CodeScene

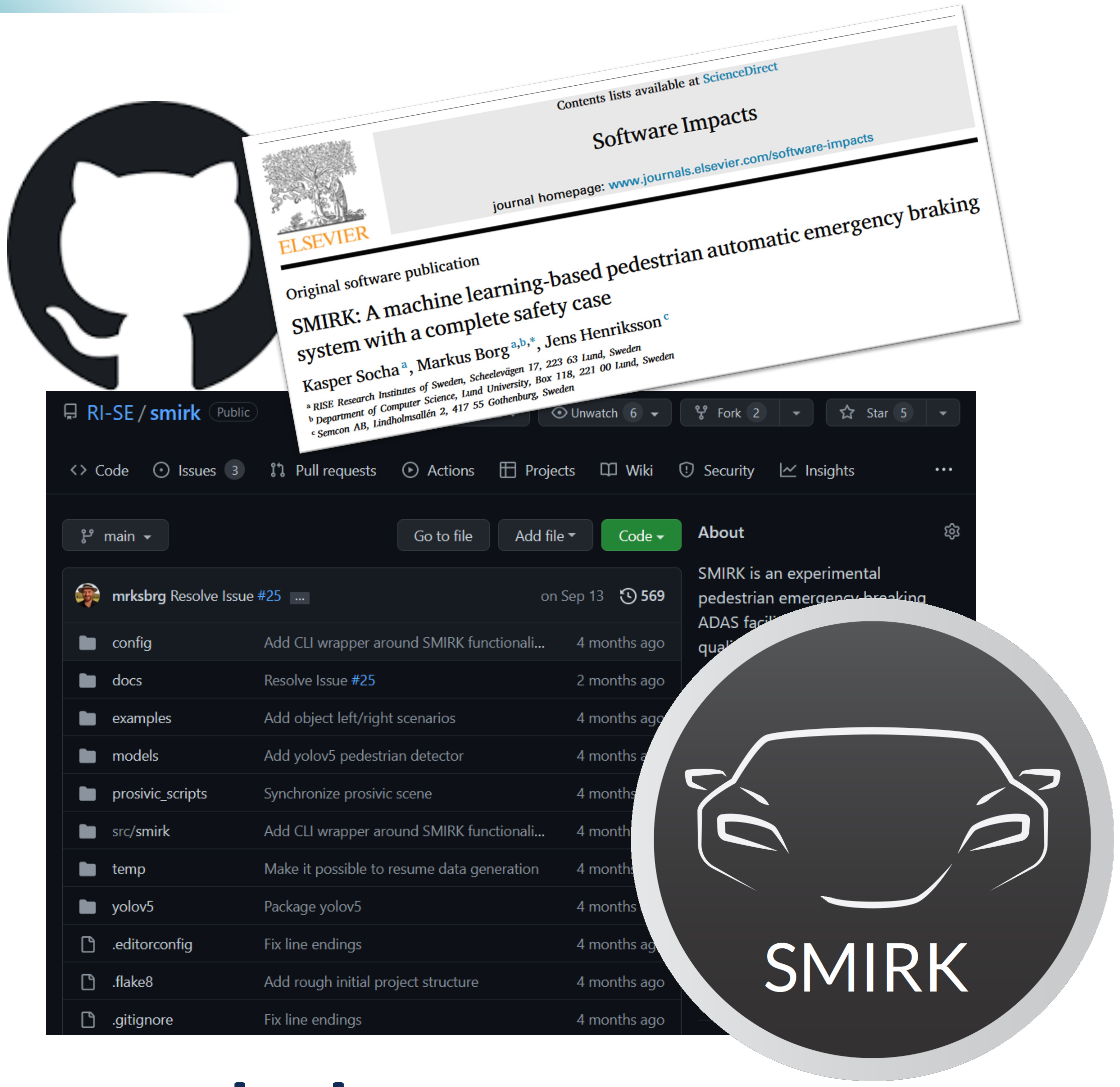
Ergo, SMIRK is Safe: A Safety Case for a Machine Learning Component in a Pedestrian Emergency Brake System

Markus Borg

markus.borg@codescene.com

10th Scandinavian Conference on System & Software Safety, Nov 22, 2022

Open ML safety case



Open ML-based demonstrator

CodeScene

Introduction

Who is Markus?

Development engineer, ABB 2007-2010

- Process automation

PhD student, Lund University 2010-2015

- Traceability, change impact analysis

Senior researcher, RISE 2015-2022

- AI engineering and functional safety

Principal researcher, CodeScene 2022-

- Software engineering intelligence

LUND UNIVERSITY

RISE Research Institutes of Sweden

CodeScene

CodeScene

codescene.com

Markus Borg

Mashid Helali

RI.SE

Kasper Socha

Thanh Bui

Piotr Tomaszewski

semcon

Jens Henriksson

QRTECH
an EMBRON Company

Sankar Sathyamoorthy

Olof Lennartsson

INFOTIV

Elias Sonnsjö Lönegren

COMBITECH

Sebastian Brink

Standards and guidelines are high-level…

… must get our hands dirty with ML details

Lack of:

- experience reports
- open demonstrator systems

"How to demonstrate and share a complete ML safety case for an open ADAS?"

CodeScene

Two teasers! Development of

1. SMIRK
2. Safety case

Contents lists available at ScienceDirect

Software Impacts

journal homepage: www.journals.elsevier.com/software-impacts

ELSEVIER

Original software publication

SMIRK: A machine learning-based pedestrian automatic emergency braking system with a complete safety case

Kasper Socha [a], Markus Borg [a,b,*], Jens Henriksson [c]

[a] RISE Research Institutes of Sweden, Scheelevägen 17, 223 63 Lund, Sweden
[b] Department of Computer Science, Lund University, Box 118, 221 00 Lund, Sweden
[c] Semcon AB, Lindholmsallén 2, 417 55 Gothenburg, Sweden

ARTICLE INFO

Keywords:
Automotive demonstrator
Advanced driver-assistance system
Pedestrian automatic emergency braking
Machine learning
Computer vision
Safety case

ABSTRACT

SMIRK is a pedestrian automatic emergency braking system that facilitates research on safety-critical systems embedding machine learning components. As a fully transparent driver-assistance system, SMIRK can support future research on trustworthy AI systems, e.g., verification & validation, requirements engineering, and testing. SMIRK is implemented for the simulator ESI Pro-SiVIC with core components including a radar sensor, a mono camera, a YOLOv5 model, and an anomaly detector. ISO/PAS 21448 SOTIF guided the development, and we present a complete safety case for a restricted ODD using the AMLAS methodology. Finally, all training data used to train the perception system is publicly available.

arXiv > cs > arXiv:2204.07874

Computer Science > Software Engineering

[Submitted on 16 Apr 2022 (v1), last revised 15 Sep 2022 (this version, v2)]

Ergo, SMIRK is Safe: A Safety Case for a Machine Learning Component in a Pedestrian Automatic Emergency Brake System

Markus Borg, Jens Henriksson, Kasper Socha, Olof Lennartsson, Elias Sonnsjö Lönegren, Thanh Bui, Piotr Tomaszewski, Sankar Raman Sathyamoorthy, Sebastian Brink, Mahshid Helali Moghadam

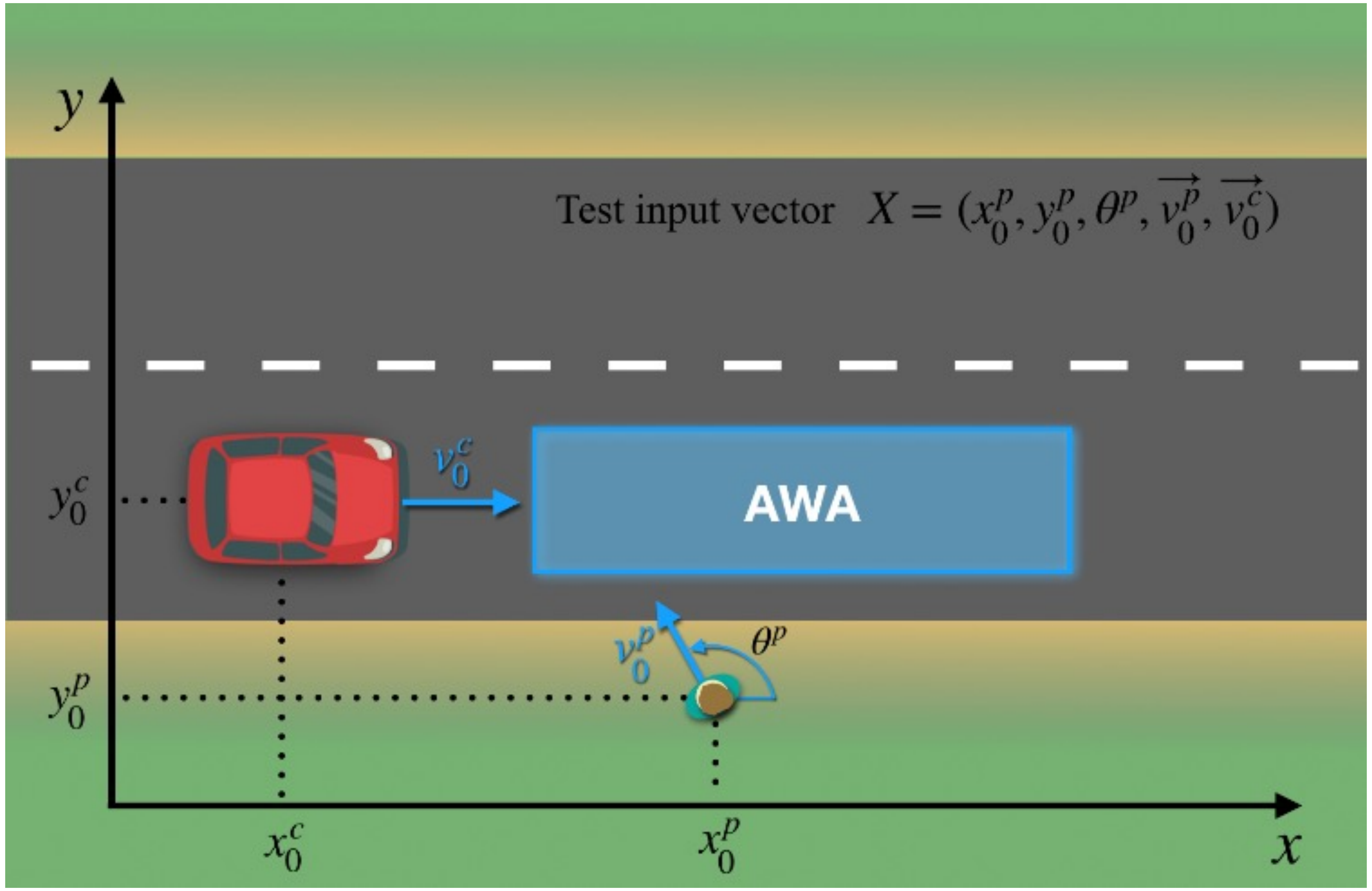
Integration of Machine Learning (ML) components in critical applications introduces novel challenges for software certification and verification. New safety standards and technical guidelines are under development to support the safety of ML-based systems, e.g., ISO 21448 SOTIF for the automotive domain and the Assurance of Machine Learning for use in Autonomous Systems (AMLAS) framework. SOTIF and AMLAS provide high-level guidance but the details must be chiseled out for each specific case. We initiated a research project with the goal to demonstrate a complete safety case for an ML component in an open automotive system. This paper reports results from an industry-academia collaboration on safety assurance of SMIRK, an ML-based pedestrian automatic emergency braking demonstrator running in an industry-grade simulator. We demonstrate an application of AMLAS on SMIRK for a minimalistic operational design domain, i.e., we share a complete safety case for its integrated ML-based component. Finally, we report lessons learned and provide both SMIRK and the safety case under an open-source licence for the research community to reuse.

