

Challenges in Testing and Validating Collaborative Robotics Technologies

Michael Wagner
Co-Founder and CEO
Edge Case Research

Edge Case Research



Formed in 2013 by Carnegie Mellon researchers to **make complex software more robust**

Team of over ten people with deep experience in dependability, robotics, and testing

Clients across markets including consumer electronics, automotive, defense, robotics, mining, industrial, power, finance, etc.

Collaborative Robotics Today

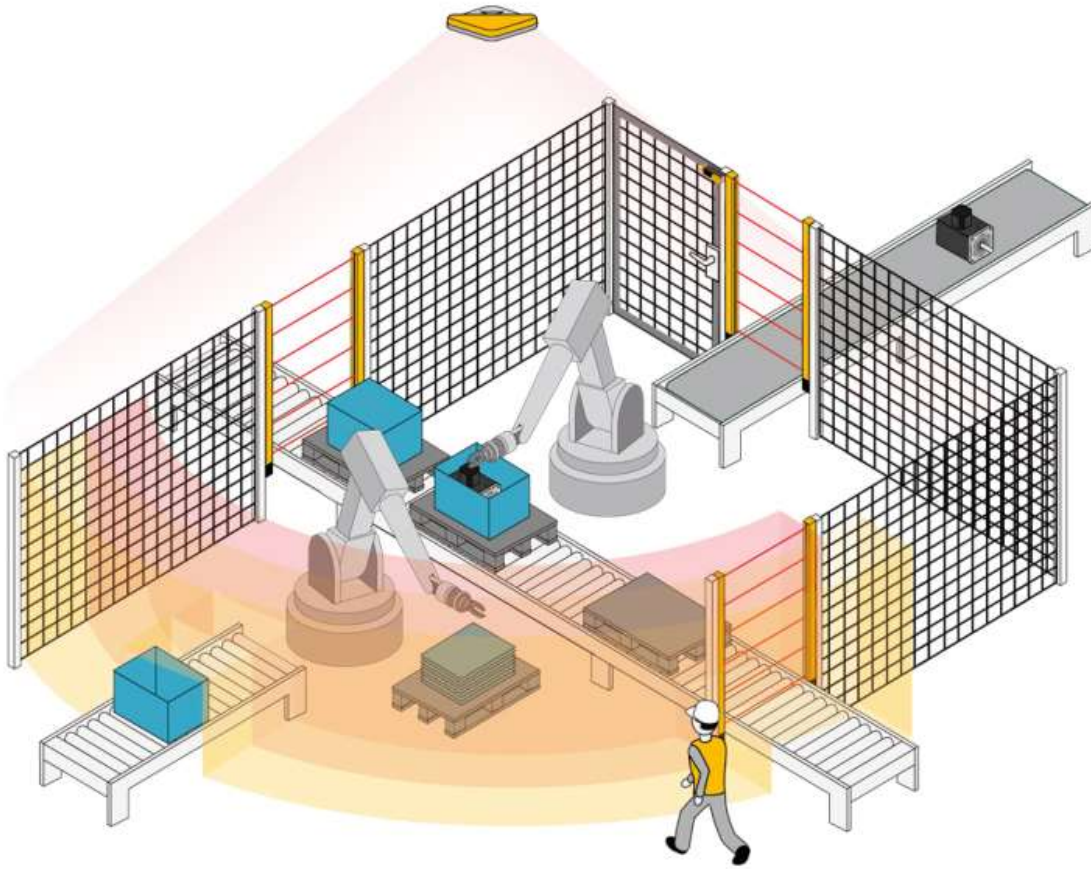


Image: Pilz Automation Safety



Image: Rethink Robotics



Image: Robotiq

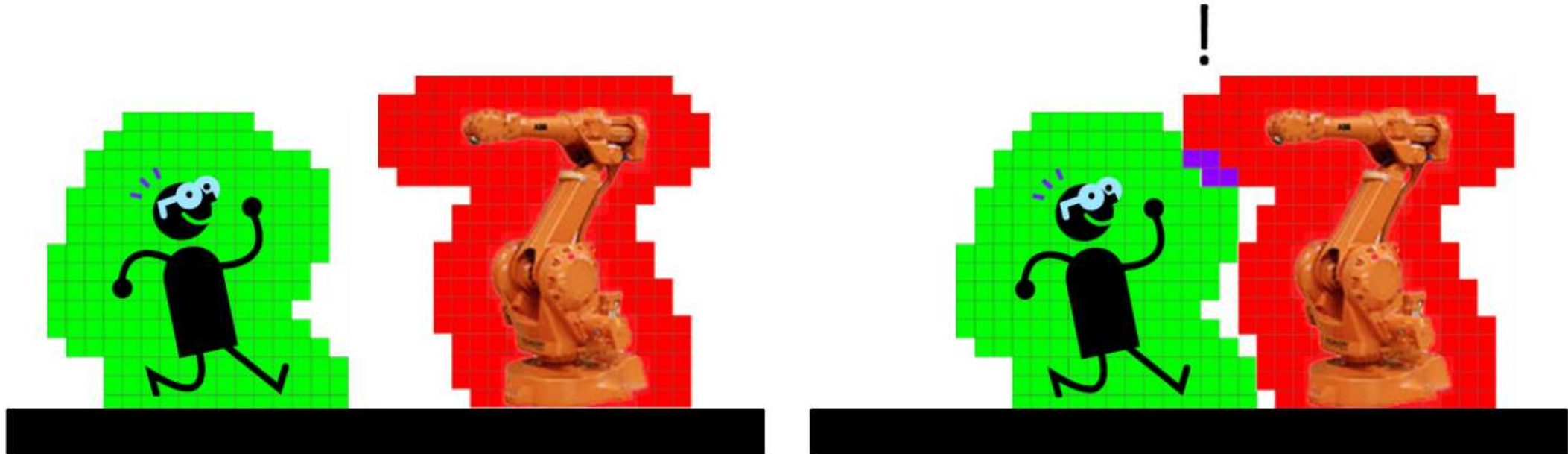
Collaborative Robotics of the Future



Image: Peter Yang

For some tasks, maximizing productivity will demand more sophisticated interaction than what is achievable with discretized safety zones.

