor noise and resizing of images. This component might also reject images that do not fulfil certain characteristics. This allows the ML component to make certain assumptions on the characteristics of the received images. Criteria for rejecting images may be, for example, due to too low lighting, or too high noise, detectable camera damage or sensor damage (dead pixels), etc. On rejecting an image, a safe state mechanism at the level of the encompassing system could be triggered.

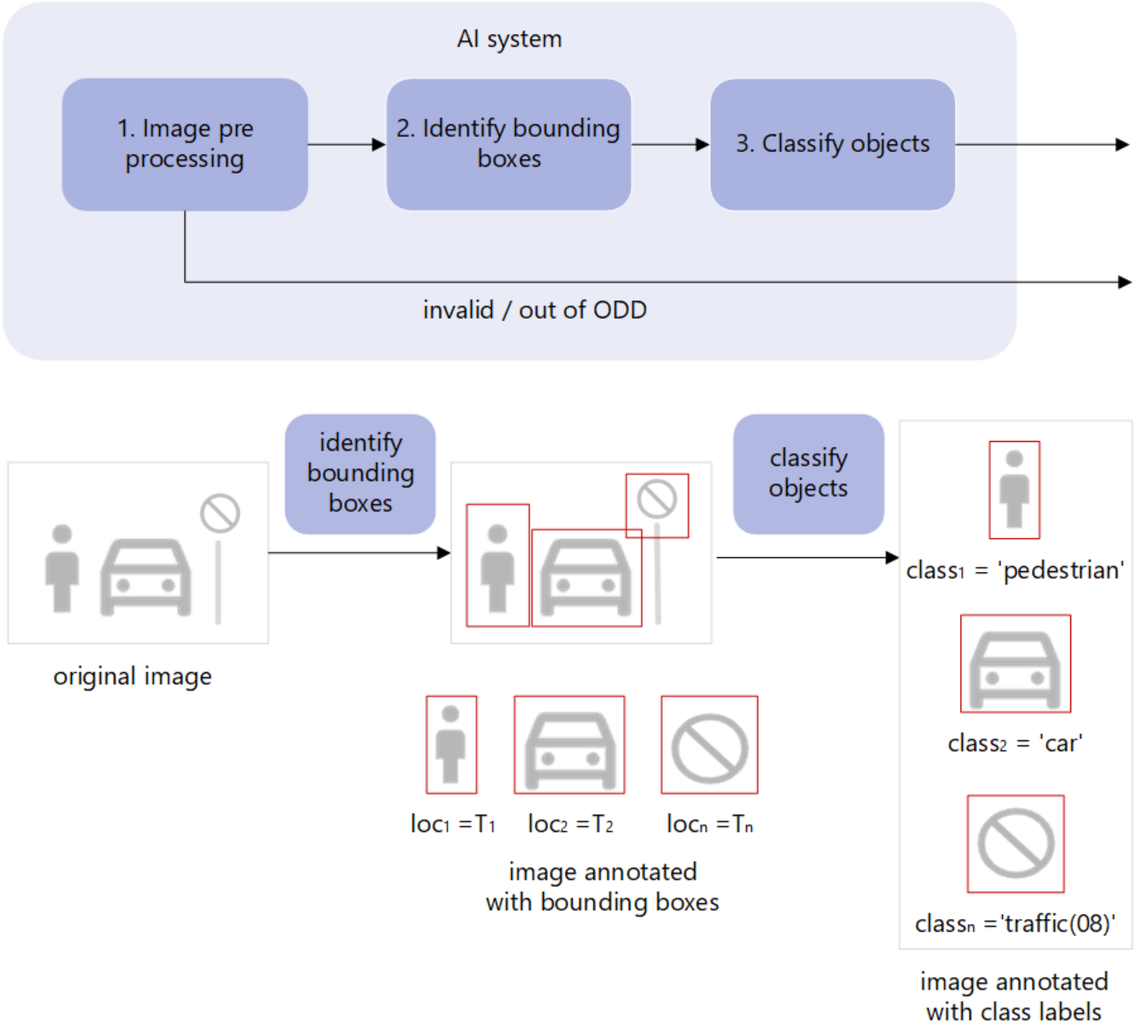
2. **Identify bounding boxes**

   The video frame is analysed and a vector li= (x1, x2, y1, y2) is created for each of the n objects detected where the x and y co-ordinates define the corners of the box.

3. **Classify objects**

   The contents of each bounding box are passed to an object classifier which allocates a label to the object. In this example, the label will include the set of types of traffic signs included in the ODD for the vehicle.

Figure 4 – Functionality of the AI System



The information produced by the object detector is then used, in conjunction with data from other information channels, to update an object database which is responsible for tracking objects of interest in the operational environment and estimating the location and trajectory of those objects.

Failures to identify an object's class (road sign type) and position correctly may lead to objects being missed, added erroneously, or placed in the wrong position within the scene. Each of these failures may give rise to a safety hazard.

The capacity of the object database is finite and therefore merging and discarding objects is necessary. Finally, the information in the object database, which represents the current belief about the state of the world, is passed to a decision framework which derives appropriate actions which are enacted by a set of actuating devices and their control electronics.

2.5 Allocated safety requirements: Work products

2.5.1 Input space definition

The definition of the input space for the AI system considers both the semantic and syntactic input space. The objective for the semantic input space definition is to define the scope of the intended operating environment within which the vehicle must operate safely. An ODD specification following the structure of BS ISO 34503[2] is used to define this semantic input space. Due to the complexity of the operating environment, the detail of the ODD specification is developed incrementally, informed by an understanding of the features of the environment that impact the AI system output.

The syntactic input space defines the data formats and relevant characteristics of the actual inputs to the AI system from the vehicle sensor inputs. This can include characteristics such as size and resolution of images. The nature of the syntactic input space depends upon design decisions for the encompassing system, so as for the semantic space is developed incrementally based upon the characteristics of the chosen sensor inputs.

2.5.2 Specification of the safety requirements allocated to the AI system

The AI system is used to identify traffic signs present within the operational ODD. It is important for the safe functioning of the vehicle that the type and position of each sign is correctly identified. For example, the vehicle needs to recognize the presence of stop signs in sufficient time to safely come to a halt, or correctly identify signs indicating a change in speed limit to safely decelerate/accelerate.

Safety requirements are identified through vehicle safety analysis activities performed in accordance with existing vehicle safety standards and guidance, such as BS ISO 26262 (series, in particular parts 2 to 6) and BS ISO 21448[3]. This analysis determines a number of safety requirements that are allocated to the ML component. In the case of safety requirements relating to stop signs, for example, an analysis is performed in line with BS ISO 26262 (series) to determine from a safety perspective how many missed stop signs are considered acceptable. This analysis results in the specification for this use case of a permissible number of stop sign misses per 1,000 miles of driving.

