Systematic Analysis, Testing, and Improvement of CPSML

Tommaso Dreossi

Joint work with:

Daniel Fremont, Shromona Ghosh, Xiangyu Yue, Alexandre Donze

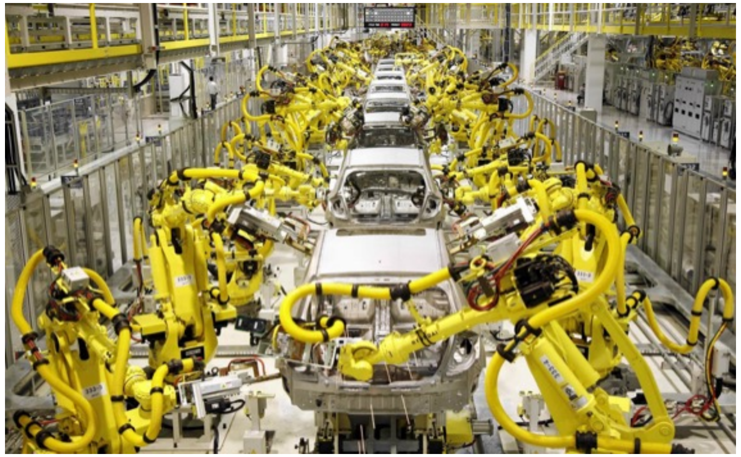
Kurt Keutzer, Alberto Sangiovanni-Vincentelli, Sanjit A. Seshia

UC Berkeley

Cyber-Physical Systems (CPS)