A General Formulation of the Problem



Safety (green) and danger (red) zones. As long as these zones are disjoint (left) then safety is maintained and robots may operate normally. Once they intersect (right) safety may be compromised and robots must achieve a safe state. (Anderson-Sprecher, 2011)

Example: The Hybrid Safety System



Carnegie Mellon developed the HSS for the Office of Naval Research's Automated Weapons Assembly program for semi-automated work cell for shipboard munitions assembly.

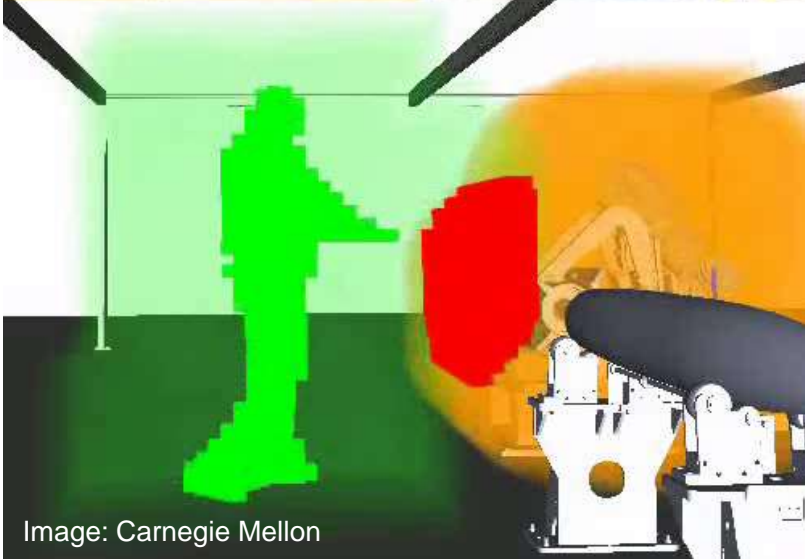


Image: Carnegie Mellon

Autonomy Meets Safety

“Validation: The process of determining that the requirements are the correct requirements and that they are complete.”

Software Considerations in Airborne Systems
and Equipment Certification
(RTCA DO-178C)

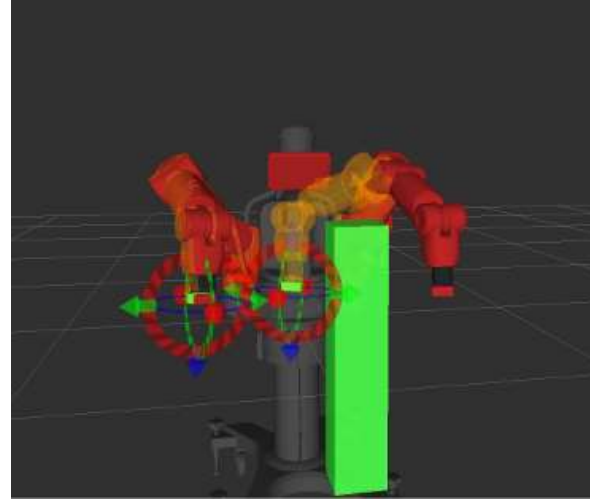
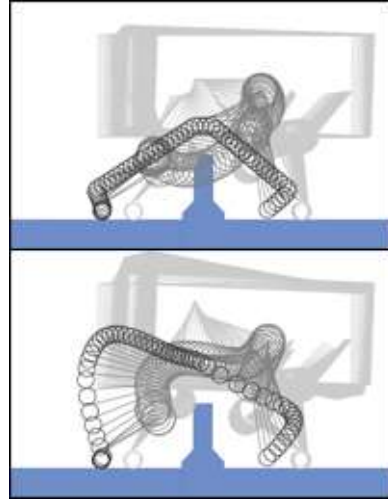
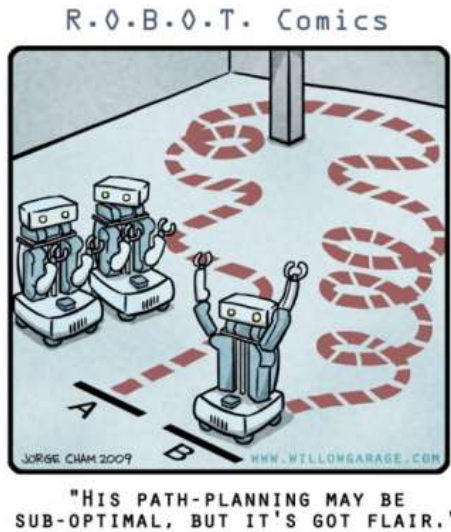
Autonomy Meets Safety

“Uncertainty arises because of both laziness and ignorance. **It is inescapable in complex, nondeterministic, or partially observable environments.**”

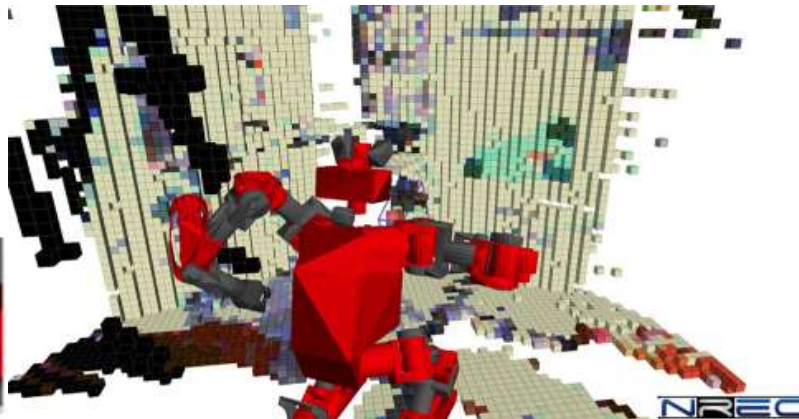
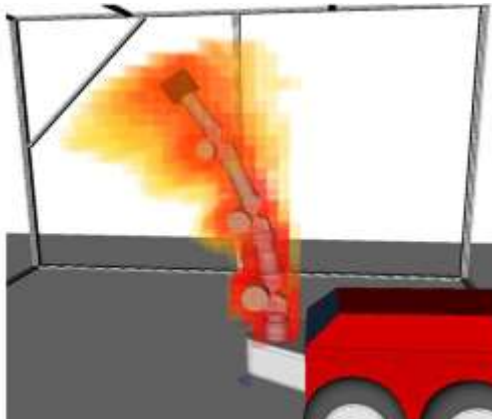