CodeScene

Development of SMIRK

SMIRK

Reverse engineering from PeVi



RESEARCH-ARTICLE

Testing advanced driver assistance systems using multi-objective search and neural networks

Authors: Raja Ben Abdessalem, Shiva Nejati, Lionel C. Briand, Thomas Stifter Authors Info & Affiliations

Publication: ASE 2016: Proceedings of the 31st IEEE/ACM International Conference on Automated Software Engineering • August 2016 • Pages 63–74 • https://doi.org/10.1145/2970276.2970311

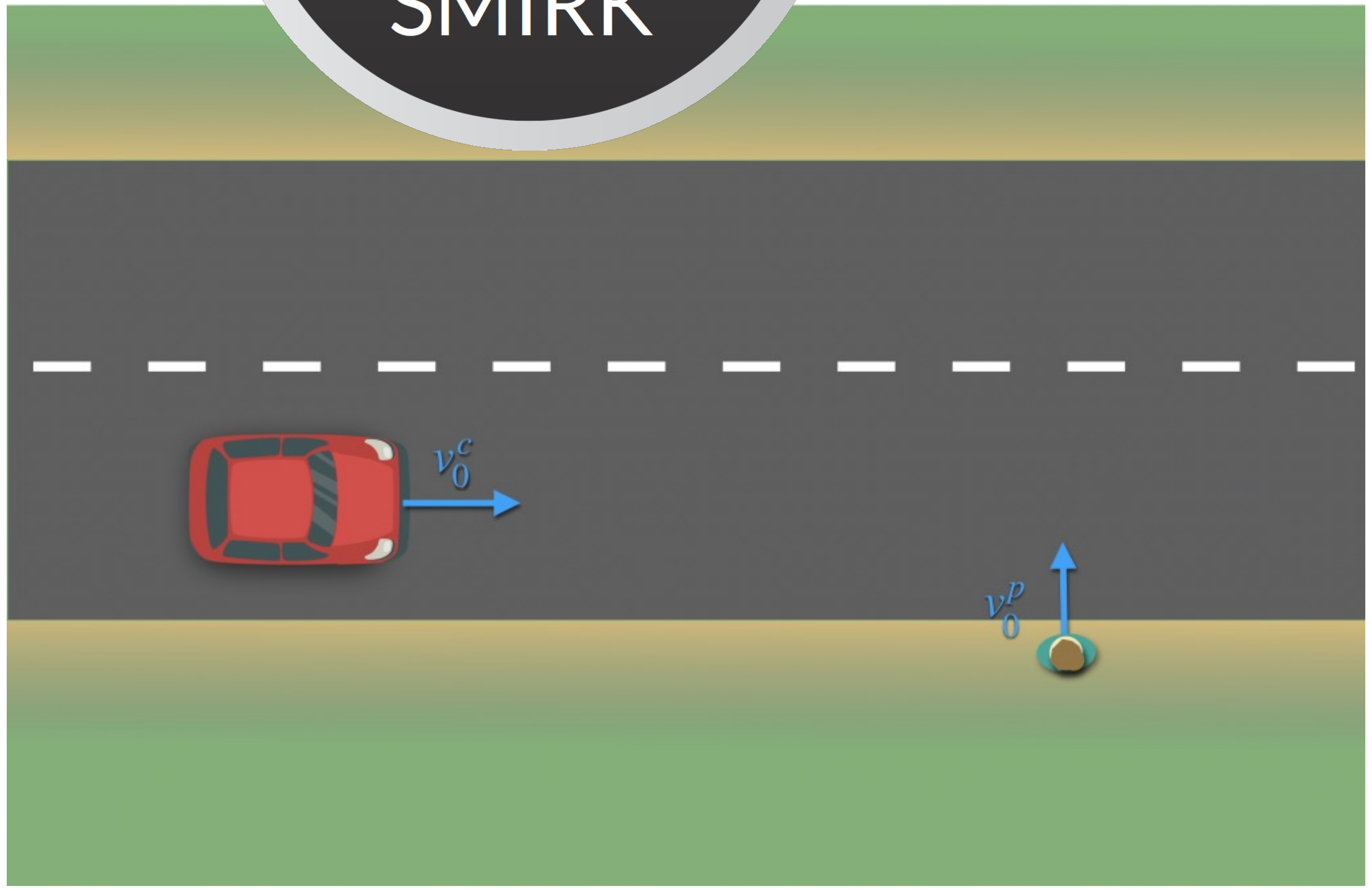
UNIVERSITÉ DU LUXEMBOURG

a sense for innovation IEE

Open Source ADAS MVP

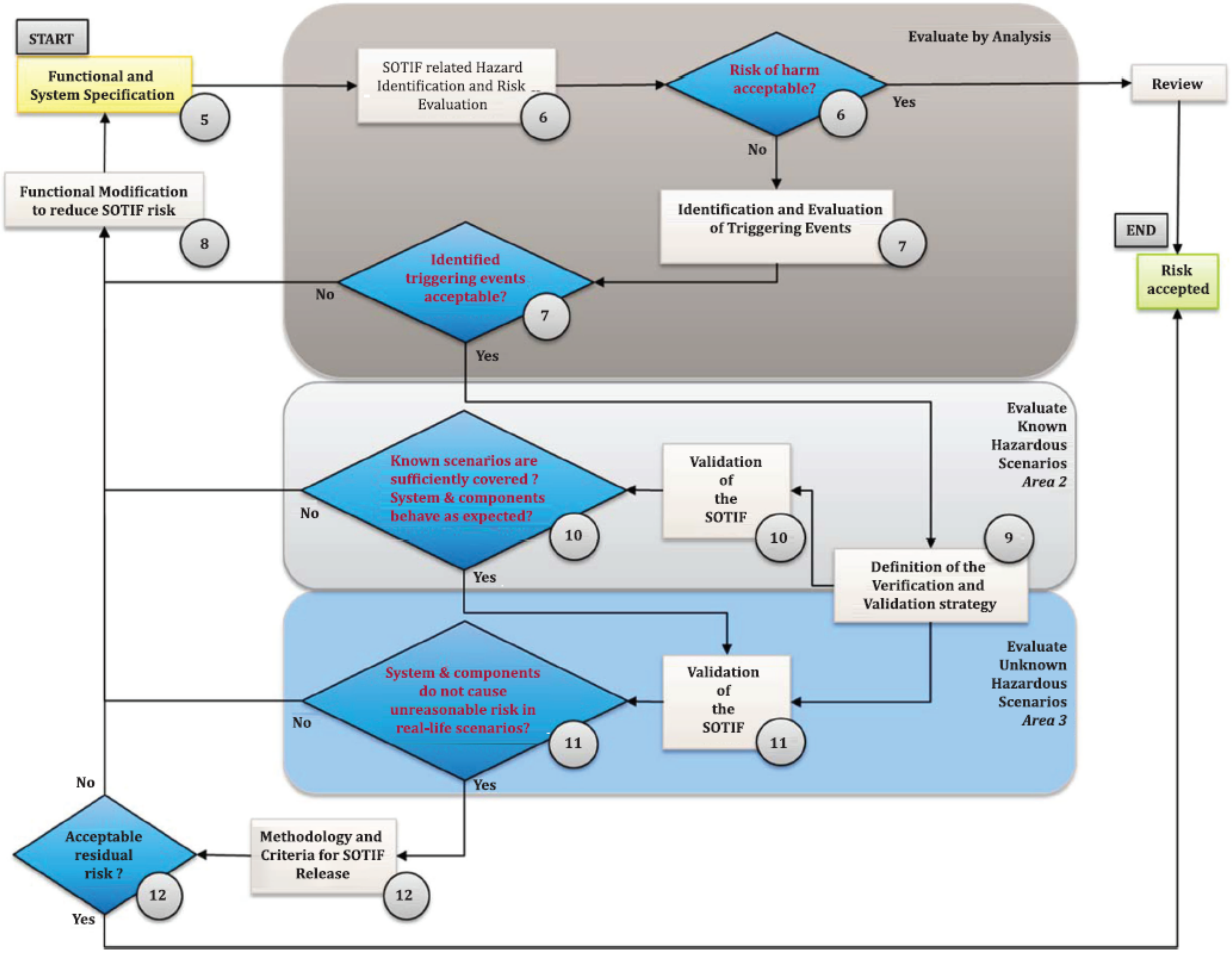
- In ESI Pro-SiVIC
- Pedestrian emergency braking
- Mono-camera and radar
- ML-based pedestrian recognition