Integration of computation with physical processes

Building systems



Factory automation

Automotive

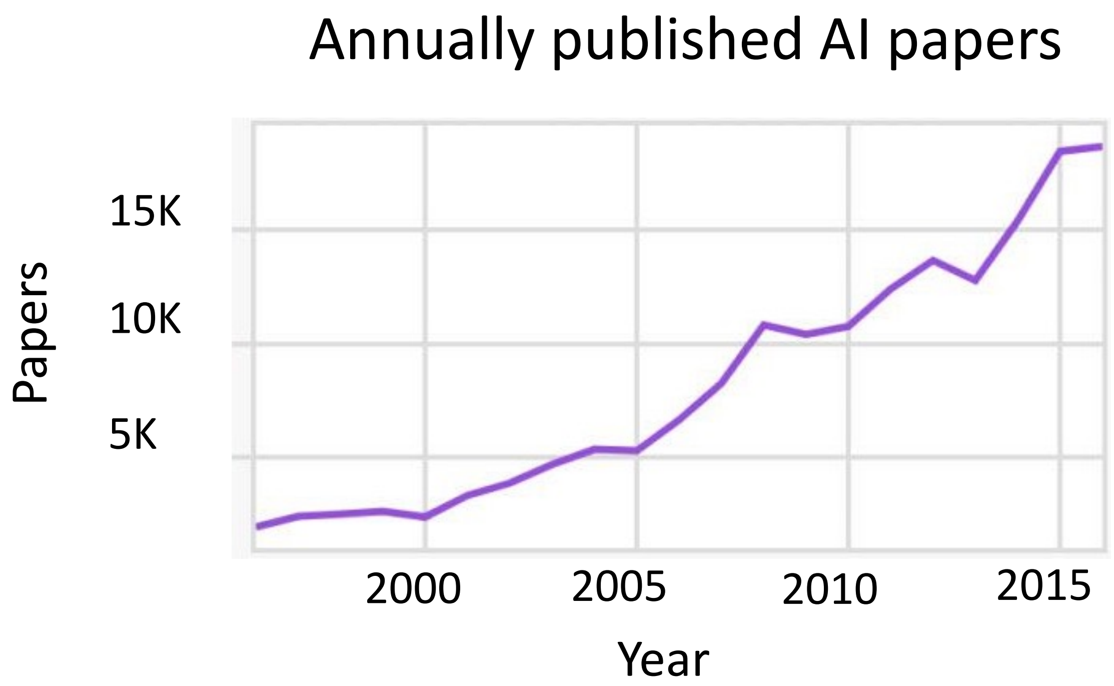
Power generation

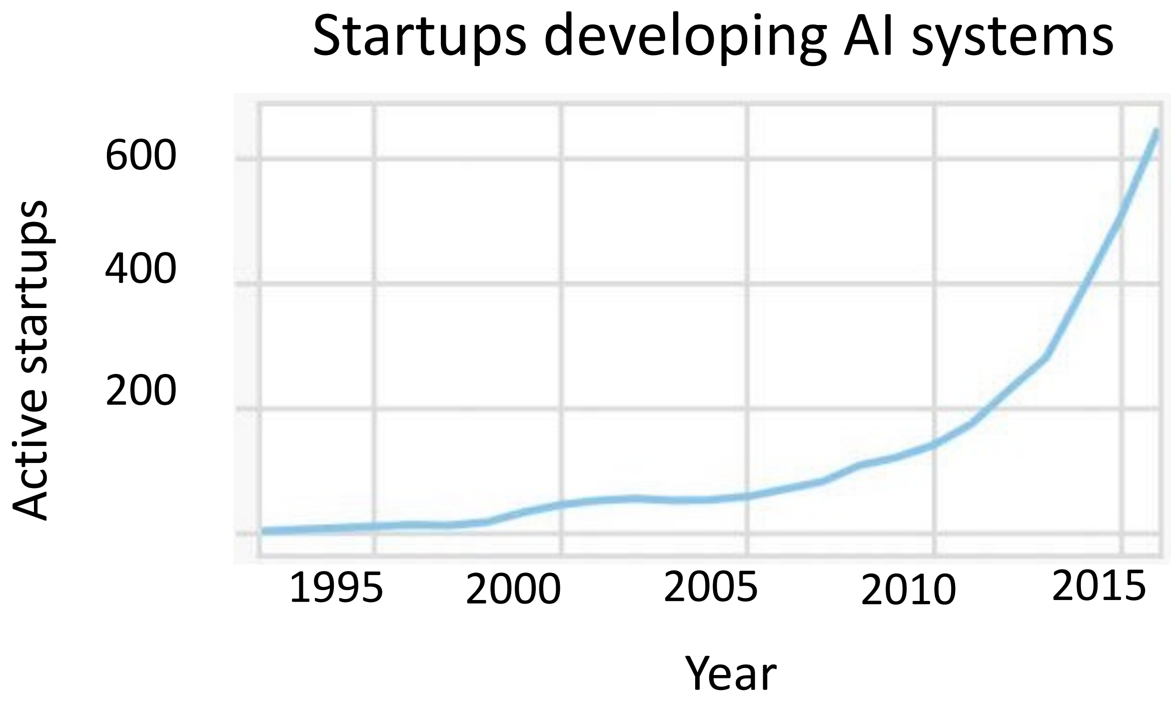
Avionics

Smart cities

Cyber-Physical Systems + ML/AI (CPSML)

Growing use of Machine Learning/AI in CPS





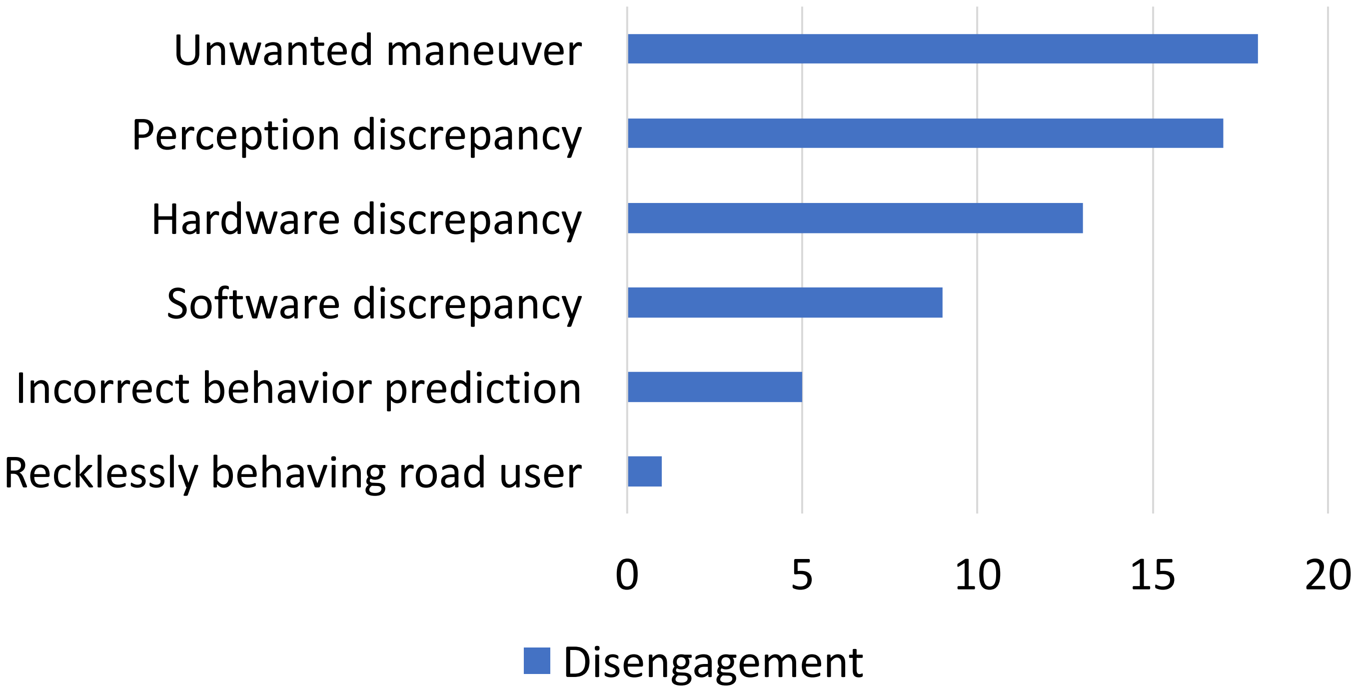
Source: AIindex.org, Scopus, Crunchbase, VentureSource, Sand Hills Econometrics