Stuart Russell and Peter Norvig

Collaborative Robotics Technologies



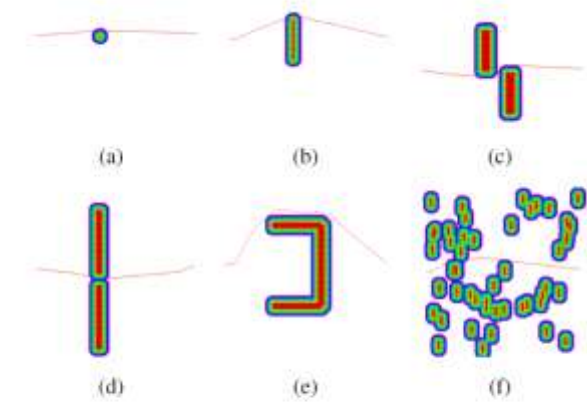
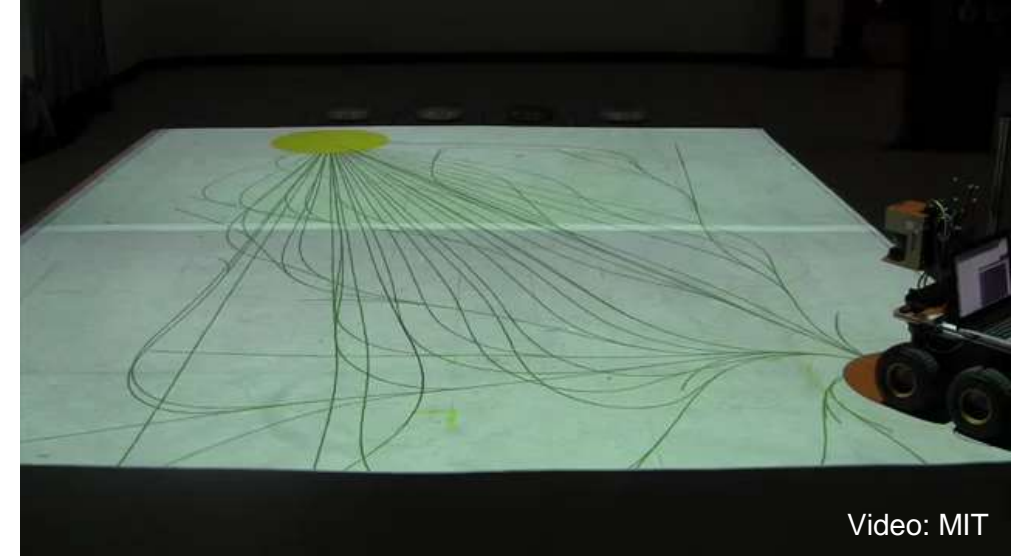
High-Dimensional
Motion Planning



Perception
and Modeling

Motion Planning

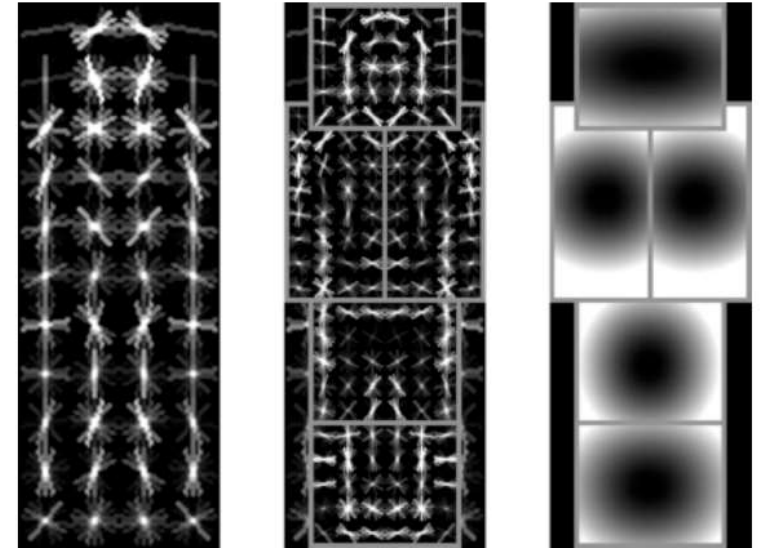
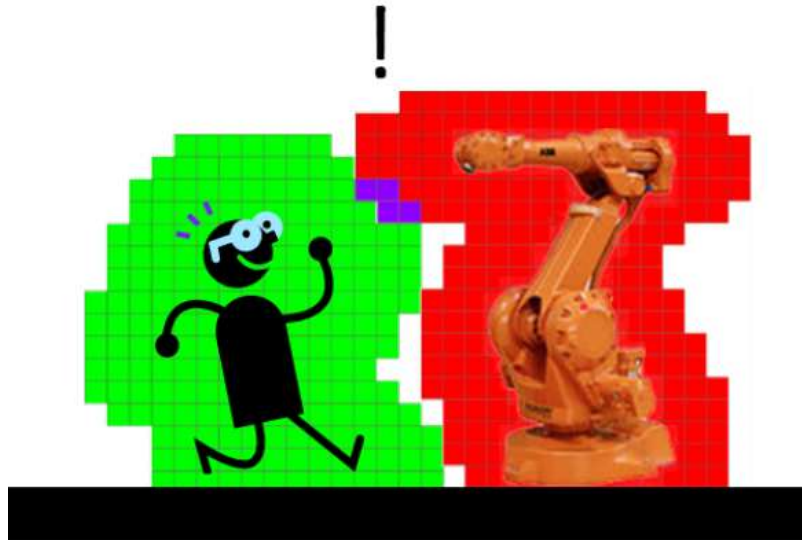
- Highest performance approaches involve Monte Carlo sampling
- Implications for testing and validation:
 - If you can carefully control random number generator, maybe you can reproduce behavior in unit test
 - Test reproducibility may be impossible at the system level
 - Validation demands a large, statistically significant number of tests across many different scenarios
 - *Algorithm failures may not indicate test failures*