SMIRK

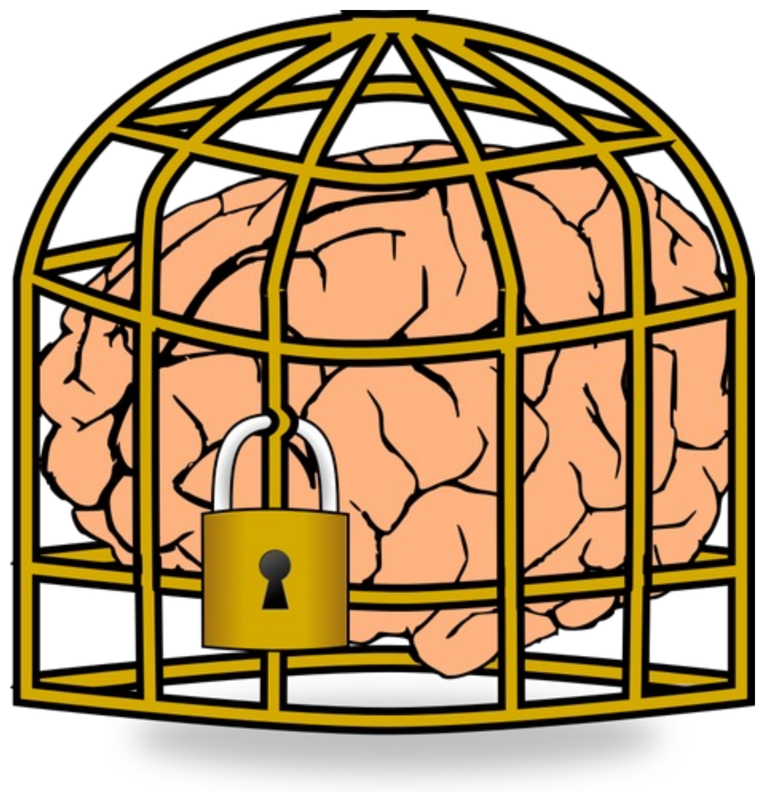


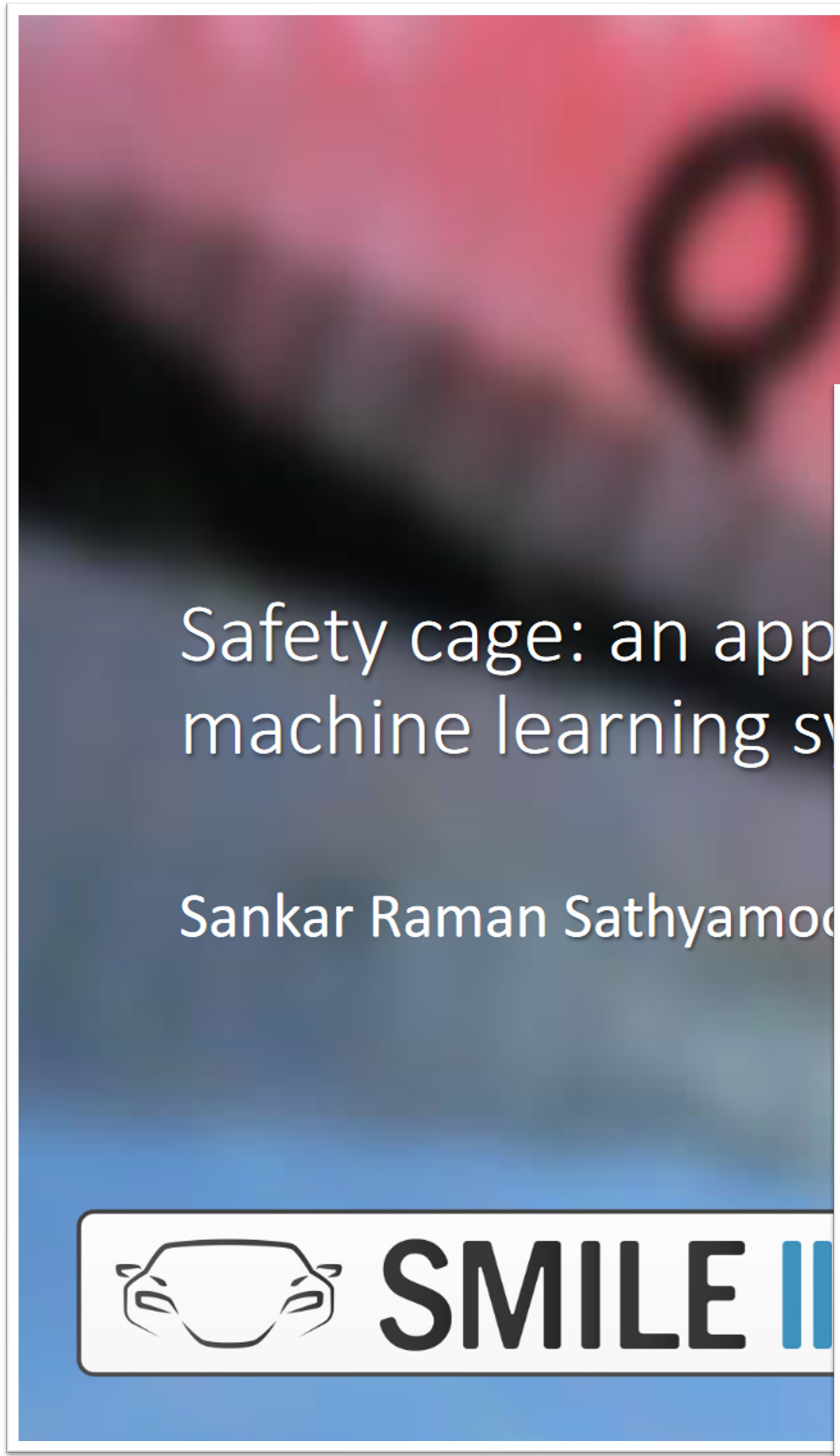
github.com/RI-SE/smirk

Follow the process in ISO 21448 SOTIF



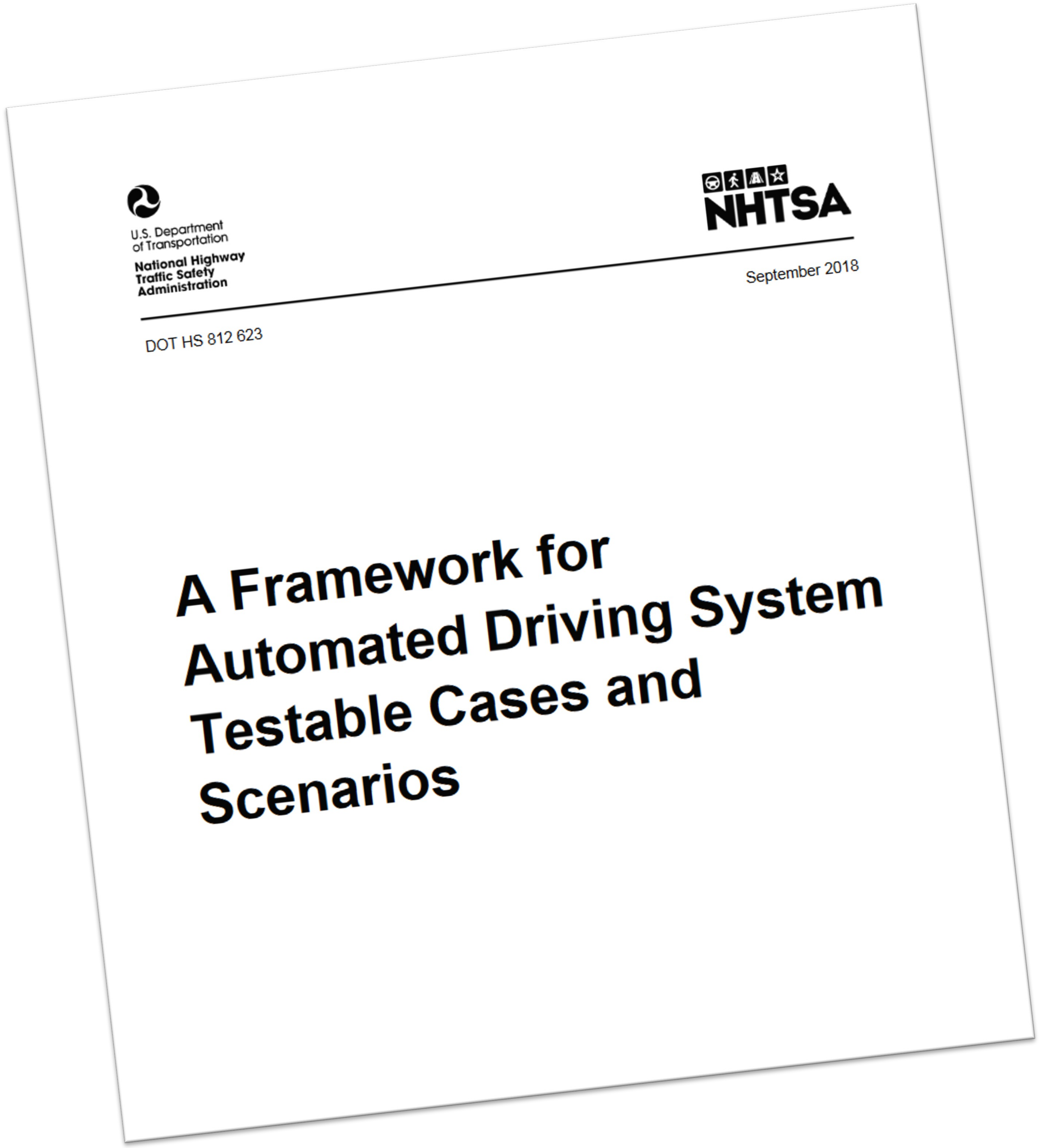
Primary hazard to tackle:
False postives




QRTECH
INNOVATIVE ENGINEERING
Safety cage: an app
machine learning sy
Sankar Raman Sathyamo
SMILE II


SMILE II – Use cases

MVP Operational Design Domain


U.S. Department of Transportation
National Highway Traffic Safety Administration

NHTSA

September 2018

DOT HS 812 623

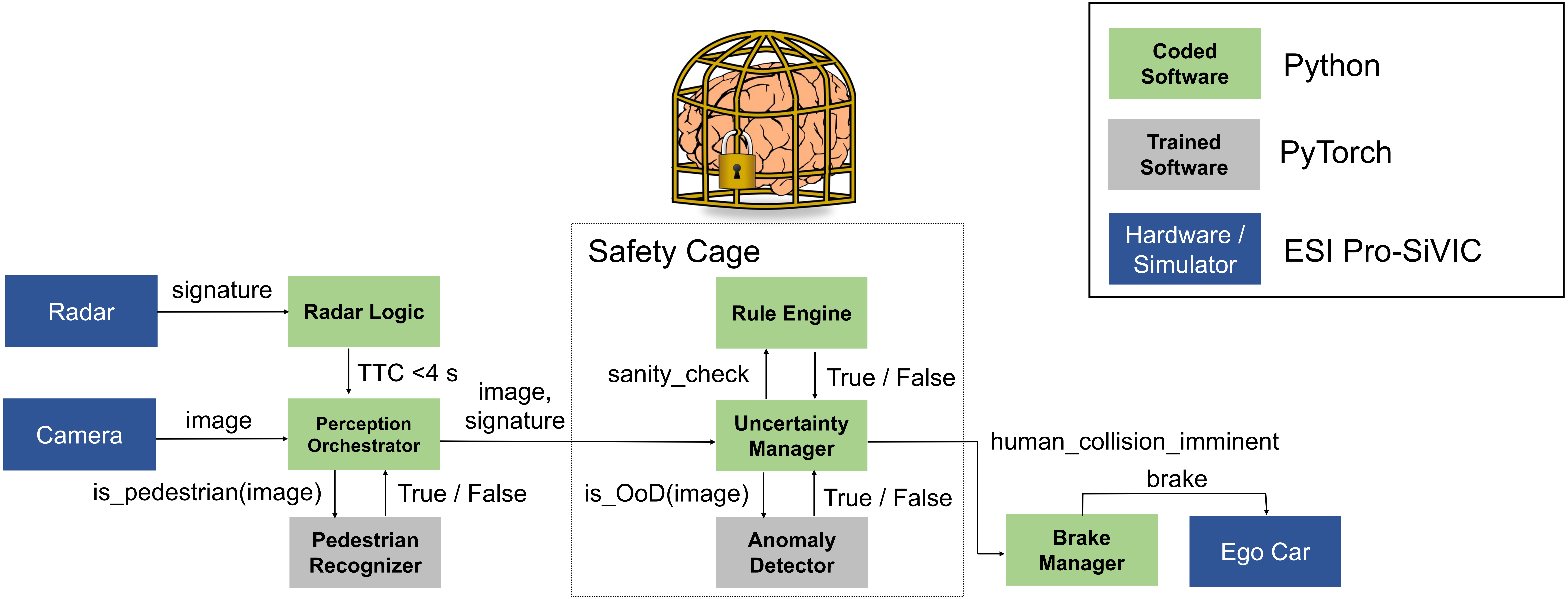
A Framework for Automated Driving System Testable Cases and Scenarios

Thorn E, Kimmel SC, Chaka M, et al (2018)

Tech. rep., US Department of Transportation
National Highway Traffic Safety Administration


ODD Elements
Physical Infrastructure
Roadway Types
highway, urban, rural, parking, multi-lane, single lane, on/off ramps, intersections, roundabouts, ...
Roadway Surfaces
asphalt, concrete, mixed, grating, brick, dirt, gravel, ...
Roadway Edges
line markers, temporary line markers, shoulder, concrete barriers, rails, cones, ...
Operational Constraints
Objects
Straight rural road, good conditions, single pedestrian
Environmental Conditions
Weather
wind, rain, snow, sleet, temperature, ...
Weather-Induced Roadway Conditions
standing water, icy roads, snow on road, ...
Particulate Matter
fog, smoke, smog, dust, ...
Illumination
day, sun front-lighting, dawn, dusk, night, street lights, headlights, oncoming vehicle lights, ...

Logical View of the SMIRK Architecture



Requirements engineering...

System requirements

3.3 Machine Learning Safety Requirements [H]

This section refines SYS-SAF-REQ into two separate requirements corresponding to false positives and false negatives, respectively.

- **SYS-ML-REQ1: The pedestrian recognition component shall detect pedestrians if the radar tracking component returns TTC < 4s for the corresponding object.**
- **SYS-ML-REQ2: The pedestrian recognition component shall reject input that does not resemble the training data.**

3.3.1 Performance Requirements

This section specifies performance requirements corresponding to the ML safety requirements with a focus on quantitative targets for the pedestrian recognition component. All requirements below are restricted to pedestrians on or close to the road.

- **SYS-PER-REQ1: The pedestrian recognition component shall identify pedestrians with an accuracy of 0.93 when they are within 50 meters.**
- **SYS-PER-REQ2: The false negative rate of the pedestrian recognition component shall not exceed 7% for pedestrians when they are detected by the radar tracking component within 50 meters.**
- **SYS-PER-REQ3: The false positive rate of the pedestrian recognition component shall not exceed 0.01% for objects detected by the radar tracking component with a TTC < 4s**
- **SYS-PER-REQ4: In a sequence of images from a video feed any pedestrian to be detected shall not be missed in more than 1 out of 5 frames.**
- **SYS-PER-REQ5: The pedestrian recognition component shall determine the position of pedestrians within 50 cm of their actual position.**
- **SYS-PER-REQ6: The pedestrian recognition component shall allow an inference speed of at least 10 FPS on the target platform.**

PRECOG

Data requirements

2.1 Relevant

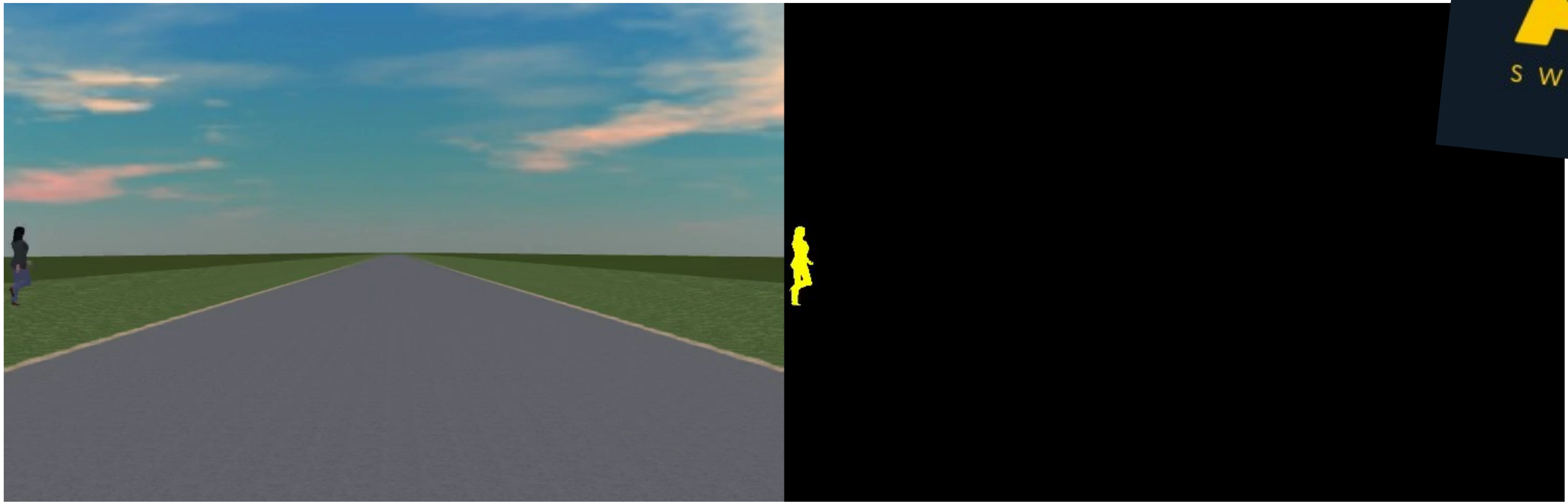
This desideratum considers the intersection between the dataset and the supported dynamic driving task in the ODD. The SMIRK training data will not cover operational environments that are outside of the ODD, e.g., images collected in heavy snowfall.

- DAT-REL-REQ1: All data samples shall represent images of a road from the perspective of a vehicle.
- DAT-REL-REQ2: The format of each data sample shall be representative of that which is captured using sensors deployed on the ego vehicle.
- DAT-REL-REQ3: Each data sample shall assume sensor positioning representative of the positioning used on the ego vehicle.
- DAT-REL-REQ4: All data samples shall represent images of a road that corresponds to the ODD.
- DAT-REL-REQ5: All data samples containing pedestrians shall include one single pedestrian.
- DAT-REL-REQ6: Pedestrians included in data samples shall be of a type that may appear in the ODD.
- DAT-REL-REQ7: All data samples representing non-pedestrian OOD objects shall be of a type that may appear in the ODD.