For the AI system, we are interested in the required probability of failing to detect any single stop sign, considering the number of stop signs that the vehicle is expected to encounter per 1,000 miles of driving.

[2] BS ISO 34503, *Road vehicles – Test scenarios for automated driving systems – Specification for operational design domain*

[3] BS ISO 21448, *Road vehicles – Safety of the intended functionality*

The input to the AI system is a series of image frames obtained from a video camera. This means that for each stop sign present, the AI system is presented with multiple frames in which the sign's position and orientation varies. The AI system outputs a detection result for each frame which updates or confirms the belief that a stop sign is present in the operating environment, as well as updating or confirming the sign's location with respect to the ego vehicle.

The decision component has multiple opportunities to determine the presence of each stop sign; similarly, the decision component can determine whether sufficient or insufficient evidence exists for action to be taken and if a decision should be taken or delayed. Based upon the frame rate of the input images, and the decision framework implemented by the decision component, the system architect can specify how many frames of detection are required, per stop sign instance, for the vehicle to make a sufficiently safe response.

Vehicle dynamics may also be considered to define how close the vehicle may be to the stop sign before it is detected. To be safe, this needs to allow enough time for the vehicle to stop comfortably without excessive braking. Determining this distance requires assumptions with regard to the maximum speed that the vehicle may be travelling, and the worst-case braking performance of the vehicle in adverse road conditions.

Based on an analysis of these factors, as part of the system safety process, the following safety requirements (SR) can be derived for the ML component:

Safety Requirement 1 (SR1): *The ML component shall correctly detect stop signs present on the planned path of the vehicle in their correct location in 95% of frames where a stop sign exists within 80 metres of the vehicle.*

False detection of stop signs is also a safety concern since a vehicle stopping inadvertently increases the potential for rear-end collisions[4]. In addition, any unpredictable behaviour from a vehicle can lead to unsafe behaviour in other vehicles, such as unnecessary and sudden overtaking. A false detection may arise if the ML component indicates a sign that is not actually present, or mis-classifies another object as a stop sign (such as a tree or an advertising board).

As for missed detections, specifying safety requirements related to the false detection of stop signs requires an analysis based upon assumptions regarding the vehicle systems and the operating environment; for example, a system may use data fusion to estimate the likelihood of a stop sign given the GPS location of the vehicle. This results in the specification of the following additional safety requirement for the ML component:

Safety Requirement 2 (SR2): *Where a stop sign does not exist at the location on the planned path of the vehicle, the ML component shall not detect a stop sign with a probability > 99% per frame.*

The position of a sign within a road network can be important in understanding to which vehicles the sign is applicable. In dense and complex road configurations, identifying a sign in an incorrect position could lead to an unnecessary response, and therefore potentially hazardous behaviour. Determining the relevance of visible signs requires an understanding of anticipated road geographies and sign orientations; for example, the incorrect placing of signs applying only to

[4] Leilabadi, S.H. and Schmidt, S. In-depth analysis of autonomous vehicle collisions in California. In: *2019 IEEE Intelligent Transportation Systems Conference (ITSC).* IEEE. October 2019, 889-893

vehicles exiting the motorway may cause vehicles to brake abruptly and thus compromise safety. This results in the specification of one more safety requirement for the ML component such as:

Safety Requirement 3 (SR3): *The AI system shall detect stop signs within 1.2 metres of their true position.*

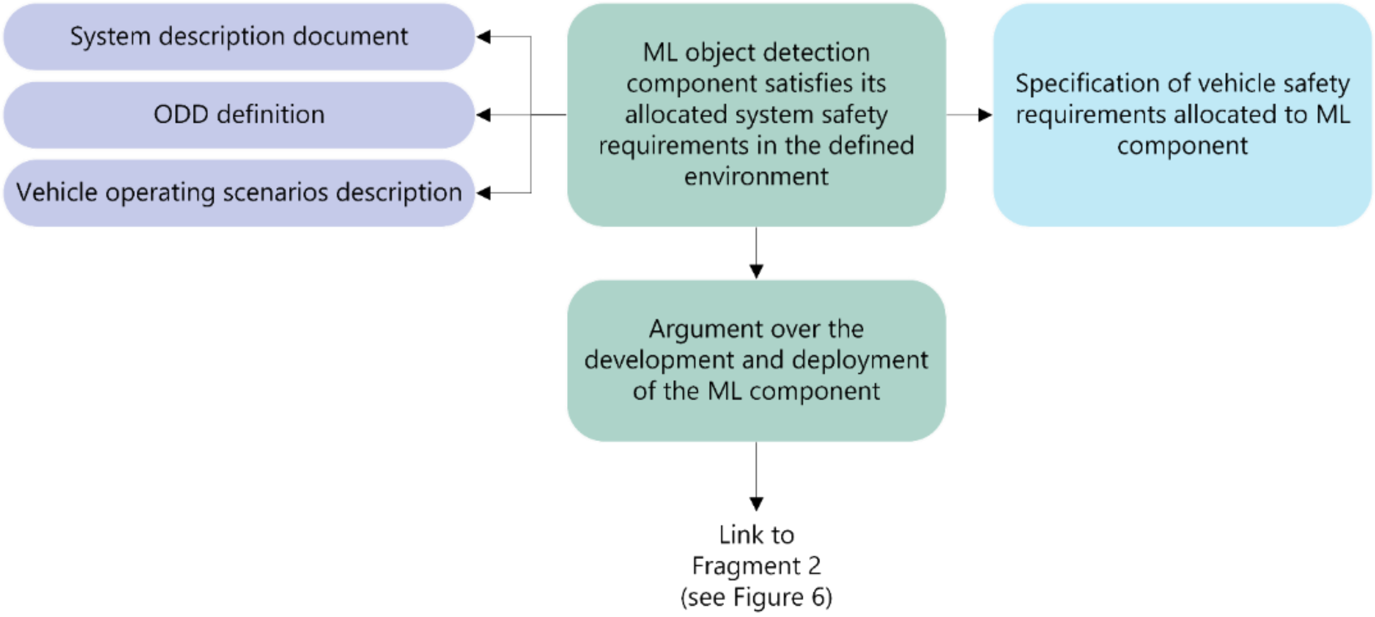
2.6 Iterative approach to refining the AI safety requirements and the input space definition

As a result of later activities, such as AI system design, test and safety analysis, the AI safety requirements and the input space definition may need to be further refined. In addition, the known insufficiencies of the AI system, including characteristics of the operating environment in which the insufficiencies are triggered, are systematically analyzed and reported to the encompassing system safety process, so that mitigating measures against these insufficiencies can be implemented.

2.7 Safety assurance argument development

Using the information described above, it is possible to start the development of the safety assurance argument for the ML component as shown in Figure 5. The other parts of the safety argument are developed in the rest of the use case. These argument fragments together form the complete safety assurance argument for the ML component.