(Choudhury et al., 2015)

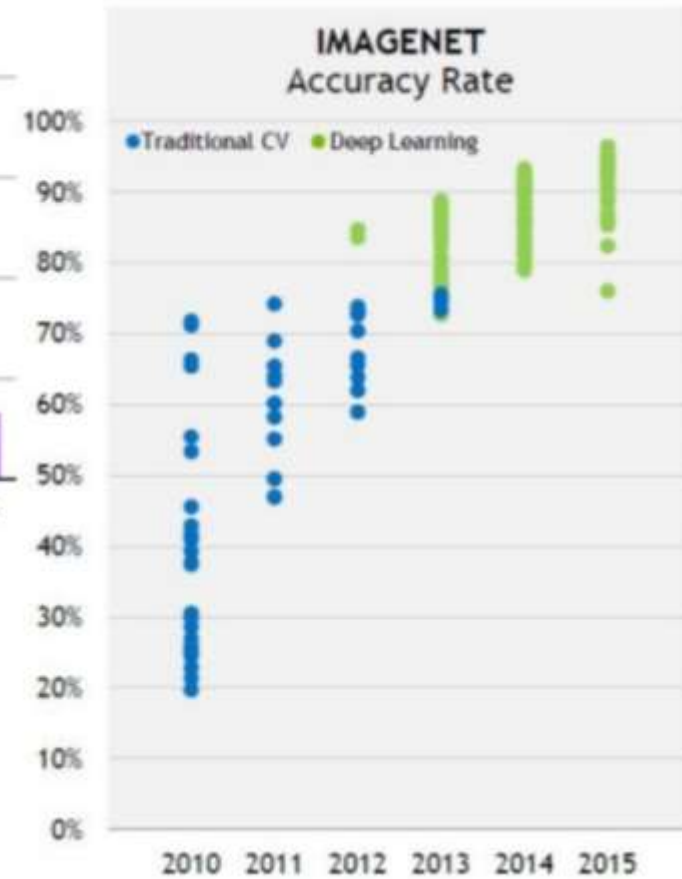
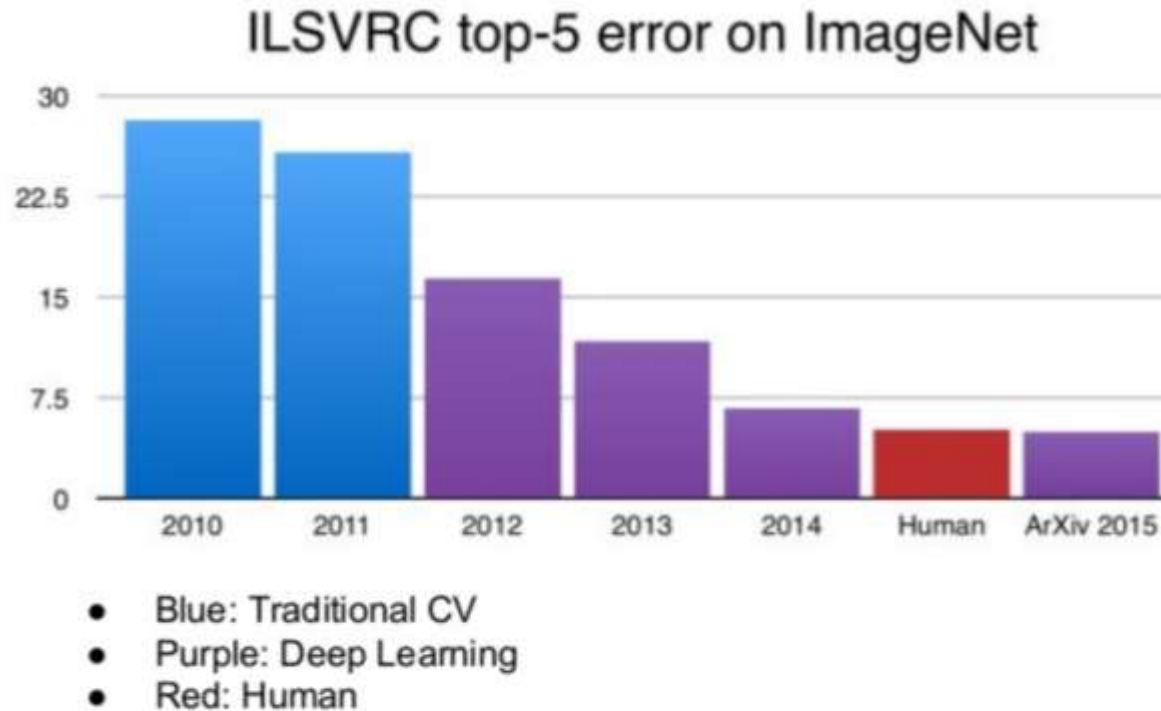
Perception

- Ultimately we want the safety system to sense the location, speed, and posture of all persons in the work cell
- 3D perception is an extremely complex problem!