CodeScene

Generate Training Data in ESI Pro-SiVIC

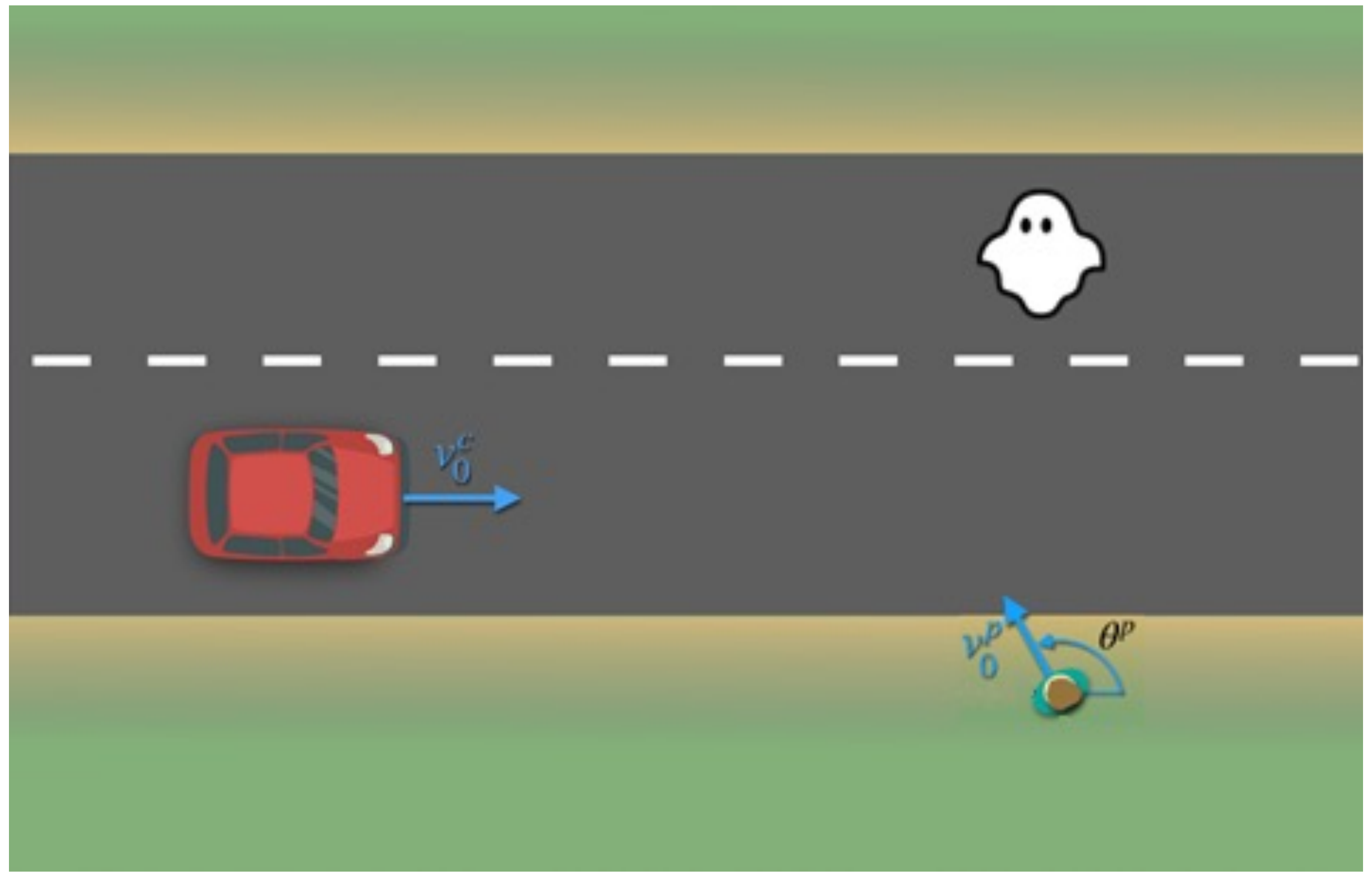
Synthetic data that cover the Operational Design Domain

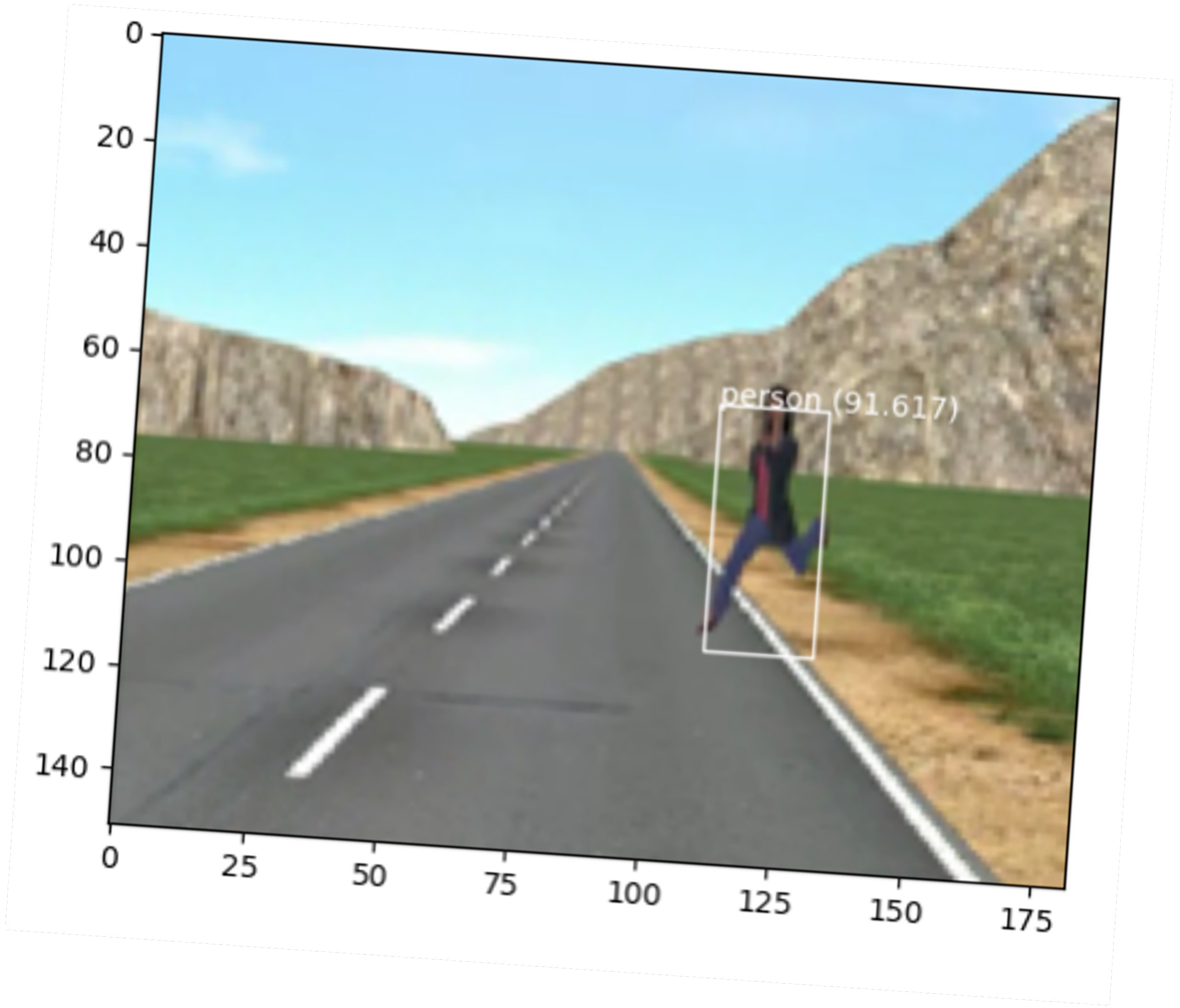


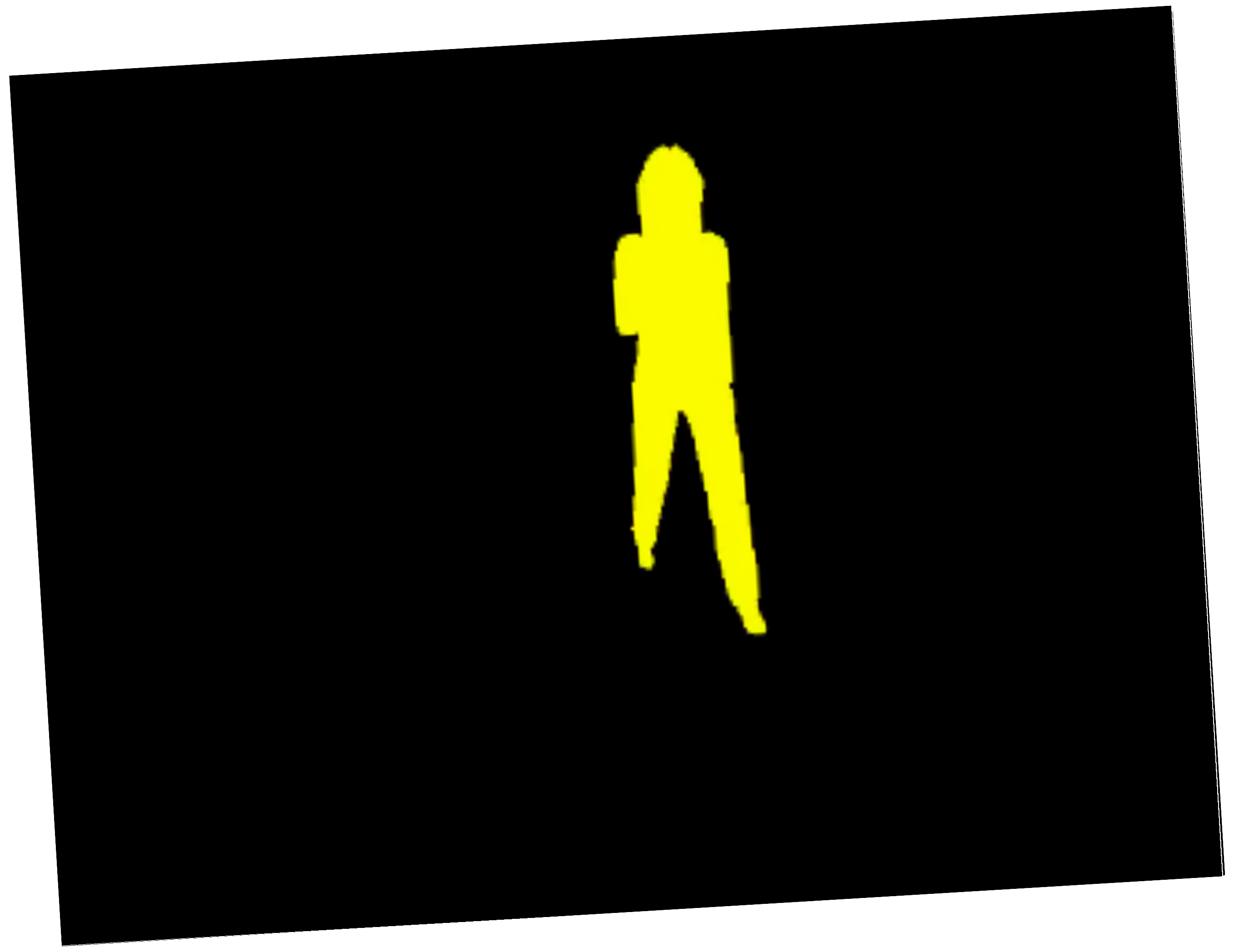
AI SWEDEN

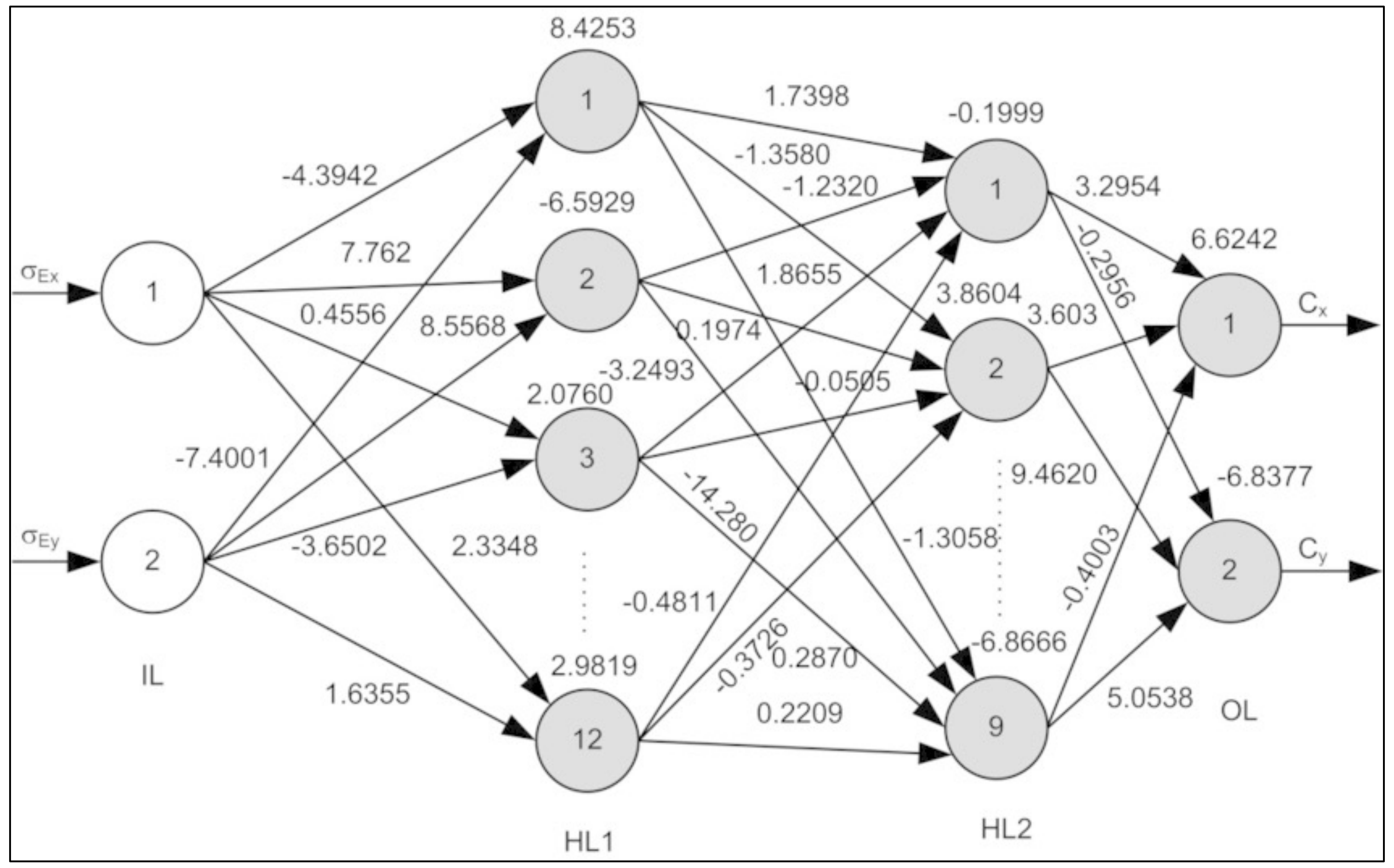
https://github.com/RI-SE/smirk/tree/main/pedestrian-generator

