Edge Cases and Autonomous Vehicle Safety

SSS 2019, Bristol
7 February 2019

Prof. Philip Koopman

Carnegie Mellon University

@PhilKoopman

Edge Case Research
Overview

- Making safe robots
  - Doer/Checker safety

- Edge cases matter
  - Robust perception matters

- The heavy tail distribution
  - Fixing stuff you see in testing isn’t enough

- Perception stress testing
  - Finding the weaknesses in perception

- UL 4600: autonomy safety standard
98% Solved For 20+ Years

- Washington DC to San Diego
  - CMU Navlab 5
  - Dean Pomerleau
  - Todd Jochem

- AHS San Diego demo Aug 1997

TRIP COMPLETE !!!
2797/2849 miles (98.2%)
NREC: 30+ Years Of Cool Robots

Carnegie Mellon University Faculty, staff, students
Off-campus Robotics Institute facility

Software Safety
Before Autonomy Software Safety

- The Big Red Button era
Traditional Validation Meets Machine Learning

- Use traditional software safety where you can

..BUT..

- Machine Learning (inductive training)
  - No requirements
    - Training data is difficult to validate
  - No design insight
    - Generally inscrutable; prone to gaming and brittleness
APD (Autonomous Platform Demonstrator)

TARGET GVW: 8,500 kg
TARGET SPEED: 80 km/hr

Approved for Public Release. TACOM Case #20247 Date: 07 OCT 2009

Safety critical speed limit enforcement
Safety Envelope Approach to ML Deployment

- Specify unsafe regions

- Specify safe regions
  - Under-approximate to simplify

- Trigger system safety response upon transition to unsafe region
Architecting A Safety Envelope System

- **“Doer” subsystem**
  - Implements normal, untrusted functionality

- **“Checker” subsystem – Traditional SW**
  - Implements failsafes (safety functions)

- **Checker entirely responsible for safety**
  - Doer can be at low Safety Integrity Level
  - Checker must be at higher SIL

(Also known as a “safety bag” approach)
Validating an Autonomous Vehicle Pipeline

Perception presents a uniquely difficult assurance challenge
Validation Via Brute Force Road Testing?

- If 100M miles/critical mishap...
  - Test 3x–10x longer than mishap rate
  - Need 1 Billion miles of testing

- That’s ~25 round trips on every road in the world
  - With fewer than 10 critical mishaps...
Brute Force AV Validation: Public Road Testing

- Good for identifying “easy” cases
  - Expensive and potentially dangerous


© 2019 Philip Koopman
Closed Course Testing

- Safer, but expensive
  - Not scalable
  - Only tests things you have thought of!
Simulation

- Highly scalable; less expensive
  - Scalable; need to manage fidelity vs. cost
  - Only tests things you have thought of!


Udacity

Apollo

http://bit.ly/2toFdeT
What About Edge Cases?

- You should expect the extreme, weird, unusual
  - Unusual road obstacles
  - Extreme weather
  - Strange behaviors

- Edge Case are surprises
  - You won’t see these in testing
    ➔ Edge cases are the stuff you didn’t think of!

© 2019 Philip Koopman
Just A Few Edge Cases

- Unusual road obstacles & obstacles
- Extreme weather
- Strange behaviors

https://dailym.ai/2K7kNS8
https://en.wikipedia.org/wiki/Magic_Roundabout_(Swindon)
https://goo.gl/J3SSYu
Why Edge Cases Matter

Where will you be after 1 Billion miles of validation testing?

Assume 1 Million miles between unsafe “surprises”

- Example #1:
  100 “surprises” @ 100M miles / surprise
  - All surprises seen about 10 times during testing
  - With luck, all bugs are fixed

- Example #2:
  100,000 “surprises” @ 100B miles / surprise
  - Only 1% of surprises seen during 1B mile testing
  - Bug fixes give no real improvement (1.01M miles / surprise)

https://goo.gl/3dzguf
The Real World: Heavy Tail Distribution(?)

- **Common Things Seen In Testing**
- **Edge Cases Not Seen In Testing**

**Random Independent Arrival Rate (exponential)**

**Power Law Arrival Rate (80/20 rule)** (Heavy Tail Distribution)

Many *Different*, Infrequent Scenarios Total Area is the same!
Need to find “Triggering Events” to inject into sims/testing
Need to collect surprises
- Novel objects
- Novel operational conditions

Corner Cases vs. Edge Cases
- Corner cases: infrequent combinations
  - Not all corner cases are edge cases
- Edge cases: combinations that behave unexpectedly

Issue: novel for person ≠ novel for Machine Learning
- ML can have “edges” in unexpected places
- ML might train on features that seem irrelevant to people

Edge Cases Part 1: Triggering Event Zoo

https://goo.gl/Ni9HhU
What We’re Learning With Hologram

- A scalable way to test & train on Edge Cases

Your fleet and your data lake → Hologram cluster tests your CNN → Hologram cluster identifies weaknesses & helps retrain your CNN → Your CNN becomes more robust
Edge Cases Part 2: Brittleness

Malicious Image Attacks Reveal Brittleness:

QuocNet:  
Car  Not a Car  Magnified Difference

AlexNet:  
Bus  Magnified Difference  Not a Bus

ML Is Brittle To Environment Changes

- Sensor data corruption experiments

**Contextual Mutators**

- Defocus & haze are a significant issue

Exploring the response of a DNN to environmental perturbations from “Robustness Testing for Perception Systems,” RIOT Project, NREC, DIST-A.
Context-Dependent Perception Failures

- Perception failures are often context-dependent
  - False positives and false negatives are both a problem

Will this pass a “vision test” for bicyclists?
Example Triggering Events via Hologram

Mask-R CNN: examples of systemic problems we found

- "Camouflage"
- "Children"
- "Bare legs"
- "Red objects"
- "Columns"
- "Sun glare"
- "Single Lane Control"

Notes: These are baseline, un-augmented images. (Your mileage may vary on your own trained neural network.)
Centered on a Safety Case
- Credit for existing safety standards
- Mix & match cross-standards techniques
- Discourages questionable practices

“Unknowns” are first class citizens
- Balance between analysis & field experience
- Field monitoring required to feed back to argumentation
- Assessment findings & field data used to update standard

Plan: public standard by end of 2019
- Standards Technical Panel forming Spring 2019 to review draft
List of pitfalls we've seen in autonomy safety arguments

- Conformance to an existing standard
- Proven in Use
- Field Testing
- Vehicle simulation
- Formal proof of correctness
- Other issues (e.g., unwarranted independence assumptions)

Also, SafeAI Paper on taxonomy of ODD/OEDR

Ways To Improve AV Safety

- More safety transparency
  - Independent safety assessments
  - Industry collaboration on safety

- Minimum performance standards
  - Share data on scenarios and obstacles
  - Safety for on-road testing (driver & vehicle)

- Autonomy software safety standards
  - Traditional software safety ... PLUS ...
  - Dealing with surprises and brittleness
  - Data collection and feedback on field failures

Thanks!