Figure 5 – ML safety assurance argument: Fragment 1


System description document
ODD definition
Vehicle operating scenarios description
ML object detection component satisfies its allocated system safety requirements in the defined environment
Specification of vehicle safety requirements allocated to ML component
Argument over the development and deployment of the ML component
Link to Fragment 2 (see Figure 6)

3 AI system design

3.1 Objectives and requirements from PD ISO/PAS 8800

The objectives of the design phase are to:

- Select and justify appropriate AI technologies for use in the AI system.
- Identify appropriate architectural and development measures to fulfil the safety requirements prior to deployment.
- Identify appropriate architectural measures to mitigate residual functional insufficiencies of the AI system revealed after deployment.
- Identify measures for ensuring the safety requirements of the AI system are fulfilled within its target execution environment.

This involves fulfilling the following requirements.

- A justification shall be provided, that the selected AI technologies and AI methods are capable of fulfilling the AI safety requirements.
- AI safety requirements shall be allocated to AI components.
- Sufficient measures, such as architectural, development or a combination of both, shall be defined to ensure the AI safety requirements are fulfilled by the AI components.
- Sufficient measures, such as architectural, development or a combination of both, shall be defined to reduce the risk resulting from contributing AI errors of the AI components.
- The effectiveness of the chosen combination of architectural and development measures resulting from the two points above shall be supported by an argument.
- Safety analysis of the AI system outputs and, where reasonably practicable, of its architectural elements shall be performed to determine whether the safety requirements allocated to the AI system can be met.
- The differences between the development environment and the target execution environment shall be identified and evaluated regarding their potential impact on the safety requirements. If necessary, appropriate AI architectural and development measures shall be defined.
- AI components that are AI models or contain AI models shall be trained using the training dataset and evaluated using the validation dataset.

3.2 Work products

This phase of the AI safety lifecycle results in the following work products:

- AI component or AI system architecture.
- AI component or AI system development process.
- Implemented AI component.

3.3 Refinement and validation of ML safety requirements

The first step in assuring the design of the ML component is to refine the system safety requirements allocated to the AI system (see **2.5.2**) into ML-specific safety requirements (MLR) for the object detector. The ML safety requirements must be specified in a form suitable for use in the construction and verification of the ML component. This is achieved through specifying requirements on the performance and robustness of the ML component. In specifying the ML safety requirements, it is necessary that the system context and the operational scope defined by the ODD should be considered along with the allocated safety requirements.

The ML component returns a set of bounding boxes and associated class labels for each video frame it receives as input from the vehicle systems. Whilst many different measures may be used to assess the performance of the component, this use case considers the mean average precision (mAP) which incorporates measures of recall, precision and localization accuracy, and is understood as a composite measure for such components. In this way, the single metric combines a weighted assessment of the three system safety requirements (SR1, SR2 and SR3). It is important to note that defining such composite metrics requires an understanding of how the three measures in combination are reflected in the composite metric.

For example, sign localization in the ML component is not measured as a distance in metres, as in the real world, or a projection into pixel space, but as the intersection over union (IOU) which compares the ground truth bounding box for the stop sign with the predicted bounding box. The system safety requirements can be satisfied by selecting an appropriate limit on the mAP that represents an acceptable trade-off of the different measures.

Based on this assessment, ML safety requirements can be defined to specify the required performance and robustness of the ML component. A requirement for the performance of the component can be specified in terms of the mAP as:

***Machine Learning Requirement 1 (MLR1)**: The mean average precision for the ML component in detecting a stop sign shall be no less than 0.90.*

To validate MLR1 it is necessary to justify that the choice of the threshold on mAP satisfies SR1, SR2 and SR3 system safety requirements. This requires that distances from video images are calculated and used to show that an IOU that violates the mAP also violates the distance requirement in SR3.

To specify an ML robustness requirement, it is necessary to identify the conditions under which MLR1 must be satisfied by the ML component. This requires consideration of how the features of the operational environment of the system may vary within expected bounds (as defined in the ODD). Features of interest are defined through an examination of the operating environment as well as an understanding of the fragility of the ML approach in the presence of variabilities in the operating environment. Assessing the impact of environmental features on the robustness of the ML component may be undertaken through simulation and sample augmentation or in controlled, lab-based, studies. For this example, a simplified list of features and dimensions of variability are provided in Table 3 and a requirement for robustness is defined as:

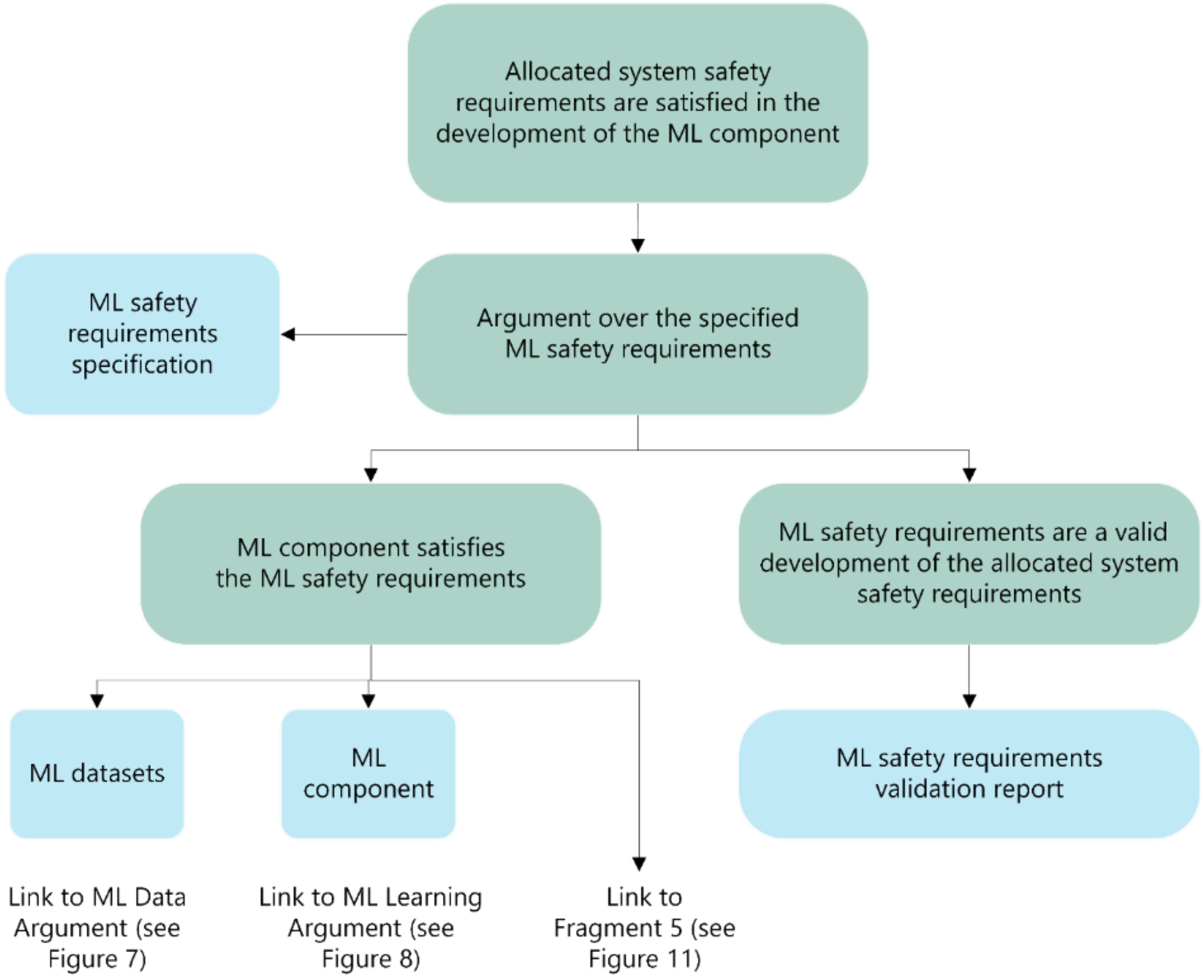
***Machine Learning Requirement 2 (MLR2)**: The mean average precision defined in MLR1 shall be satisfied over the features and ranges of variability defined in Table 3.*