(Felzenszwalb et al., 2010)

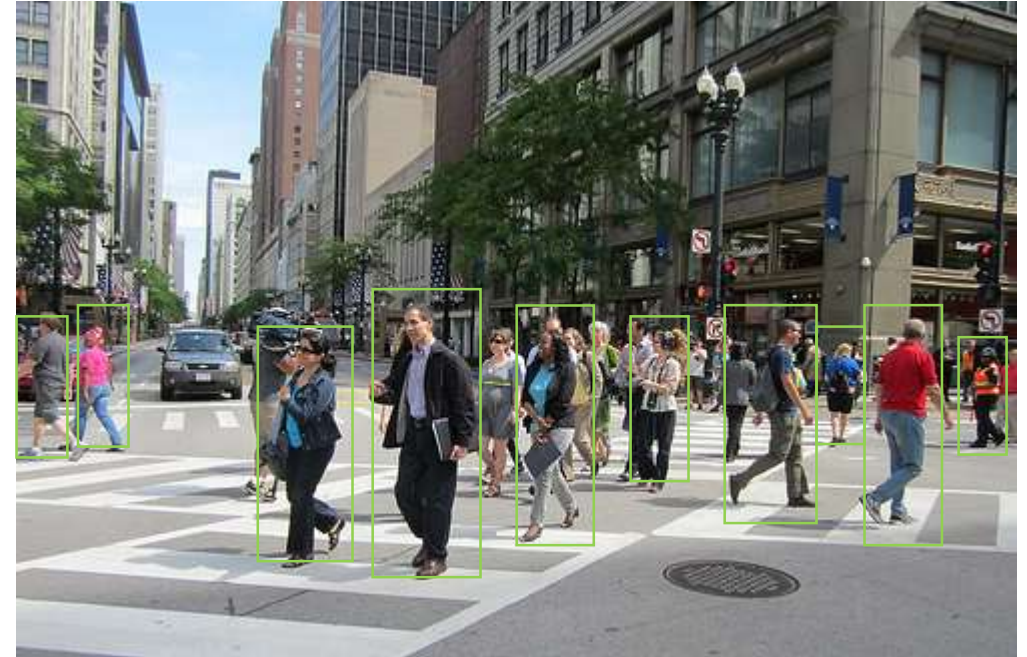
Perception is Deep Learning



(Sapunov, 2016)

Challenge: Legibility of Deep Learning

- **Can humans understand how deep learning works?**
- Deep learning “learns” features and rules from training data
- Commonly the weighting is inscrutable, or at least not intuitive
 - There is an unknown (significant?) chance results are brittle
 - E.g., accidental correlations in training data, sensitivity to noise



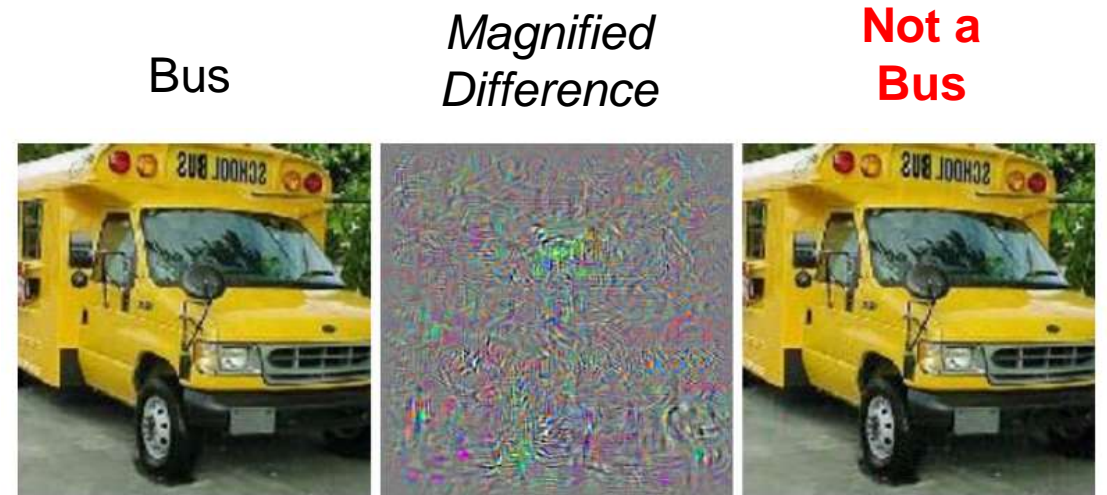
Challenge: Brittleness



Car

**Not a
Car**

*Magnified
Difference*



Bus

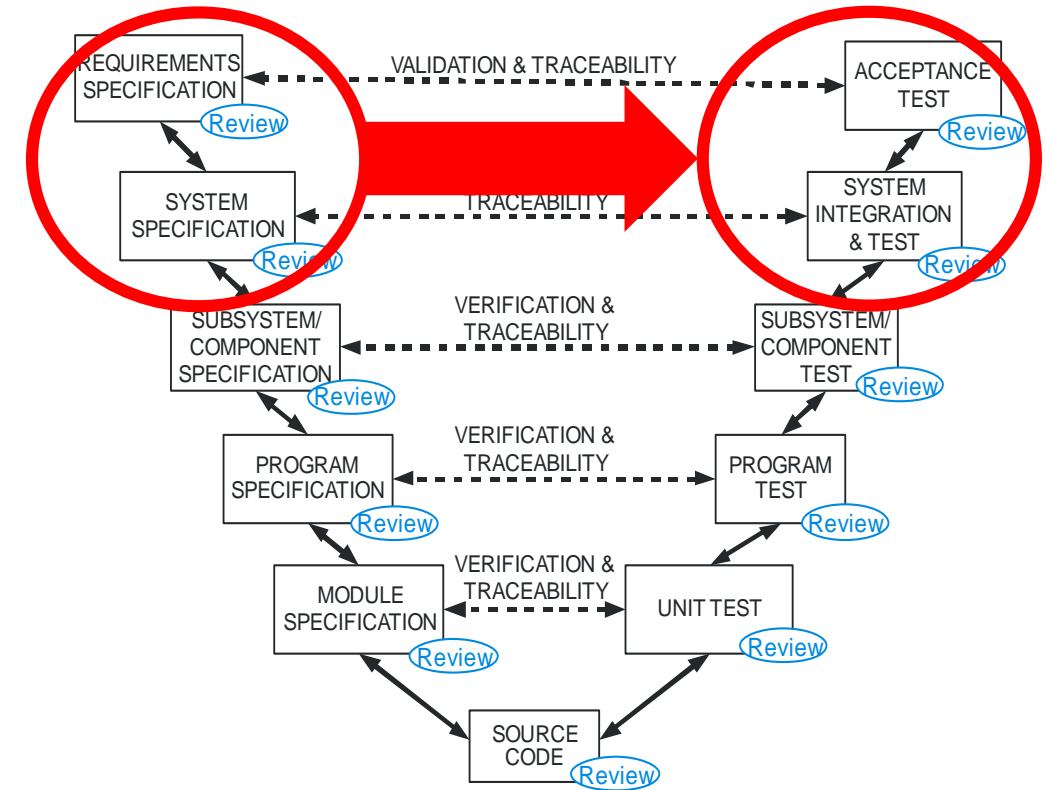
*Magnified
Difference*

**Not a
Bus**

(Szegedy et al., 2013)