The SMIRK MVP

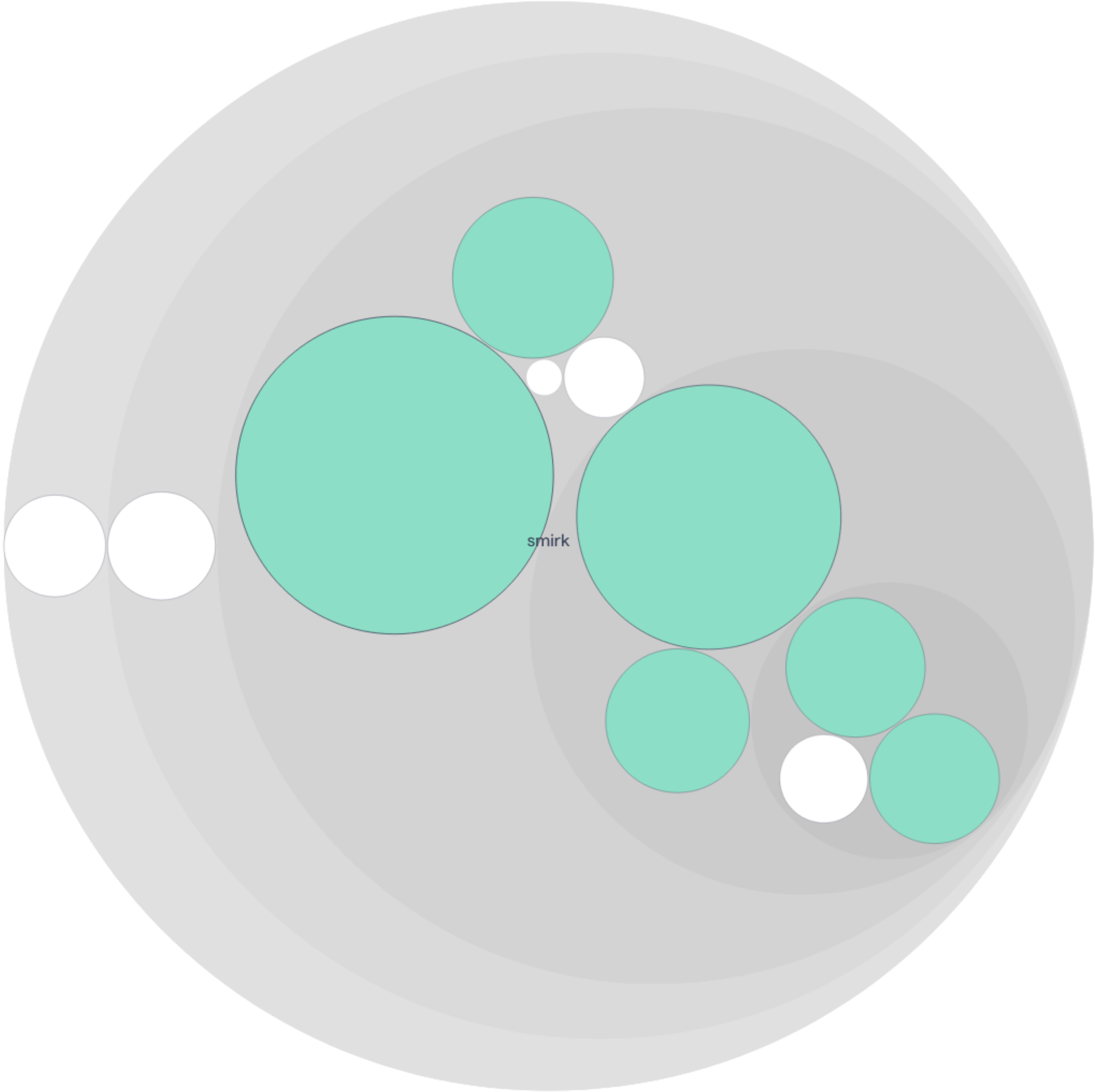








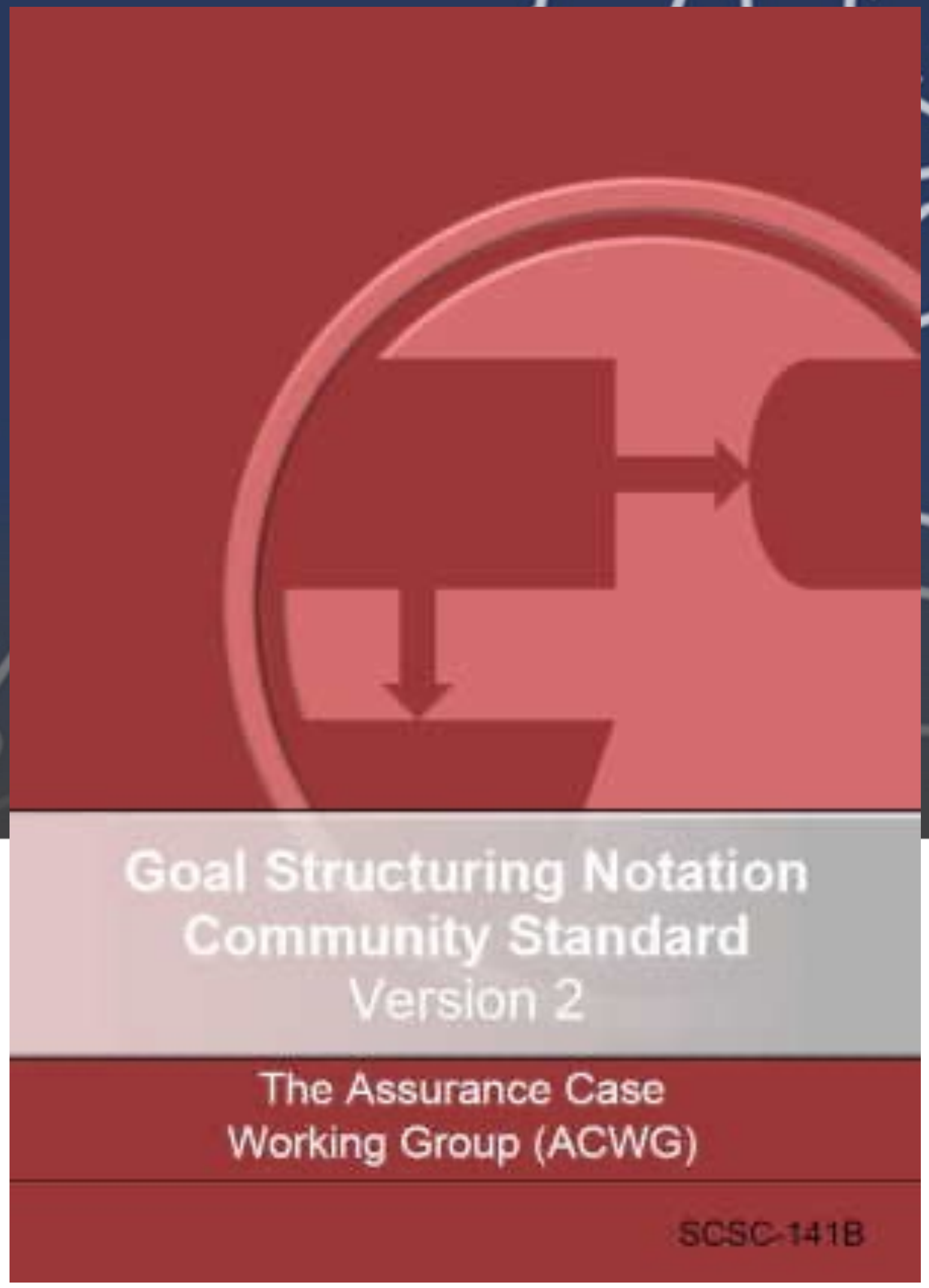
SMIRK CodeScene Analysis



11 files

Good code health

Safety Case Development Using AMLAS



ASSURING AUTONOMY
INTERNATIONAL PROGRAMME

Guidance on the Assurance of Machine Learning in Autonomous Systems (AMLAS)

Richard Hawkins, Colin Paterson, Chiara Picardi, Yan Jia, Radu Calinescu and Ibrahim Habli.

Assuring Autonomy International Programme (AAIP)
University of York

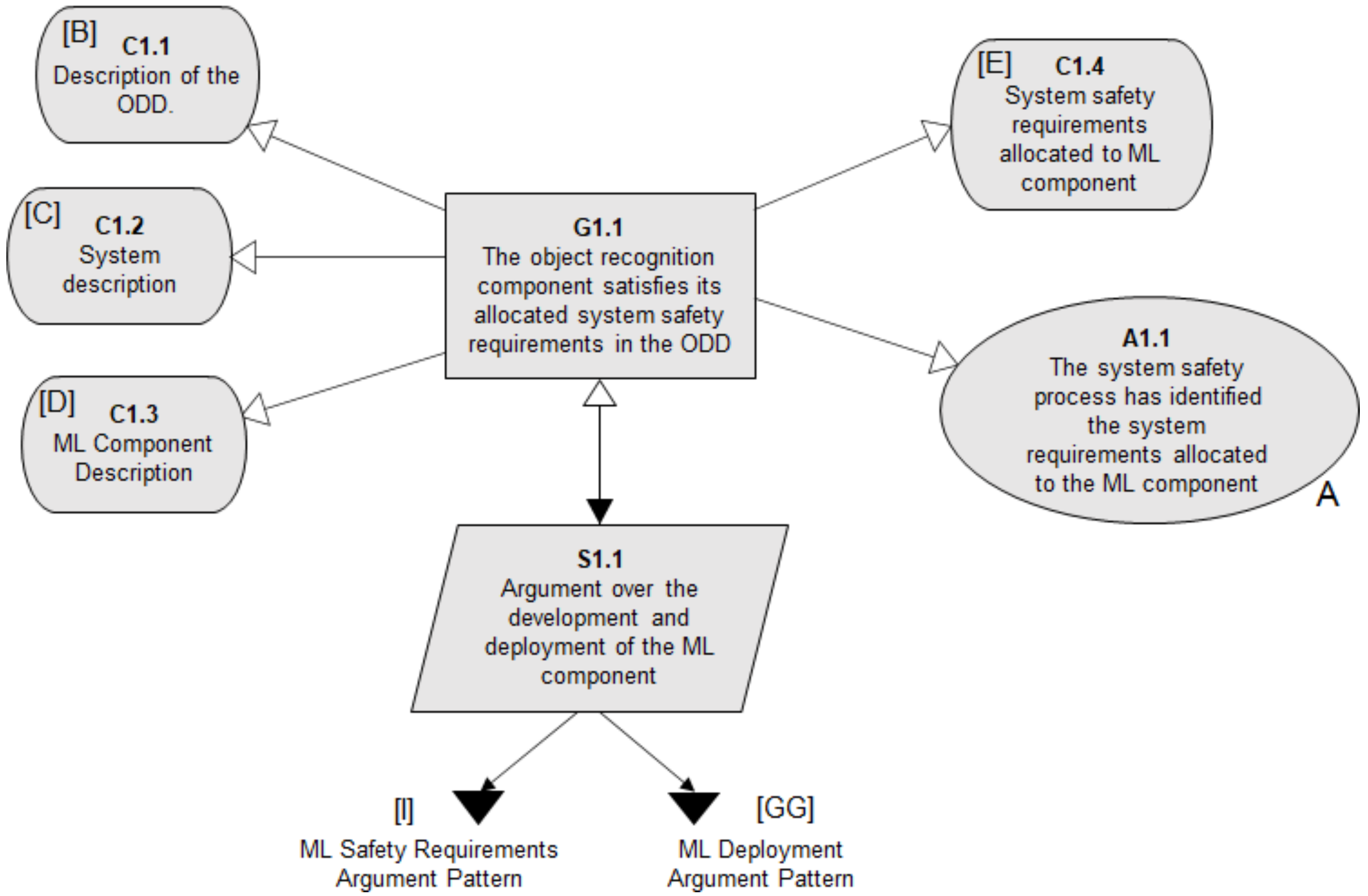
Version 1.1, March 2021

Contact : firstname.lastname@york.ac.uk.