Using the information described in this section it is possible to further develop the safety assurance argument for the ML component as shown in Figure 6.

Table 3 – Operating domain features and dimensions of variability for ML component robustness

Feature	Variability
Weather: Rain	(none, low, med, high)
Weather: Fog	(none, low, med, high)
Lighting: Glare angle	(none, 0, 30, 60)
Lighting: Ambient	(normal, bright, dark)
Environment: Obscuration	(none, 10%, 50%)
Environment: Damage	(none, minor, severe)

Figure 6 – ML safety assurance argument: Fragment 2


Allocated system safety requirements are satisfied in the development of the ML component
ML safety requirements specification
Argument over the specified ML safety requirements
ML component satisfies the ML safety requirements
ML safety requirements are a valid development of the allocated system safety requirements
ML datasets
ML component
ML safety requirements validation report
Link to ML Data Argument (see Figure 7)
Link to ML Learning Argument (see Figure 8)
Link to Fragment 5 (see Figure 11)

4 Data considerations

4.1 Objectives and requirements from PD ISO/PAS 8800

The objectives of the data lifecycle are to:

- Define the dataset lifecycle of activities related to the gathering, creation, analysis, verification and validation, management and maintenance of the datasets used in the development of the AI system.
- Identify the dataset insufficiencies that may impact the safety of the AI system.
- Identify the data-related safety properties that have a bearing on the safety of the AI system and that support dataset safety analysis.
- Define the countermeasures to prevent or mitigate dataset insufficiencies using dataset safety analysis methods at different steps in the dataset lifecycle.
- Define the data-related work products that support providing evidence of the safety of the AI system.

This involves fulfilling the following normative requirements:

- A dataset lifecycle shall be defined for the datasets used in the development of the AI system.
- The dataset lifecycle shall be defined such that it supports iterative development of the dataset, taking into account changes in the AI safety requirements and any insufficiencies observed during the AI system deployment phase.
- The dataset lifecycle shall include activities that relate to the gathering, creation, safety analysis, verification, validation, management and maintenance of the datasets used to develop the AI system.
- Data-related safety properties of the dataset shall be identified and be used as inputs at different phases of the dataset lifecycle.
- The dataset lifecycle activities shall include safety analyses to identify potential dataset insufficiencies, their root causes and their potential to cause a violation of AI safety requirements.
- Dataset requirements of the dataset shall:
    - Address the dataset insufficiencies that can lead to violation of the AI safety requirements; and
    - Specify countermeasures to prevent the dataset insufficiencies, to mitigate them, or both.
- Traceability shall be ensured between the dataset requirements and the AI safety requirements.

4.2 Work products

This phase of the AI safety lifecycle results in the following work products:

- Dataset lifecycle.
- Evidence for the outputs of the defined phases of the dataset lifecycle.

- Evidence for the safety analyses of the dataset.
- Dataset requirements specification.

4.3 Data requirements

To create an ML model that satisfies the defined ML safety requirements, the data used to develop the model needs to be considered. Firstly, data requirements (DR) are defined for the ML component. These data requirements represent an encoding of the two ML safety requirements (MLR1 and MLR2) defined in the ML safety requirements specification.

In specifying the data requirements, four key criteria are considered:

1) Data relevance.
2) Data accuracy.
3) Data balance.
4) Data completeness.

Data relevance refers to the extent to which the development data is representative of the operating environment and architecture of the vehicle and system into which the ML component is being deployed. The target operational domain of the vehicle, as captured in the scoping work products, is considered. This enables the specification of requirements for data relevance associated with both the system architecture and the operating environment.

Two example data relevance requirements for the ML component are:

Data Requirement 1 (DR1)*: All images used shall be sufficiently representative of images obtained by the video camera used by the vehicle in operation.*[5]

Data Requirement 2 (DR2)*: All road signs in images shall be road signs found on UK roads (as defined by the Traffic Signs Regulations and General Directions 2016*[6]*).*

The requirements on *data accuracy* for the ML component focus on the creation of labels and bounding boxes in the image data. In this example, labelling is carried out using a predominantly manual process and is therefore susceptible to human error where clarity or ambiguity exists. The labelling process therefore needs to be carefully managed, indeed the probability of errors in labelling increases as the size of the team increases and variations in practice occur.

An example data accuracy requirement for the ML component is:

Data Requirement 3 (DR3)*: All bounding boxes shall be specified such that the entirety of the sign is contained within the box irrespective of any obscuration*.

Data balance typically considers the number of samples for each class present in the data sets. Ideally all data sets used as part of model development are balanced, i.e. the same number of samples exist for every class of interest. In practice, however, those samples which are of particular interest in a safety context are found to be more difficult to obtain and “rare” in datasets which are gathered naturally during system operation. Therefore, requirements for data balance consider that

[5] This will include, for example ensuring that the images are representative of the type of camera used on the target vehicle, that the images reflect the position of the mounting of the camera on the vehicle etc.
[6] GREAT BRITAIN. Traffic Signs Regulations and General Directions 2016. London: The Stationery Office.

a sufficient number of rare data items are present to allow for models to be constructed which perform appropriately. It is noted here that it is not desirable for verification data to be balanced in the way that development data is, for reasons discussed in Section 5, where particular requirements on verification data sets are described.

An example data balance requirement for the ML component is:

Data Requirement 4 (DR4)*: There shall be an equal number of samples for each road sign defined in the Traffic Signs Regulations and General Directions 2016.*

While data balance considers the number of samples for each class, data completeness concerns how the collected data sets reflect the robustness requirements specified in the ML safety requirements. This considers the extent to which all of the identified features of concern in the operating domain are present in the data samples for all classes. For this ML component, data completeness requirements are specified by considering the dimensions of variability defined as part of the robustness requirement (MLR2).