Challenge: Lack of Requirements

- The “V process” model prescribed by software safety standards such as IEC-61508 traces requirements to V&V
- But where are the requirements in a machine learning based system?
- The machine-learning software is just a framework...**the training data form *de facto* requirements**



(Koopman and Wagner, 2016)

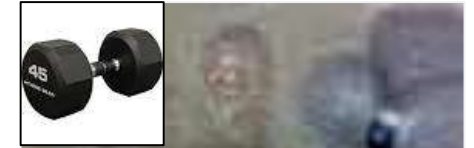
Challenge: Lack of Requirements

- How do you know the training data are “complete”?
- Incorrect training data are safety-critical artifacts and must be handled rigorously
- What if a moderately rare case isn't in the training set? It might not behave as you expect.
- People's perception of “almost the same” stimulus does not predict a deep learner's responses!



“Baseball”

(Nguyen et al., 2015)

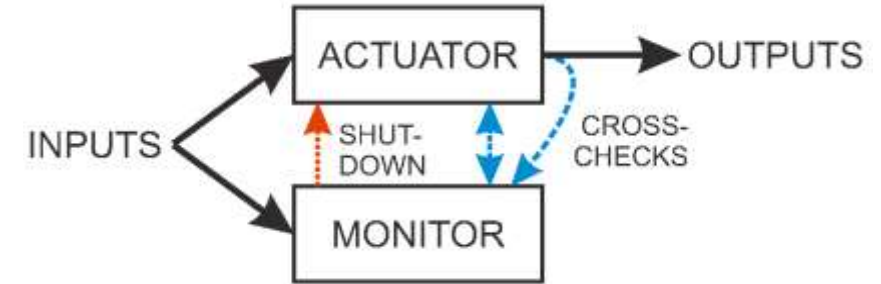


“Dumbbell”

(Mordvintsev et al., 2015)

Monitor / Actuator Architecture

- All safety requirements are allocated to Monitor
 - Monitor performs safety shutdown if unsafe outputs/state detected
 - Monitor is non-ML software that enforces a safety “envelope”
- Actuator is the perception or planning software
 - *Usually* works
 - But, might sometimes be unsafe
 - Failures are availability problems, not safety problems
- In practice, we’ve had success with this approach
 - E.g., over-speed shutdown on APD
 - **Important point: need to be clever in defining what “safe” means to create monitors**



APD is the first unmanned vehicle to use the Safety Monitor.
(Unclassified: Distribution A. Approved for Public Release. TACOM
Case # 19281 Date: 20 OCT 2009)

Why Do We Test?

- Traditional testing confirms proper functionality
- Machine learning uses inductive learning, which is vulnerable to “black swan” failures
- So given what we don’t know, we also need testing to try to ***falsify correctness hypotheses***

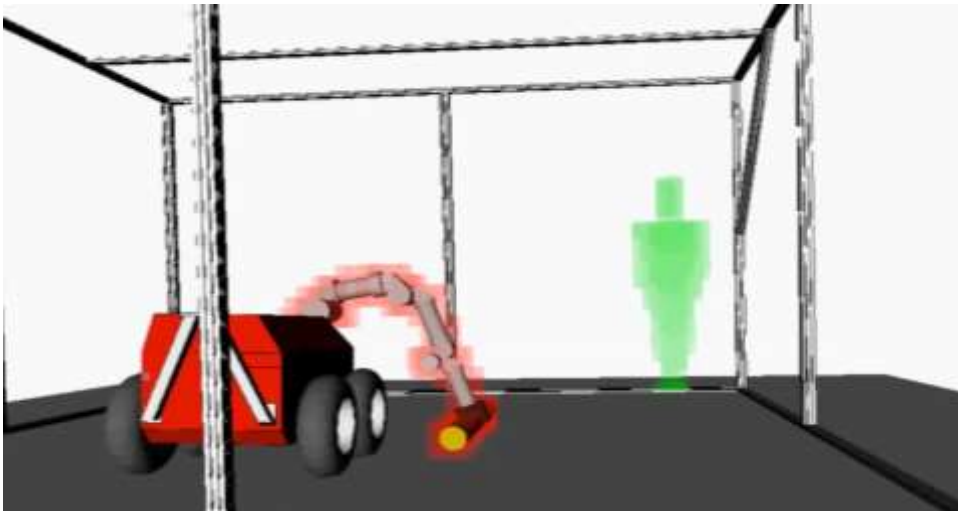
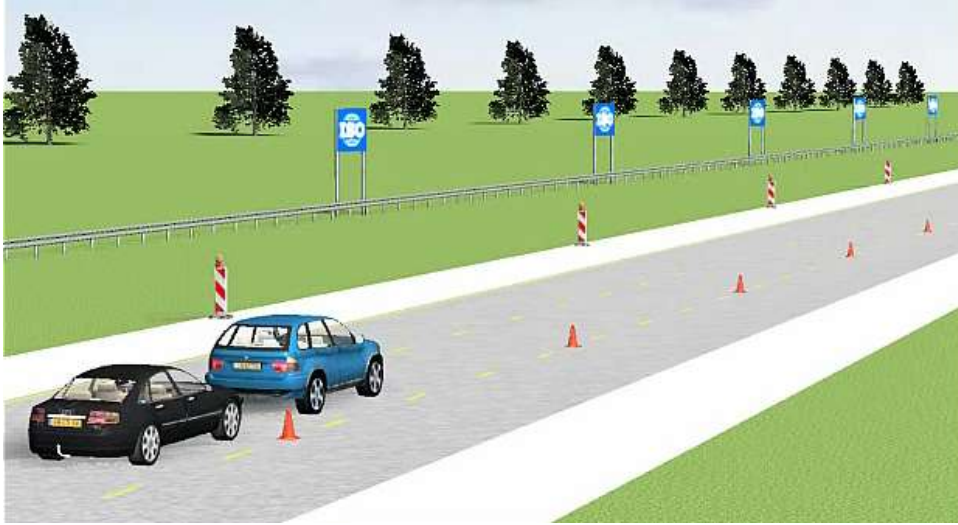


Thousands of miles of “white swans”...