Follow the AMLAS process

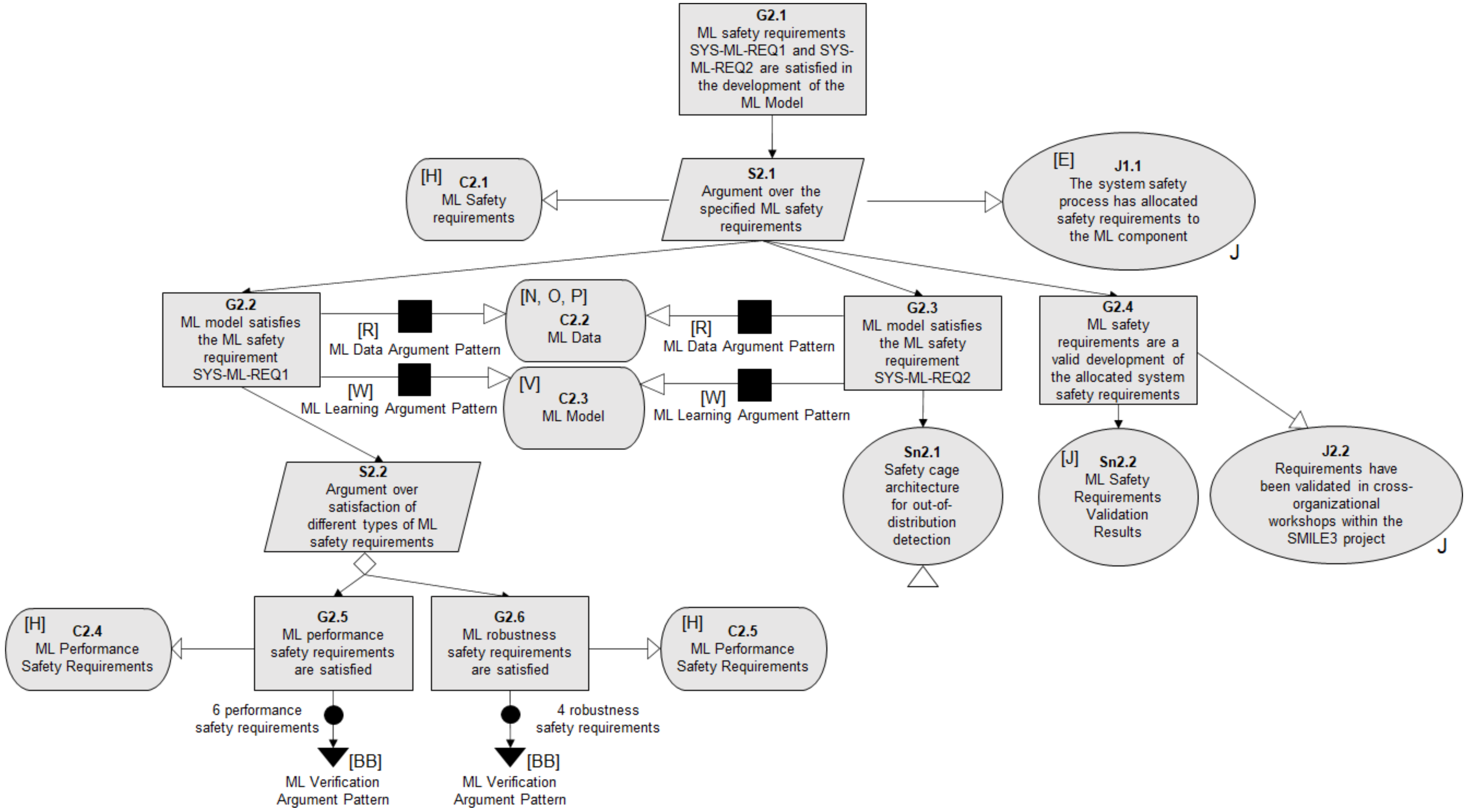
1. Safety Assurance Scoping

ID	Title	Input to	Output from	Where?	Status
[A]	System Safety Requirements	1, 6		SRS Sec 3.1	Done
[B]	Description of Operating Environment of System	1, 6		SRS Sec 4	Done
[C]	System Description	1, 6		SRS Sec 2	Done
[D]	ML Component Description	1		MLCS Sec 2	(J) Outlier detection missing
[E]	Safety Requirements Allocated to ML Component	2	1	SRS Sec 3.2	Done
[F]	ML Assurance Scoping Argument Pattern	1		SRS Sec 6	Done
[G]	ML Safety Assurance Scoping Argument		1	SRS Sec 7	Done
[H]	ML Safety Requirements	3, 4, 5	2	SRS Sec 3.3	Done
[I]	ML Safety Requirements Argument Pattern	2		SRS Sec 8	Done
[J]	ML Safety Requirements Validation Results		2	SRS Sec 9	Done
[K]	ML Safety Requirements Argument		2	SRS Sec 10	Done
[L]	Data Requirements		3	DMS Sec 2	Done
[M]	Data Requirements Justification Report		3	DMS Sec 3	Done
[N]	Development Data		3	TBD	(M) Hosting needed
[O]	Internal Test Data		3	TBD	(M) Hosting needed
[P]	Verification Data		3	TBD	(M) Hosting needed
[Q]	Data Generation Log		3	DMS Sec 4	Links to code needed
[R]	ML Data Argument Pattern	3		DMS Sec 5	Done
[S]	ML Data Validation Results		3	DMS Sec 6	(K) Validation scripts needed
[T]	ML Data Argument		3	DMS Sec 7	Done
[U]	Model Development Log		4	MLCS Sec 3	(K) Add links to code
[V]	ML Model	5, 6	4	TBD	(K) Need to upload model
[W]	ML Learning Argument Pattern	4		MLCS Sec 5	Done
[X]	Internal Test Results		4	Protocols	(K) Create test report
[Y]	ML Learning Argument		4	MLCS Sec 6	Done
[Z]	ML Verification Results		5	Protocols	(J) Measure slices
[AA]	Verification Log		5	STS Sec 3	(M) Need to describe metrics
[BB]	ML Verification Argument Pattern	5		STS Sec 5	Done
[CC]	ML Verification Argument		5	STS Sec 6	Done
[DD]	Erroneous Behaviour Log		6	DS Sec 4	(M) Need to report lessons
[EE]	Operational scenarios	6		STS Sec 4.1	Done
[FF]	Integration Testing Results		6	Protocols	(K?) Not started
[GG]	ML Deployment Argument Pattern	6		DS Sec 5	Done
[HH]	ML Deployment Argument		6	DS Sec 6	Done