An example data completeness requirement for the ML component is:

Data Requirement 5 (DR5)*: Data samples shall be obtained that represent each viable combination of features identified in Table 3.*

4.4 Creating and validating datasets

Having defined the data requirements, datasets are created to meet each requirement. Three distinct data sets are generated:

- Development data.
- Internal test data.
- Verification data.[7]

Generating these datasets involves following a defined data management process, including the collection, preprocessing and augmentation of the data. In this example, the data was collected from camera images obtained on roads in Germany, the complete set contains approximately 52,000 images with 43 classes of traffic signs. Twenty percent of images for each class are set aside for internal testing. The datasets are then validated to determine if the defined data requirements are met.

Examples of how each of the ML data requirements are validated is provided below:

- DR1: Although a different video camera is used on the target vehicle from that used when collecting the image data, the performance characteristics of the camera are comparable and the images obtained are of identical resolution.
- DR2: The images were obtained in Germany, so contain some road signs that are not applicable to vehicles operating on UK roads. This requires the datasets to be checked for images containing irrelevant road signs, which were then removed.
- DR3: All staff undertaking labelling activities are trained before commencing labelling work. In particular, they are trained on the need to individually label partially obscured signs. Random

[7] Verification is discussed further in Section 5.

sampling from the labelled sets is undertaken to validate the accuracy of the bounding boxes generated.

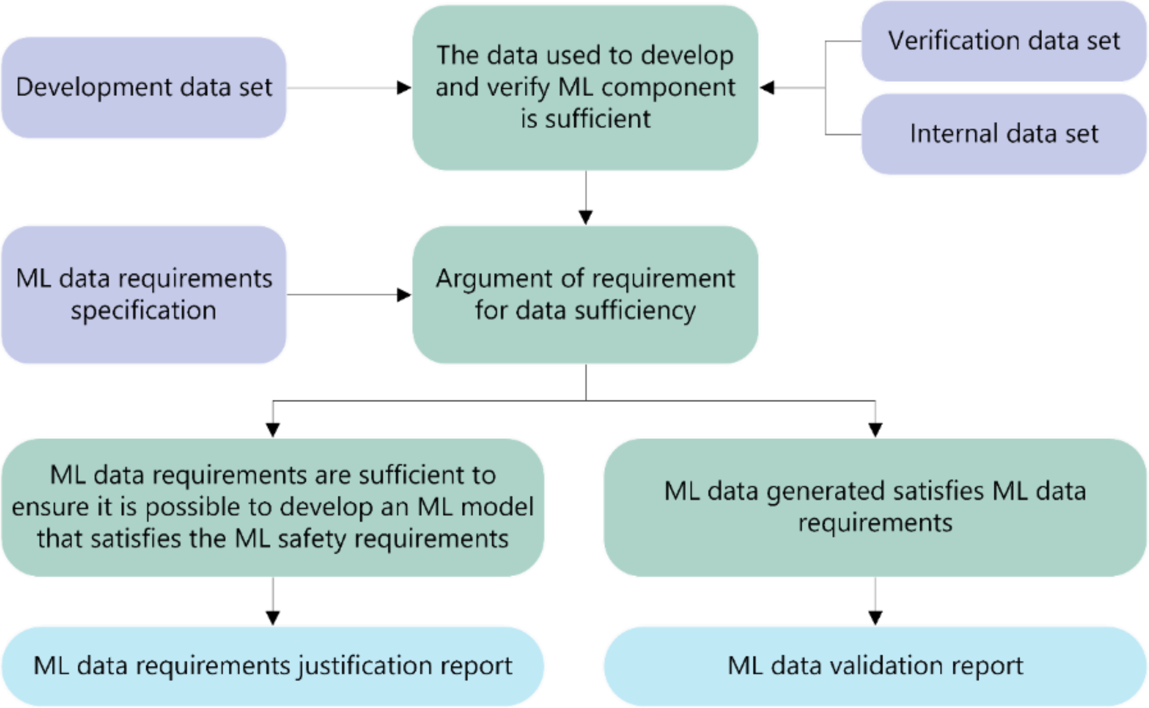
- DR4: Metadata for the datasets indicates that a set of classes is underrepresented in the dataset, i.e. caution signs related to specific types of wildlife. These classes are clustered into a single class to create a balanced dataset. In addition, there are many more speed restriction signs than necessary and therefore under-sampled from this class to maintain balance.
- DR5: The features of the operating environment defined in Table 3 is mapped to the datasets in Table 4. Analysis of the distribution of features shows that the number of samples with high levels of rain are insufficient. The dataset is therefore augmented with synthetic samples to address this shortfall. The augmentation is undertaken using a combination of simple mathematical models for fog and generative AI for glare and rain. Where a generative AI model is used, random sampling is employed to ensure that the resultant samples are relevant and accurate.

Table 4 – Data distribution by operating domain feature for this use case

Factor												
	Rain				Fog				Glare angle			
	1.1	1.2	1.3	1.4	2.1	2.2	2.3	2.4	3.1	3.2	3.3	3.4
1	✓	-	-	-	✓	-	-	-	✓	-	-	-
2	-	✓	-	-	-	✓	-	-	✓	-	-	-
3	-	✓	-	-	✓	-	-	-	-	-	-	✓

Using the information described in this section, it is possible to further develop the safety assurance argument for the ML component as shown in Figure 7.

Figure 7 - ML safety assurance argument: Fragment 3


Development data set
The data used to develop and verify ML component is sufficient
Verification data set
Internal data set
ML data requirements specification
Argument of requirement for data sufficiency
ML data requirements are sufficient to ensure it is possible to develop an ML model that satisfies the ML safety requirements
ML data generated satisfies ML data requirements
ML data requirements justification report
ML data validation report

4.5 AI component design and implementation

The development data is used to create the ML model. It is important that a development log is created and maintained to document the development decisions that are made and their rationale. In the example, a 'you only look once' (YOLO)[8] object detection model is used. This is a type of model that has been successfully used for perception in autonomous driving applications. Training an ML model is computationally expensive, so using existing models as a starting point and utilizing a transfer learning approach can significantly reduce the costs and effort required to obtain a suitable model.

Therefore, the development cycle for the model is started using a deep neural network structure from a system previously used in a perception pipeline for an autonomous truck. Although this ML component will be used as part of a different vehicle, the features present in the driving scenario are sufficiently similar for both vehicles. Therefore, the previously learnt model weights are sufficiently close to those to be learnt in the new context to allow for faster learning without compromising the ML safety requirements.