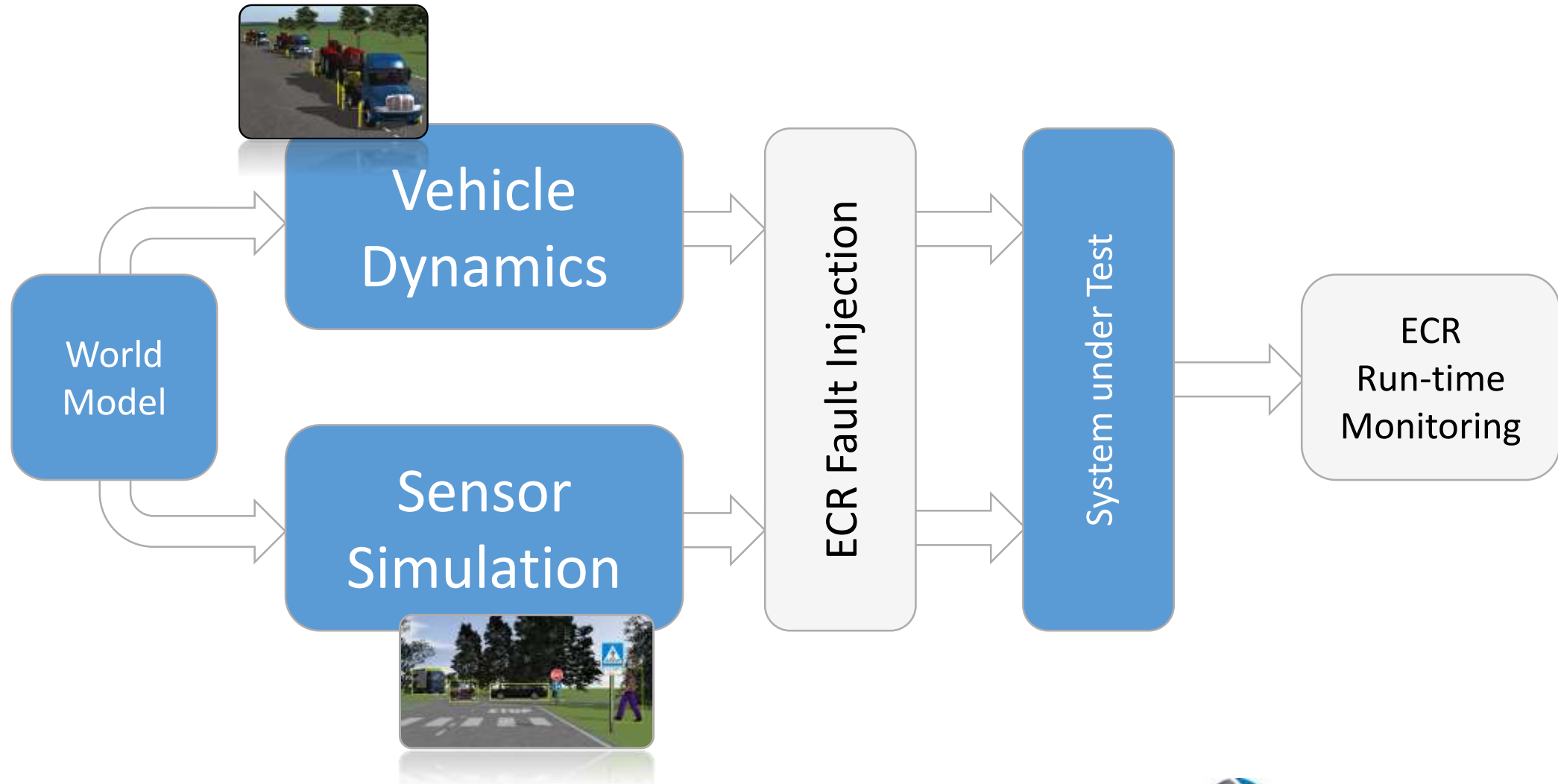
Make sure to fault inject
some “black swans”

Robustness Testing



Switchboard robustness testing cuts through the endless number of possible tests and finds problems you don't expect.

Robustness Testing



Conclusions

- **We use non-determinism, deep learning, etc. when we don't know how something works**

Conclusions

- We use non-determinism, deep learning, etc. when we don't know how something works...**and the nominal performance of these techniques is excellent!**

Conclusions

- We use non-determinism, deep learning, etc. when we don't know how something works...and the nominal performance of these techniques is excellent!
- **But they confound traditional safety practices**

Conclusions

- We use non-determinism, deep learning, etc. when we don't know how something works...and the nominal performance of these techniques is excellent!
- But they confound traditional safety practices
- **Smart choices about system architecture can help**

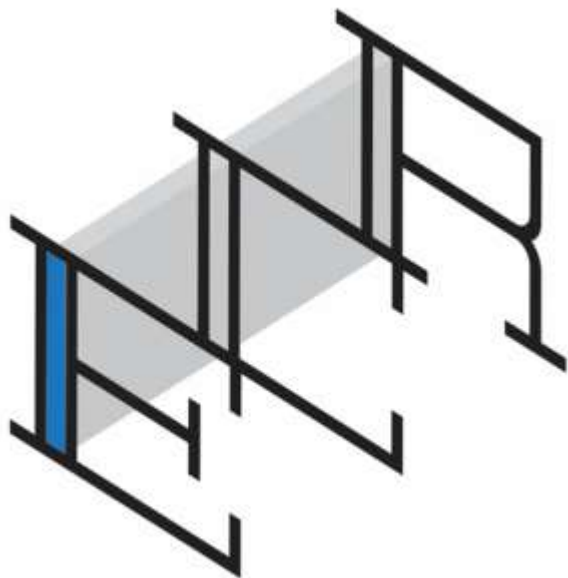
Conclusions

- We use non-determinism, deep learning, etc. when we don't know how something works...and the nominal performance of these techniques is excellent!
- But they confound traditional safety practices
- Smart choices about system architecture can help
- **Verification must aggressively pursue latent and unexpected safety risks**