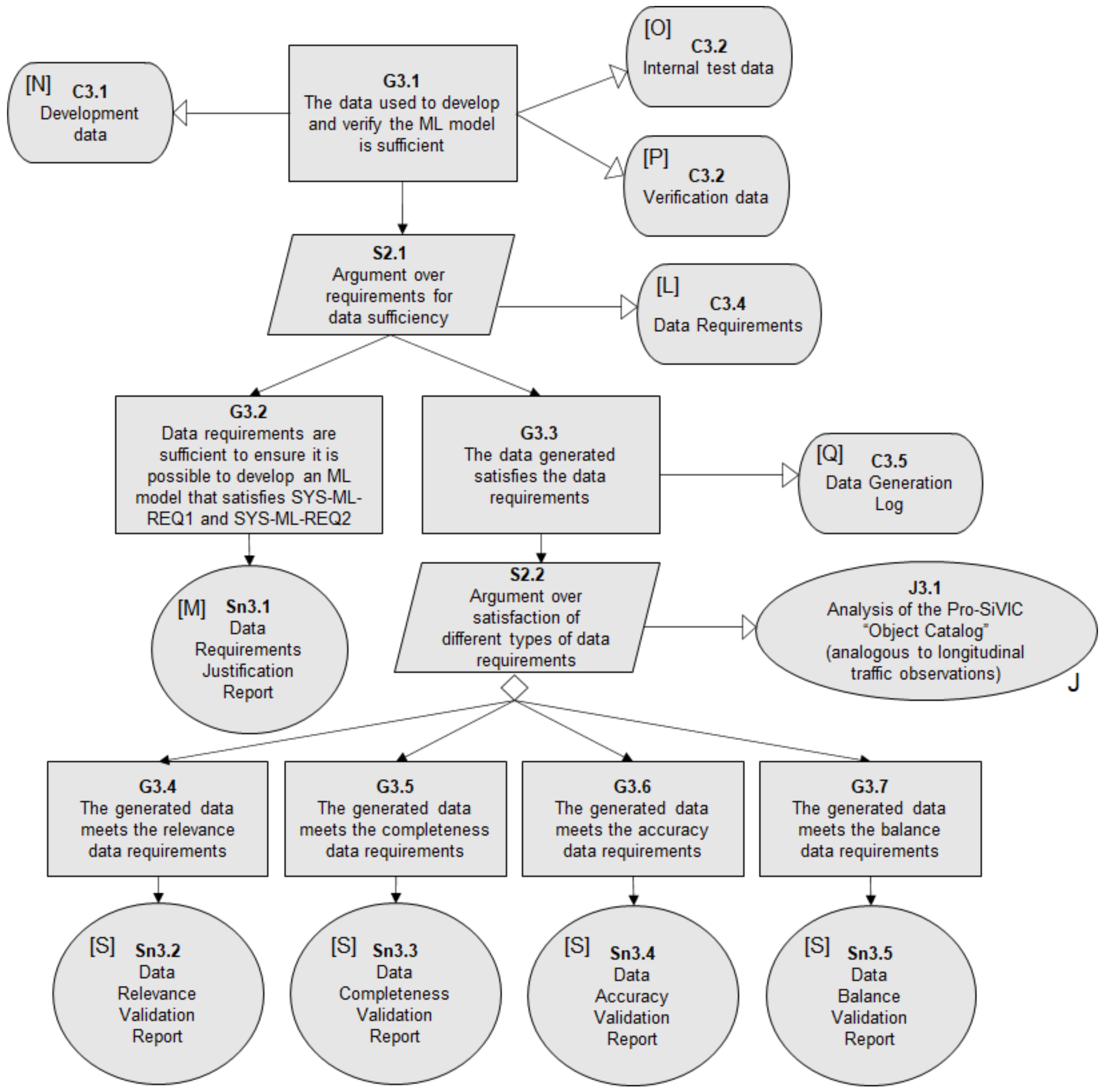
github.com/RI-SE/smirk

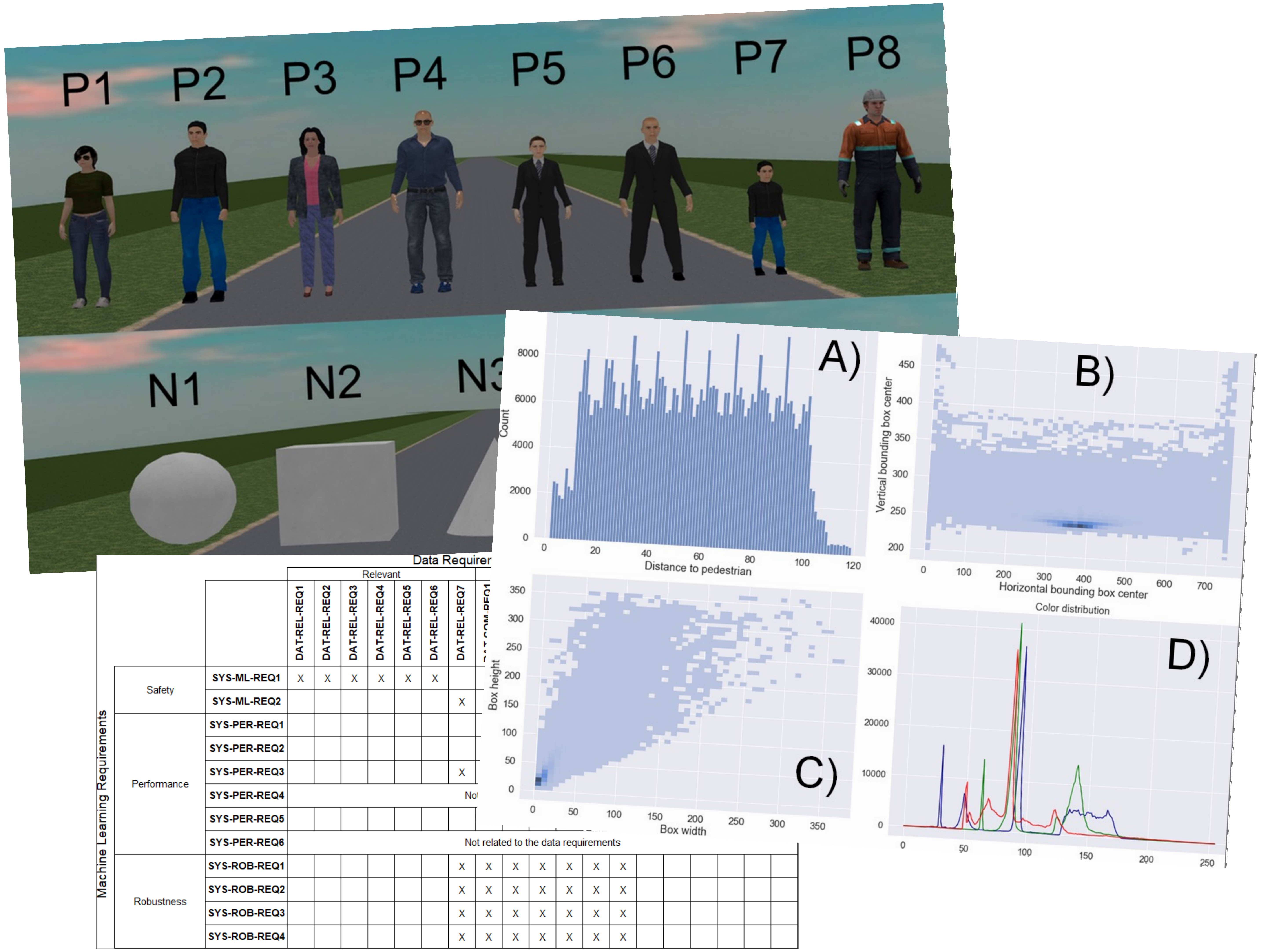
2. Requirements Assurance



Formal inspections

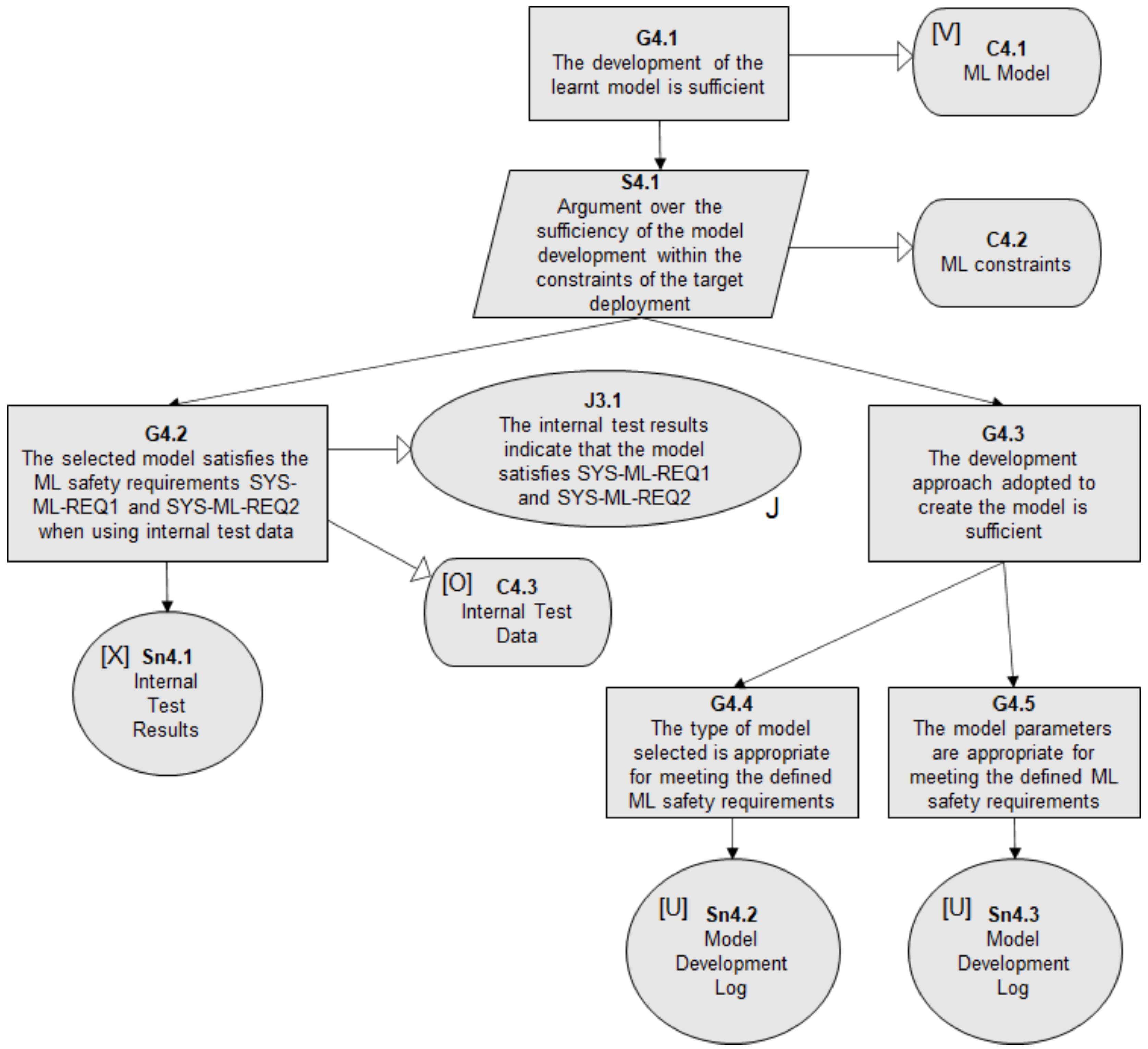
3. Data Management Assurance



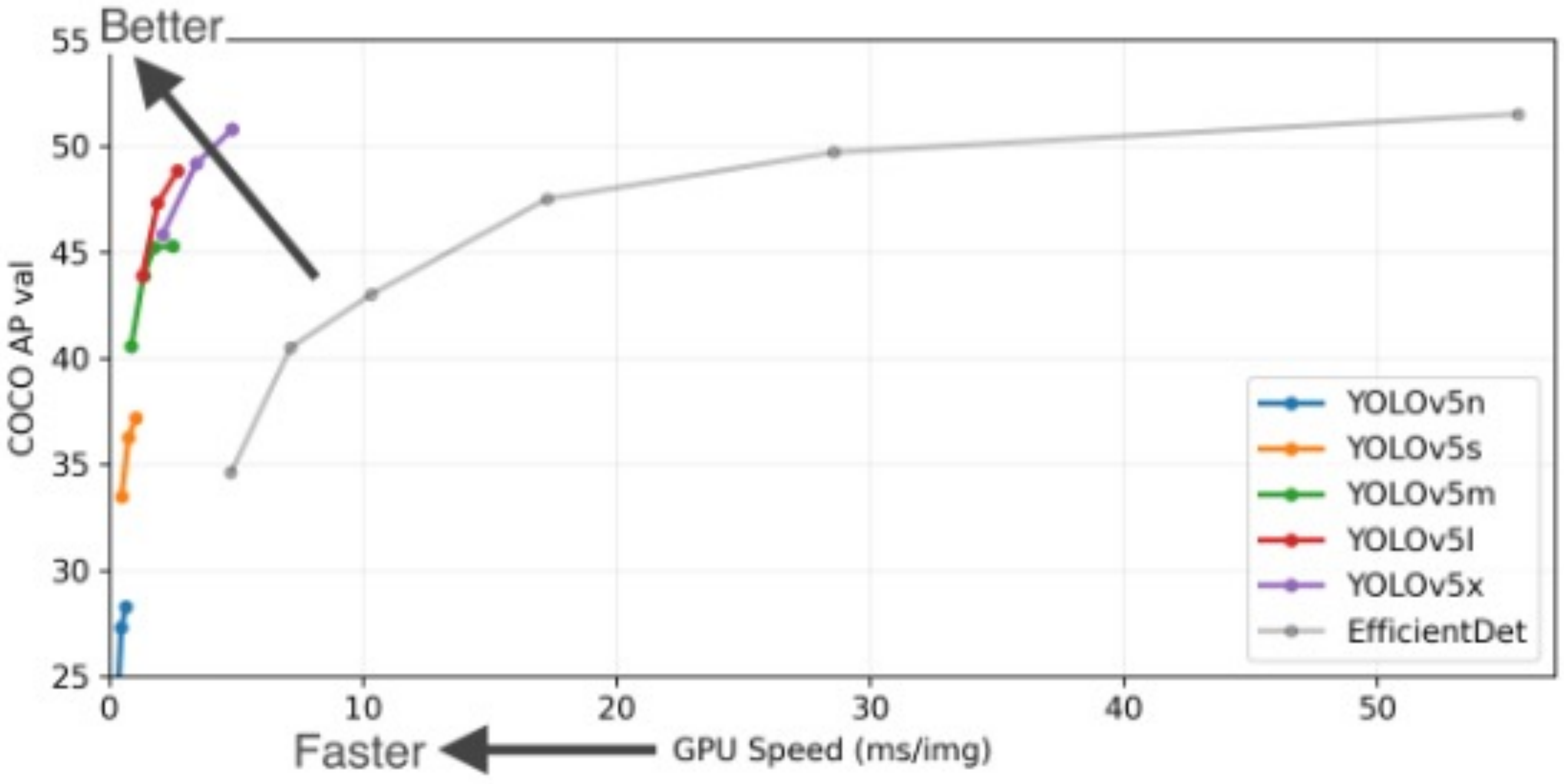


1) Relevance
2) Completeness
3) Accuracy
4) Balance

4. Model Learning Assurance

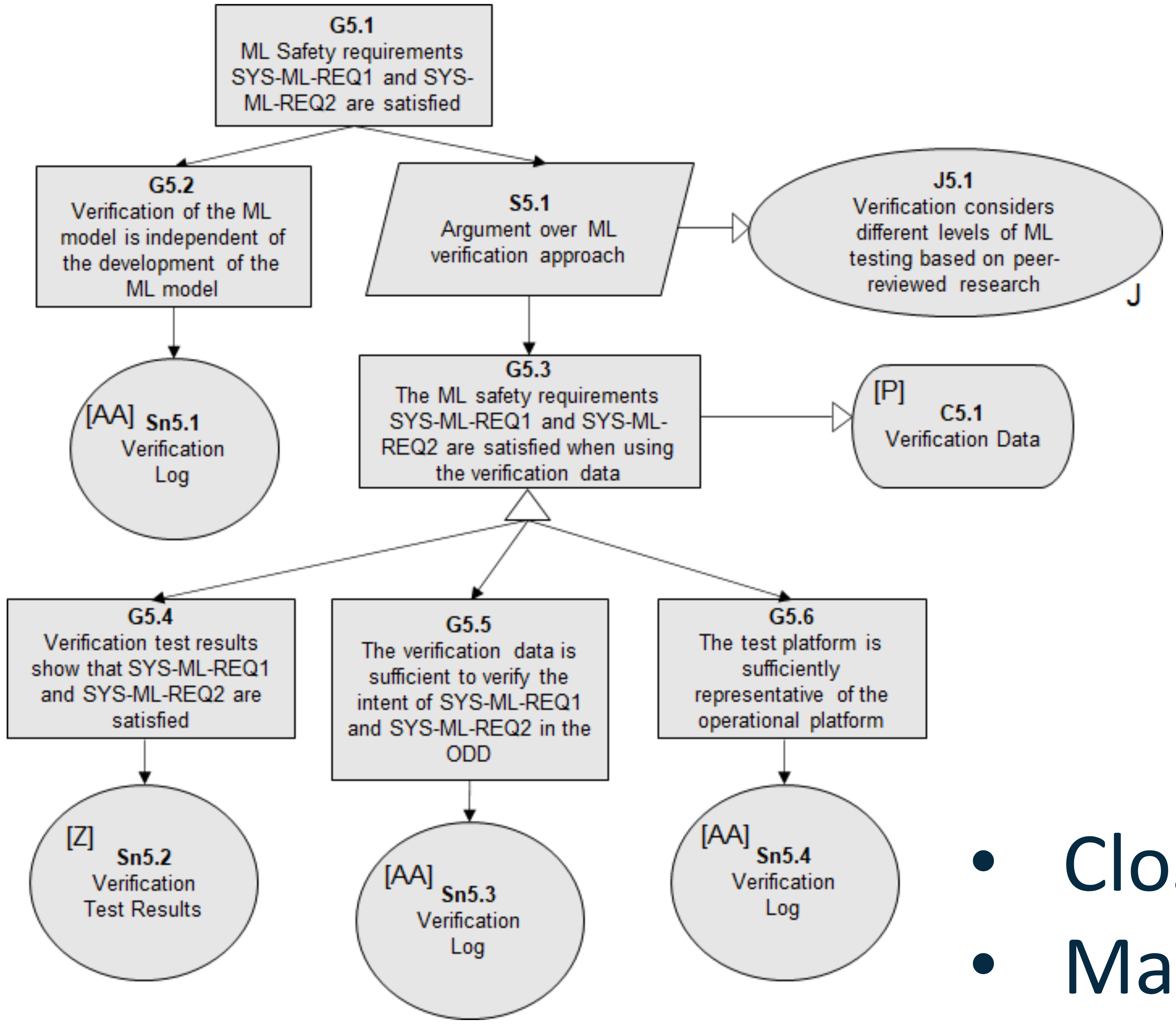


State-of-the-art architectures



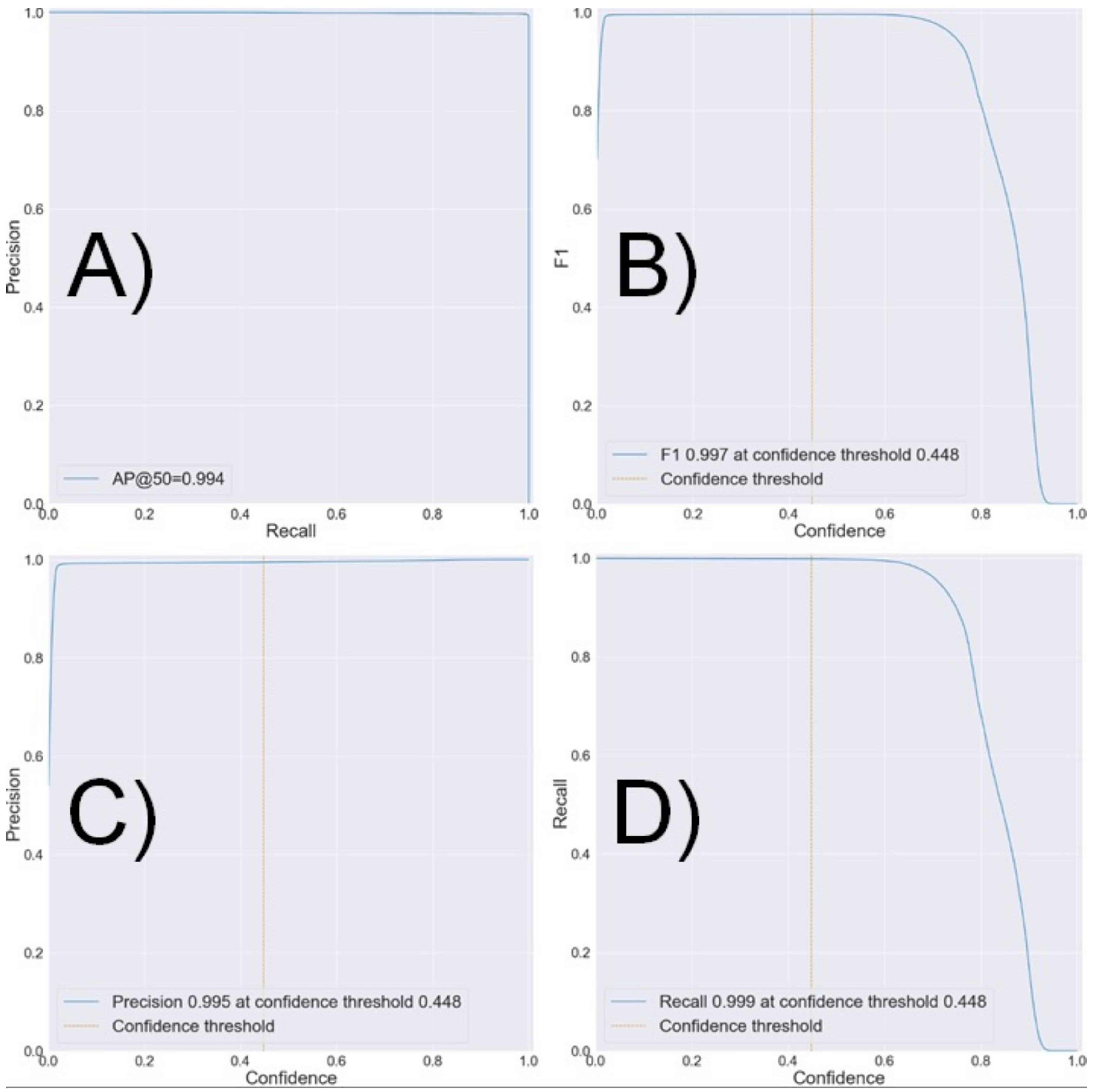
Tradeoffs

5. Model Verification Assurance

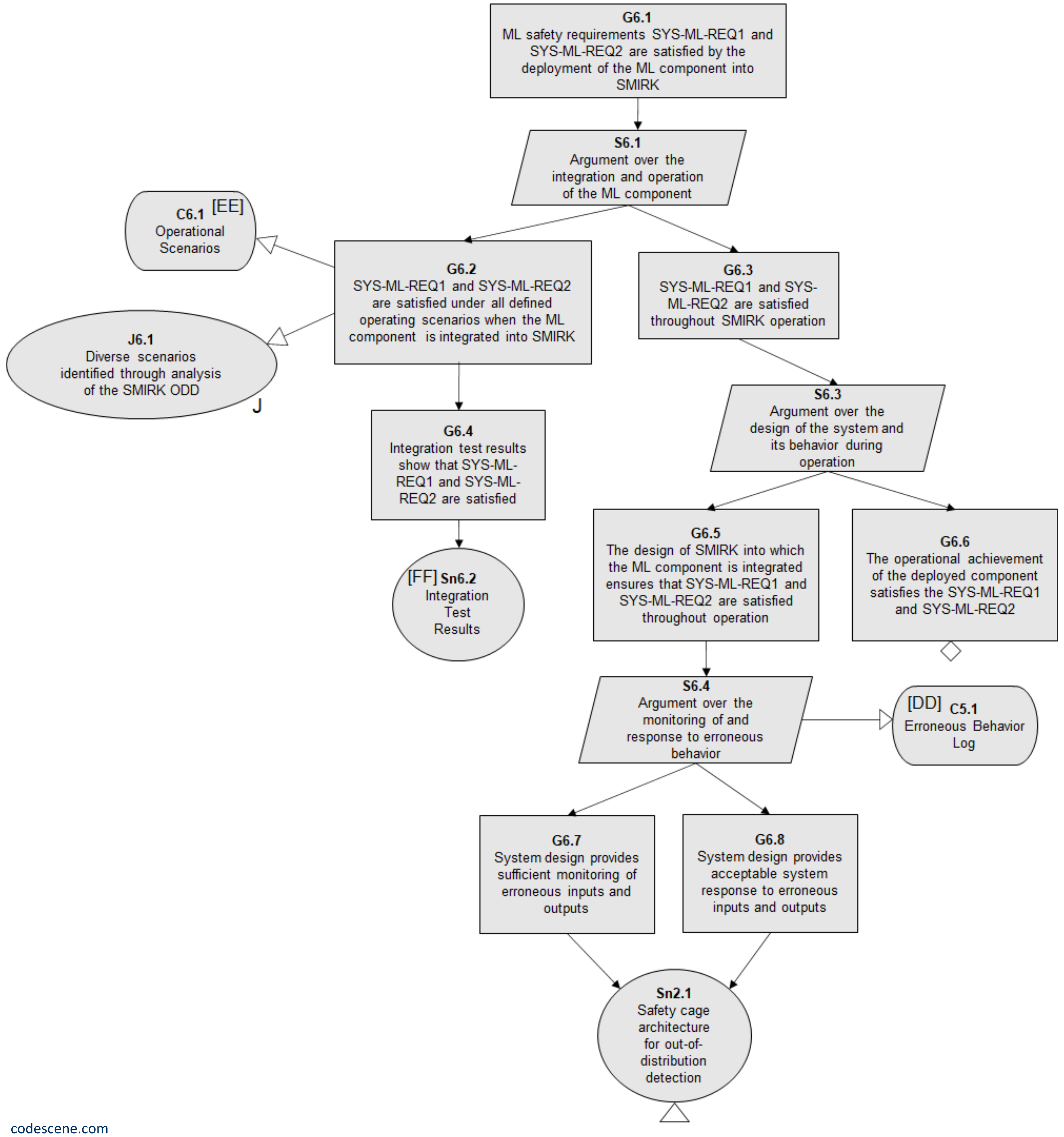


Analysis of subsets

- Close/Far away
- Male/Female/Children
- Standing/Walking/Running
- …

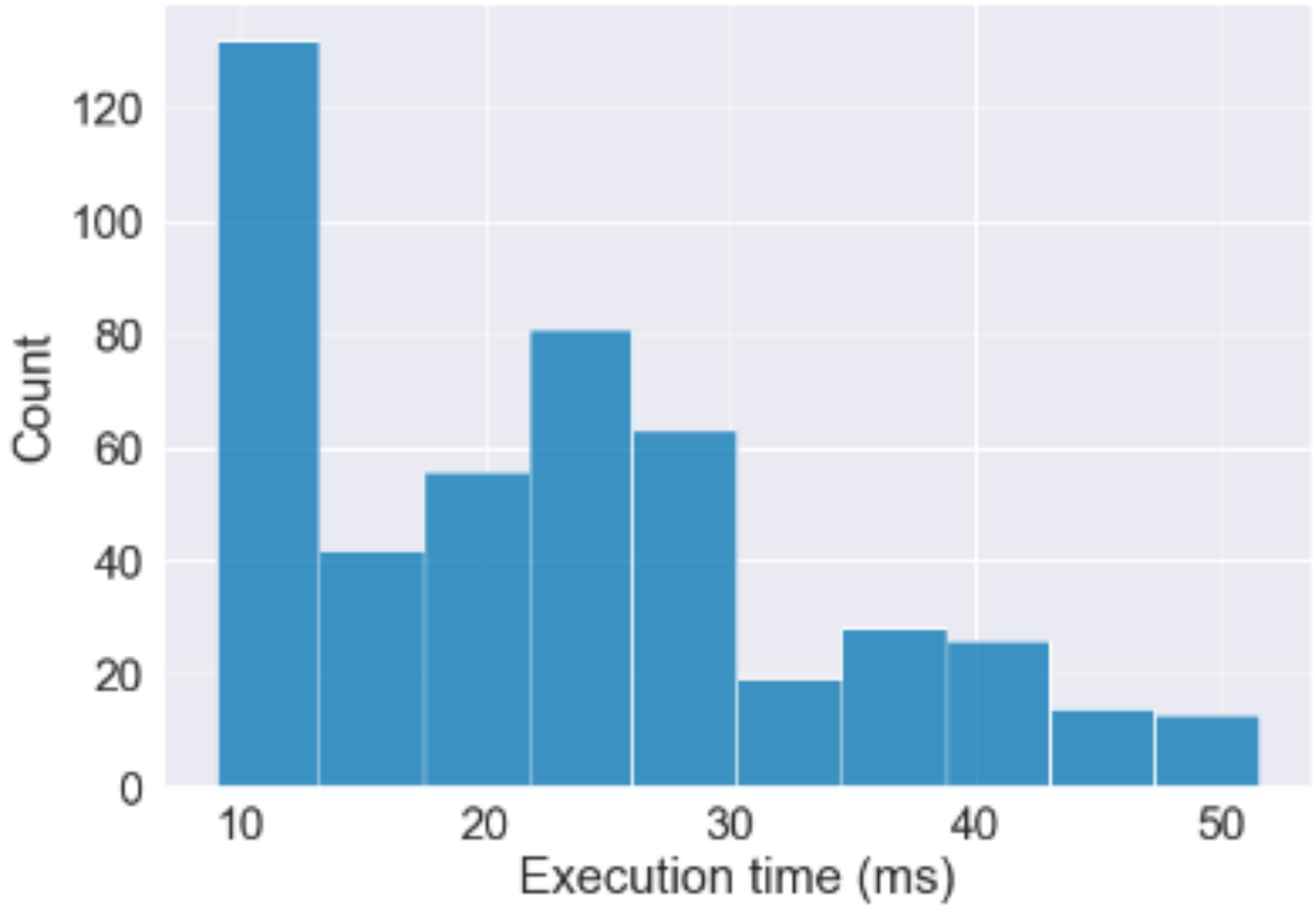


6. Model Deployment Assurance



Integration testing

- Equivalence partioning
- Pairwise testing
- Random testing



Demo

Without braking

With SMIRK

Lessons Learned and Wrap-up

Lessons Learned

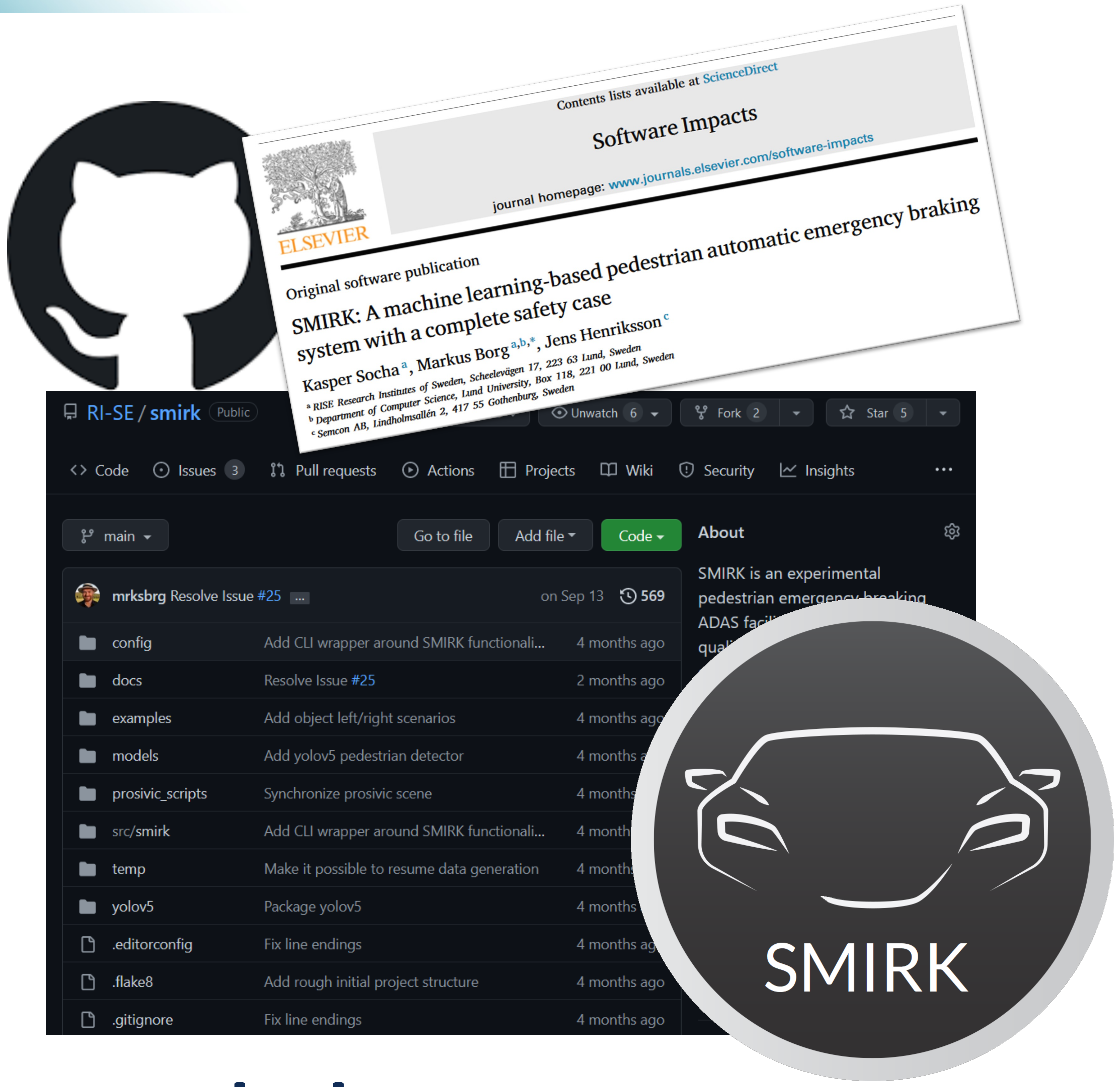
SOTIF and AMLAS compatible

Simulated data threatens validity

of negative samples

Evaluation of object detection models is hard

Open ML safety case



Open ML-based demonstrator

Open ML safety case

Requirements engineering for data

Technical debt in automotive software

Questions?

markus.borg@codescene.com

Open ML-based demonstrator

References

- Code: https://github.com/RI-SE/smirk/