Object detectors such as YOLO are known to be difficult to generalize in open world contexts, such as the self-driving vehicle case being considered. Image mix-up[9] and learning rate scheduler[10]

[8] Redmon, J., Divvala, S., Girshick, R., and Farhadi, A. You only look once: Unified, real-time object detection. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE. December 2016, 779–788.
[9] Zhang, H., Cisse, M., Dauphin, Y.N., and Lopez-Paz D. mixup: Beyond empirical risk minimization. arXiv preprint arXiv:1710.09412.
[10] Zhang, Z., He, T., Zhang, H. Zhang, Z., Xie, J., and Li, M. Bag of freebies for training object detection neural networks. arXiv preprint arXiv:1902.04103.

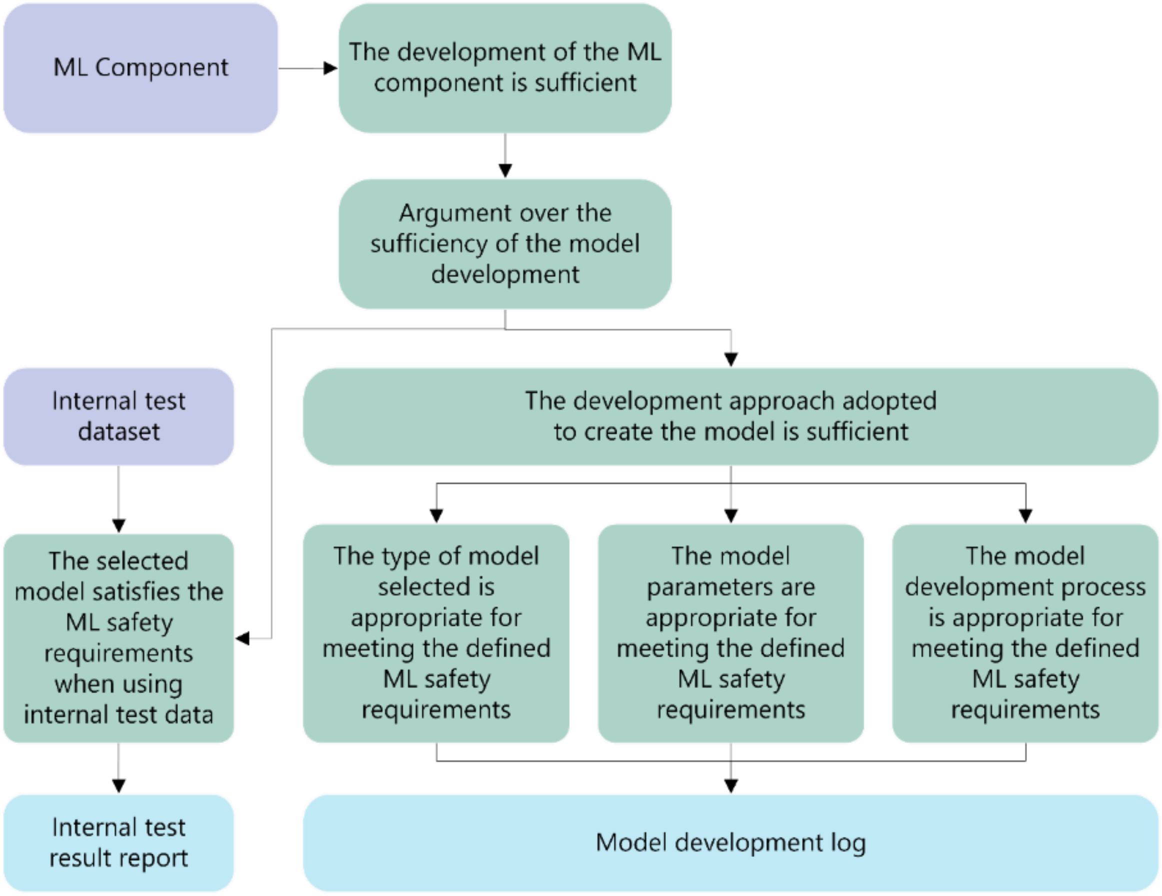
techniques are therefore applied to improve the performance of the model. The choices of model type and training approach choices are recorded in a model development log with a justification for selection and a record of the improvements in performance observed during training.

Using this approach, a large number of candidate models for the ML component are learnt and each of these models is distinct. The internal test data is applied to these candidate models and the models that satisfy ML safety requirements MLR1 (performance) and MLR2 (robustness) are identified.

To select the preferred model amongst these, the performance of the models, as mAP, is then plotted and a Pareto optimal model is selected. This is the model which provides the best trade-off between false positives and false negatives. Selecting the best trade-off requires consideration of the impact on vehicle performance if a stop sign is missed within a single video frame and when a stop sign is identified in error. This justification for model selection is recorded in the model development log.

Using the information described in this section, it is possible to further develop the safety assurance argument for the ML component as shown in Figure 8.

Figure 8 - ML safety assurance argument: Fragment 4


ML Component
The development of the ML component is sufficient
Argument over the sufficiency of the model development
Internal test dataset
The development approach adopted to create the model is sufficient
The selected model satisfies the ML safety requirements when using internal test data
The type of model selected is appropriate for meeting the defined ML safety requirements
The model parameters are appropriate for meeting the defined ML safety requirements
The model development process is appropriate for meeting the defined ML safety requirements
Internal test result report
Model development log

5 Verification and validation of the AI system

5.1 Objectives and requirements from PD ISO/PAS 8800

The objectives of this phase of the AI safety lifecycle are to:

- Verify that the AI system fulfils its AI safety requirements.
- Validate that the safety requirements allocated to the AI system are achieved when integrating into the encompassing system.

This involves fulfilling the following requirements.

- The AI system shall be verified to provide evidence for:
    - Conformity to the AI safety requirements; and
    - Confidence in the absence of unintended functionality and properties.
- Testing of an AI system shall be performed on the AI components that can be tested stand-alone and on the integrated AI system.
- Test cases for the verification of the AI components shall be derived using best practices for test case derivation. This includes using an appropriate combination of the methods listed in BS ISO 26262-6:2018[11], Clause 9, i.e. analysis of the requirements, generation and analysis of equivalent classes, analysis of boundary values and error guessing based on knowledge or experience.
- Each test case of an AI component shall include pass/fail criteria.
- Test cases of an AI component shall adequately verify the AI safety requirements allocated to the AI component within the specified input space of the AI system.
- The AI system integration approach shall specify the steps for integrating the individual AI components hierarchically into higher level AI components until the AI system is fully integrated.
- The AI system integration shall be verified to provide evidence that the hierarchically integrated AI components and the integrated AI system achieve:
    - Conformity to the AI system architectural design in accordance with Clause 10 of PD ISO/PAS 8800; and
    - Satisfaction of the AI safety requirements.
- AI system safety validation shall confirm that the safety requirements allocated to the AI system are fulfilled when the AI system is integrated into the encompassing system.

5.2 Verification and validation: Work products

This phase of the AI safety lifecycle results in the following work products:

- AI system verification report.
- Integrated AI system.