Conclusions

- We use non-determinism, deep learning, etc. when we don't know how something works...and the nominal performance of these techniques is excellent!
- But they confound traditional safety practices
- Smart choices about system architecture can help
- Verification must aggressively pursue latent and unexpected safety risks
- **Call us, we can help :)**

Contact Information



MAKE **ROBUST** SOFTWARE

Michael Wagner

Co-founder and CEO

Edge Case Research LLC

The Ice House Building

100 43rd St, Suite 208

Pittsburgh, PA 15201

USA

412-606-3842

mwagner@ecr.guru

www.ecr.guru

References

(Anderson-Sprecher, 2011) P. Anderson-Sprecher, "Intelligent Monitoring of Assembly Operations", master's thesis, tech. report CMU-RI-TR-12-03, Carnegie Mellon University, 2011.

(Choudhury et al., 2015) S. Choudhury et al., "The planner ensemble: Motion planning by executing diverse algorithms", ICRA 2015.

(Felzenszwalb et al., 2010) P. Felzenszwalb et al., "Object detection with discriminatively trained part-based models", IEEE Transactions on Pattern Analysis and Machine Intelligence, 2010.

(Mordvintsev et al., 2015) A. Mordvintsev et al., "Inceptionism: Going Deeper into Neural Networks", Google research blog: <https://research.googleblog.com/2015/06/inceptionism-going-deeper-into-neural.html>

(Koopman and Wagner, 2016) P. Koopman and M. Wagner, "Challenges in Autonomous Vehicle Testing and Validation," SAE Int. J. Trans. Safety 4(1):2016, doi:10.4271/2016-01-0128

(Nguyen et al., 2015) A. Nguyen et al., "Deep Neural Networks are Easily Fooled: High Confidence Predictions for Unrecognizable Images", CVPR 2015.

(Sapunov, 2016) G. Sapunov, "Computer Vision and Deep Learning", Founders & Developers Meetup, Moscow, April 3 2016.

(Szegedy et al., 2013) C. Szegedy et al. "Intriguing Properties of Neural Networks," arXiv preprint arXiv:1312.6199 (2013).

Backup: Standards Requirements (1)

ISO 15066

- b) In variable speed setting situations, the speeds of the robot system and of the operator are used to determine the applicable value for the protective separation distance at each instant. Alternatively, the maximum allowed robot speed can be determined based on operator speed and actual separation distance between the robot and operator. The control function to accomplish this shall comply with ISO 10218-2:2011, 5.2.2.

ISO 10218

5.2.2 Performance requirement

Safety-related parts of control systems shall be designed so that they comply with PL=d with structure category 3 as described in ISO 13849-1:2006, or so that they comply with SIL 2 with hardware fault tolerance of 1 with a proof test interval of not less than 20 years as described in IEC 62061:2005.

This means in particular:

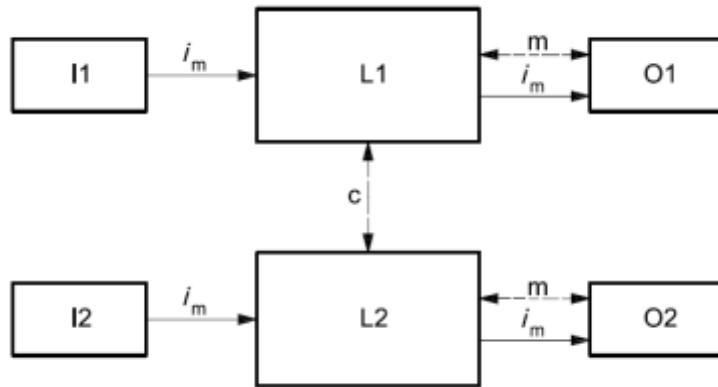
- a) a single fault in any of these parts does not lead to the loss of the safety function,
- b) whenever reasonably practicable, the single fault shall be detected at or before the next demand upon the safety function,
- c) when the single fault occurs, the safety function is always performed and a safe state shall be maintained until the detected fault is corrected,
- d) all reasonably foreseeable faults shall be detected.

The requirements a) to d) are considered to be equivalent to structure category 3 as described in ISO 13849-1:2006.

NOTE The requirement of single-fault detection does not mean that all faults will be detected. Consequently, the accumulation of undetected faults can lead to an unintended output and a hazardous situation at the machine.

Backup: Standards Requirements (2)

ISO 13849



Category 3 Architecture

Table 7 — Simplified procedure for evaluating PL achieved by SRP/CS

Category	B	1	2	2	3	3	4
DC _{avg}	none	none	low	medium	low	medium	high
MTTF _d of each channel							
Low	a	Not covered	a	b	b	c	Not covered
Medium	b	Not covered	b	c	c	d	Not covered
High	Not covered	c	c	d	d	d	e



Software Testing

We configure our robustness-testing tools to work with your software system and develop testing strategies tailored for your software, its application, and the processes in which it is being developed. We can help your development team analyze bugs and other vulnerabilities found, and suggest improvements to avoid problems in the future.



Training Services

ECR has conducted over 200 training engagements across a variety of different companies, industries, countries, and cultures. Our trainers are a mix of PhD researchers and senior software engineers, experts in the rigorous embedded software development processes that your development teams require to succeed.



Functional Safety for Autonomous Systems

Edge Case Research has executed multiple functional safety deployments of autonomous vehicles and robotics, and can help you do so as well. Our team has a deep background in this area, with multiple members of our team having over a decade of experience developing and testing autonomous robots and vehicles.