Cyber-Physical Systems + ML/AI (CPSML)

Growing use of Machine Learning/AI in CPS

Many safety-critical applications

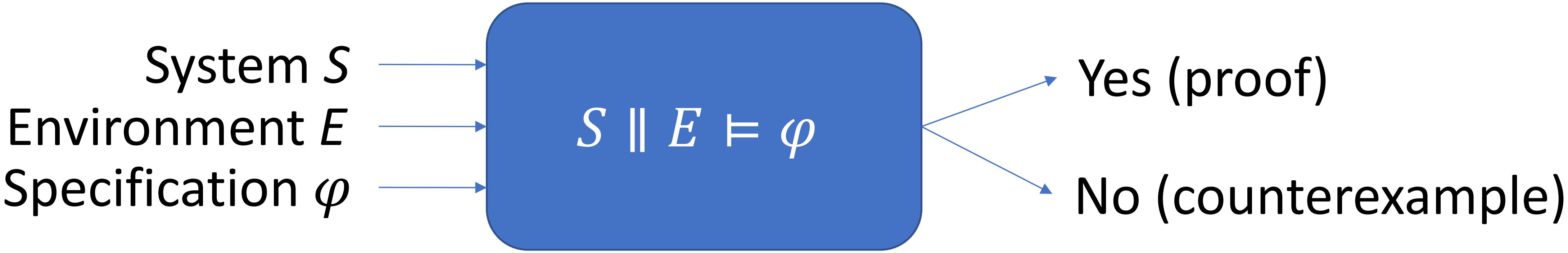
Waymo disengagement report
California, 2017