[11] BS ISO 26262-6:2018, *Road vehicles – Functional safety – Part 6: Product development at the software level*

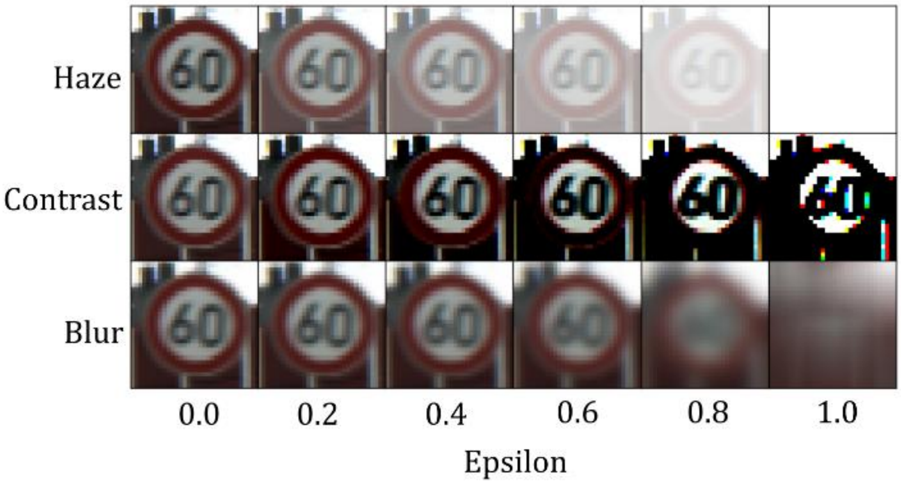
- AI system validation report.

5.3 AI component verification

Verification of the selected model is undertaken using the verification dataset. It is important that verification of the ML component is independent of the model development, so for instance, the verification dataset is created by people who are not involved in the development activities.

It was ensured that the verification data contained additional images for environment features at the extremes of the defined variability ranges. As a result of discussions with domain experts, images are also added that contain combinations of features that are seen to be particularly challenging for human drivers, or for previously deployed autonomous systems, e.g. fog combined with obscured images in low light. Image augmentation is used where it is not possible to obtain real-world images representing such cases. The image augmentation alters the existing data samples to provide the required features synthetically as shown in Figure 9.

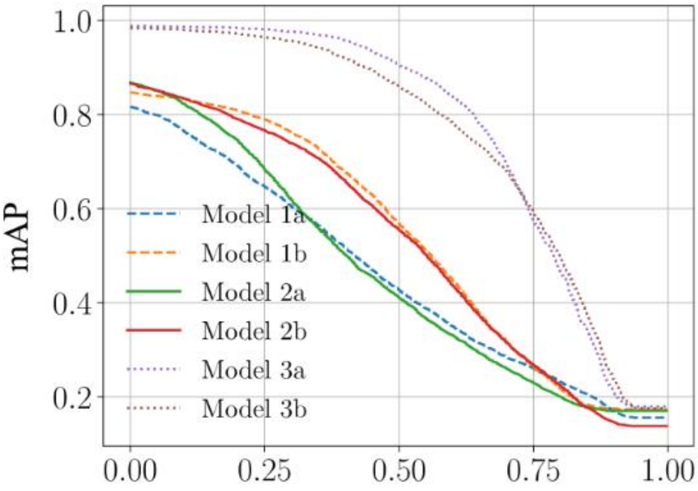
Figure 9 – Examples of image augmentation used to create verification data items (Source: University of York)



Given the sensing technology used on the target vehicle, fog is seen to be particularly challenging for the ML component. An important part of the verification is therefore to demonstrate that the ML component is no less capable of spotting stop signs in the presence of fog than a competent human driver (satisfying robustness requirement MLR2). To check model robustness, tests are undertaken to identify the level of perturbation in the identified features of the operating domain at which the model fails to satisfy the ML safety requirements. This is achieved by using the DeepCert12 approach, which provides test and formal verification evidence. It can be seen in Figure 10 that the required mAP of 0.9 is shown to be achieved for the selected model (Model 3b) up to 0.5 perturbation on haze13.

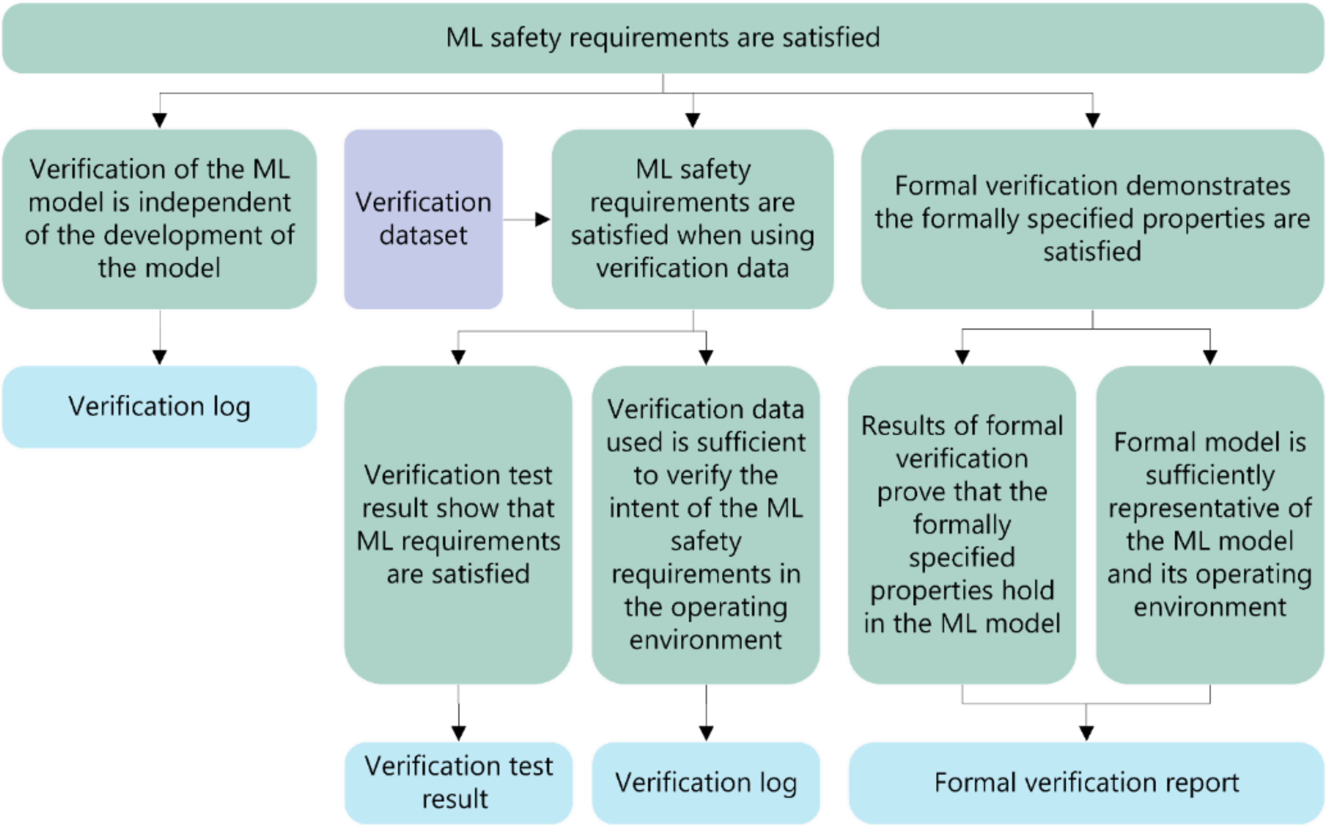
[12] Paterson, C., Wu, H., Grese, J., Calinescu, R., Pasareanu, C.S., and Barrett, C. Deepcert: Verification of contextually relevant robustness for neural network image classifiers. In: *International Conference on Computer Safety, Reliability, and Security*. Springer. September 2021, 3–17.
[13] Image haze was used to represent the effect of fog in augmented images.