- Data: https://www.ai.se/en/data-factory/datasets/data-factory-datasets/smirk-dataset
- Demonstrator: Socha, Borg, and Henriksson, SMIRK: A Machine Learning-Based Pedestrian Automatic Emergency Braking System with a Complete Safety Case, Software Impacts, Volume 13, 2022.
- Safety Case: Borg, Henriksson, Socha, Lennartsson, Lönegren Sonnsjö, Bui, Tomaszewski, Sathyamoorthy, Brink, and Helali, Ergo, Ergo, SMIRK is Safe: A Safety Case for a Machine Learning Component in a Pedestrian Automatic Emergency Brake System, https://arxiv.org/abs/2204.07874

- Ashmore, Calinescu, and Paterson, Assuring the Machine Learning Lifecycle: Desiderata, Methods, and Challenges, *ACM Computing Surveys*, 54(5), 2021.
- Ben Abdessalem, Nejati, Briand, and Stifter, Testing Advanced Driver Assistance Systems Using Multi-objective Search and Neural Networks, In *Proc. of the 31st Int'l. Conf. on Automated Software Engineering*, 2016.
- Thorn, Kimmel, Chaka *et al.*, A Framework for Automated Driving System Testable Cases and Scenarios, National Highway Traffic Safety Administration US Department of Transportation, 2018.
- Hawkins, Paterson, Picardi *et al.*, Guidance on the Assurance of Machine Learning in Autonomous Systems (AMLAS), v1.1, Assuring Autonomy Int'l. Programme, University of York, 2021.