Source: DMV CA

Challenges for Verified AI

Formal methods approach



Challenges for Verified AI

Formal methods approach

System S

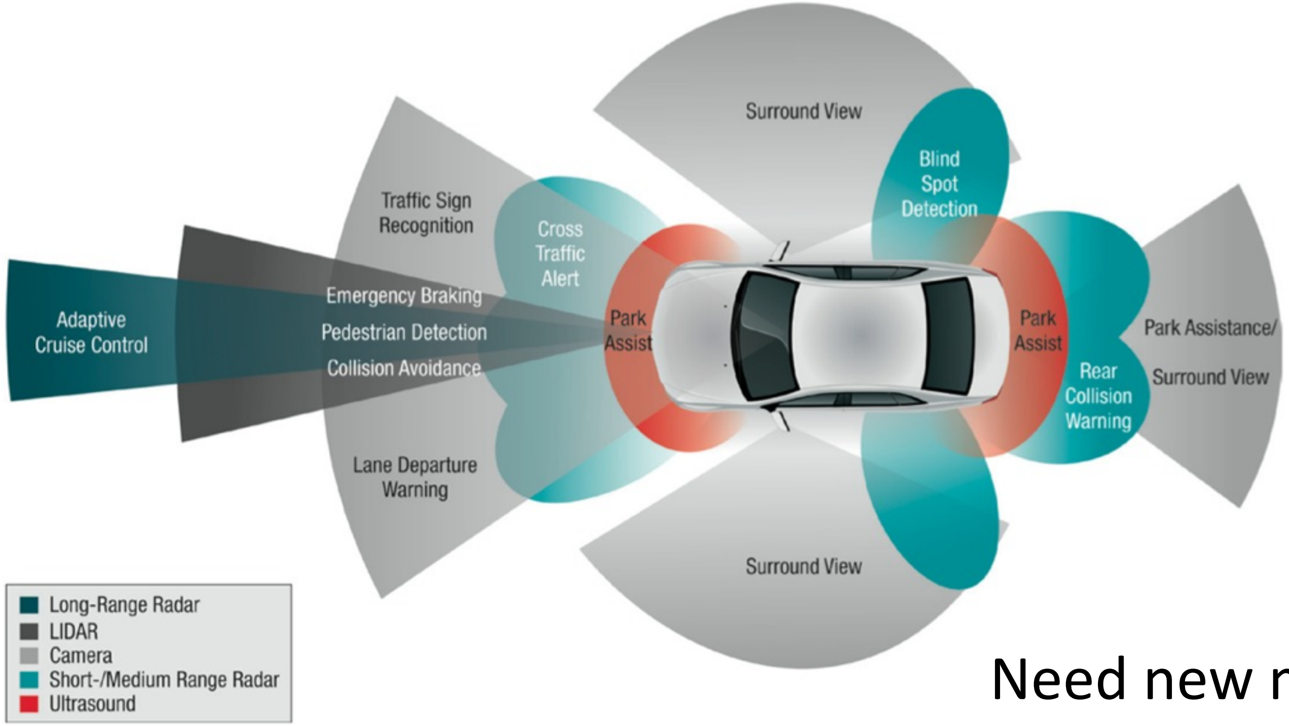
Environment E

Specification φ

$$S \parallel E \vDash \varphi$$

Yes (proof)

No (counterexample)



- Complex models
- E.g., AlexNet, 60M parameters, 650K neurons)

- Large input spaces
- E.g., KITTI images: 256^(1392x512x3)

Need new methods for *Abstraction* and *Modular Reasoning*

Challenges for Verified AI

Formal methods approach

System S

Environment E

Specification φ

$$S \parallel E \vDash \varphi$$

Yes (proof)

No (counterexample)

- Interaction with complex environments/agents

Need for representing *environment scenarios*

Challenges for Verified AI

Formal methods approach

System S

Environment E

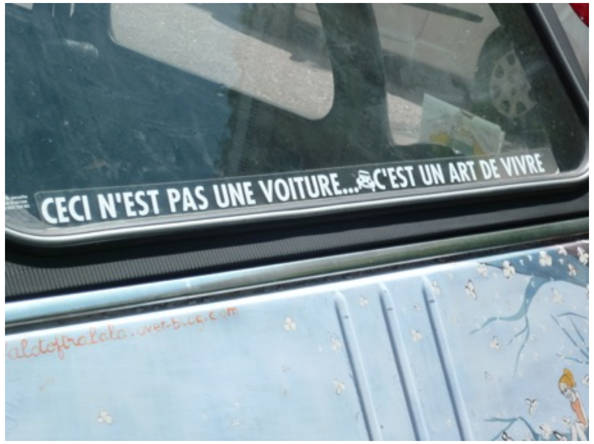
Specification φ

$$S \parallel E \vDash \varphi$$

Yes (proof)

No (counterexample)

- How do you formalize perception tasks?



Need for new *specification formalisms*

Challenges for Verified AI

Formal methods approach

System S
Environment E
Specification φ

$$S \parallel E \vDash \varphi$$

Yes (proof)
No (counterexample)

Our approach:

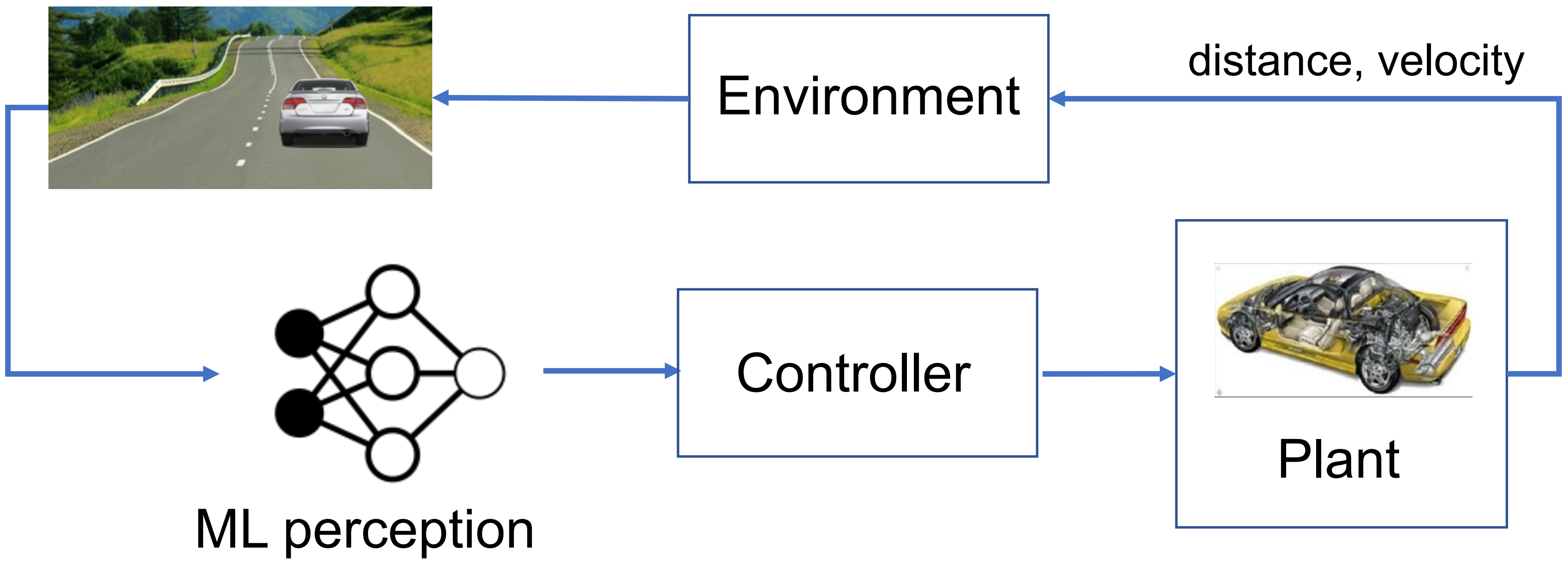
- System:
  - Compositional analysis of CPS-ML
  - Abstraction of ML modules input space
- Environment
  - Scenic – Scenario description language
- Specification
  - System-level specifications

Outline

1. Running CPSML example – Automatic emergency braking system
2. Specification
    - System- vs Module-level specification
3. System
    - Compositional falsification
    - ML input abstraction
    - Counterexample-guided augmentation
4. Environment
    - Scenic: Scenario description language
5. Conclusion

CPSML Example

Automatic Emergency Braking System (AEBS)