Figure 10 – Model robustness with respect to haze perturbation (Source: University of York)



Using the information described above, it is possible to further develop the safety assurance argument for the ML component as shown in Figure 11.

Figure 11 – ML safety assurance argument: Fragment 5


ML safety requirements are satisfied
Verification of the ML model is independent of the development of the model
Verification dataset
ML safety requirements are satisfied when using verification data
Formal verification demonstrates the formally specified properties are satisfied
Verification log
Verification test result show that ML requirements are satisfied
Verification data used is sufficient to verify the intent of the ML safety requirements in the operating environment
Results of formal verification prove that the formally specified properties hold in the ML model
Formal model is sufficiently representative of the ML model and its operating environment
Verification test result
Verification log
Formal verification report

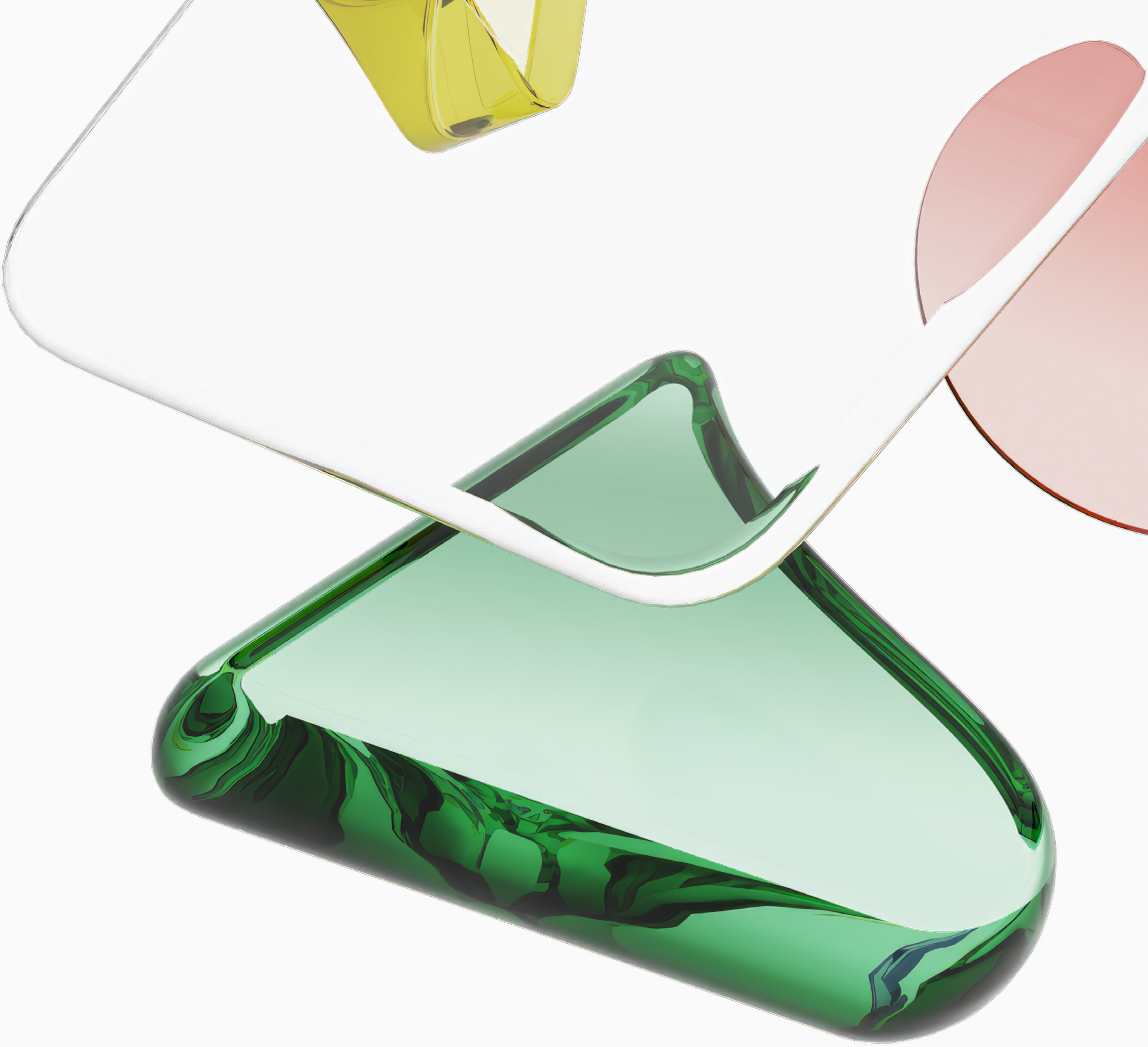
6 Summary

The use case presented in this document provides an example of how the framework recommended in PD ISO/PAS 8800 can be applied to an ML component of an automotive AI system. The use case illustrates a single iteration of the AI system safety assurance lifecycle. It describes the safety assurance activities performed throughout the stages of the AI system lifecycle, covering:

- safety requirement derivation, refinement and validation;
- data management
- design and implementation; and
- verification.

It is shown how the artefacts that were generated from performing each of the safety assurance activities are used to create a compelling safety assurance argument for the ML object detection component as part of that AI system.

For a full development, the process followed would be expected to be highly iterative at each stage, with consideration given to how the findings of each activity may necessitate re-visiting previous activities and updating work products. The findings from verification may, for example, result in a consideration of the need for additional refined requirements, additional design measures, additional data measures or a renegotiation of the requirements allocated to the AI system from the system safety process (e.g. that the system should be required to only operate up to a certain level of fog).



BSI Group
389 Chiswick High Road
London, W4 4AL
United Kingdom
+44 345 080 9000
bsigroup.com

Your partner
in progress