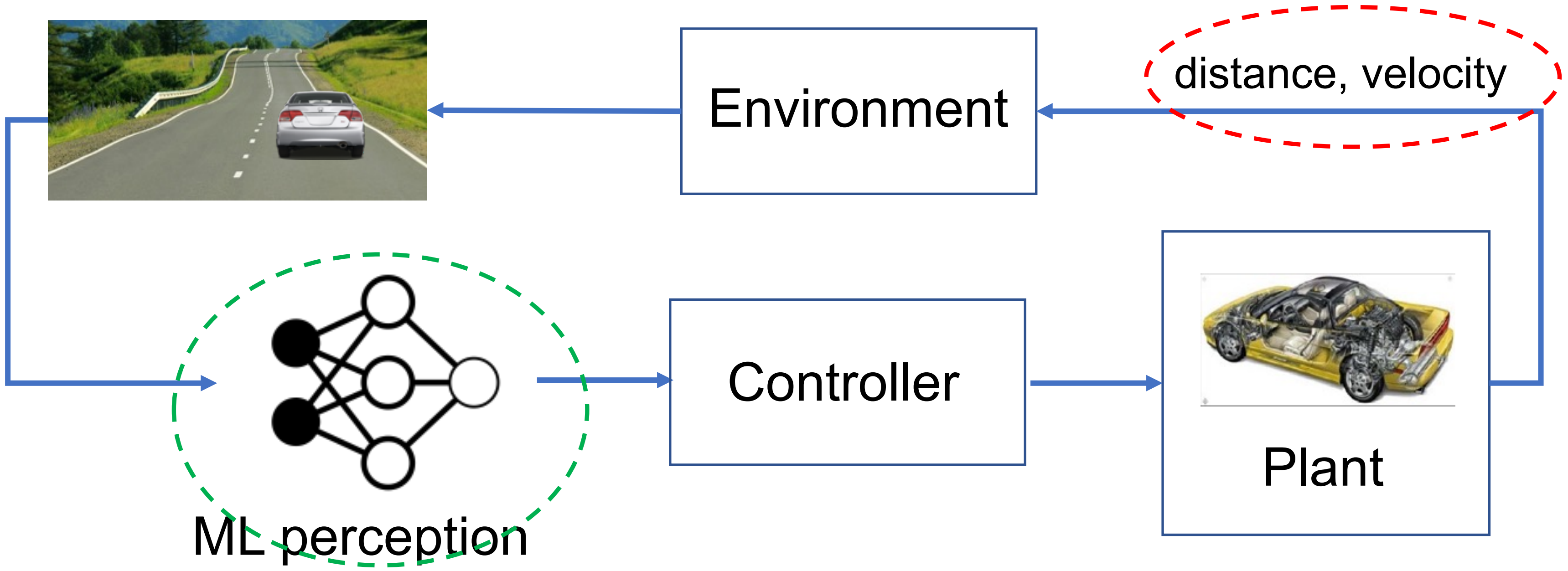


- Goal: brake when an obstacle is near

- Challenges:
  - How to explore distance/velocity?
  - How to analyze images?
  - How to combine distance, velocity, images?

Specification

System- vs Module-level Specification



- Goal: brake when an obstacle is near
- Specifications:
  - "Never collide" (distance > 0)
  - "Correctly detect obstacles"

Outline

1. Running CPSML example – Automatic emergency braking system
2. Specification
    - System- vs Module-level specification
3. System
    - Compositional falsification
    - ML input abstraction
    - Counterexample-guided augmentation
4. Environment
    - Scenic: Scenario description language
5. Conclusion

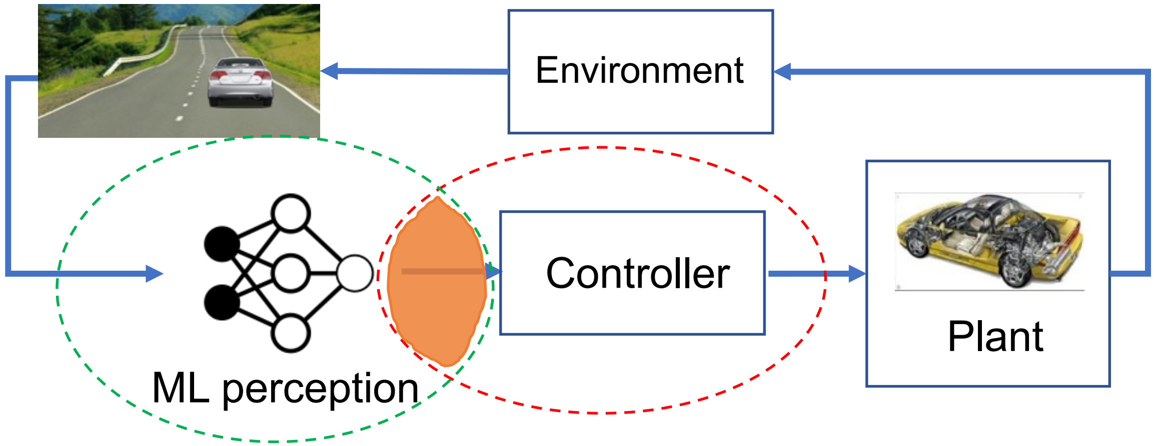
Compositional Falsification

CPSML input space intractable

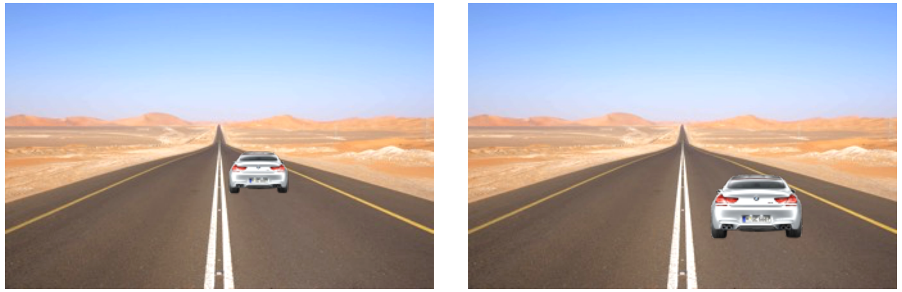
- Idea: focus on meaningful CPS+ML input combinations
- Intuition: "If car is far, misclassification won't affect our system"

Strategy:

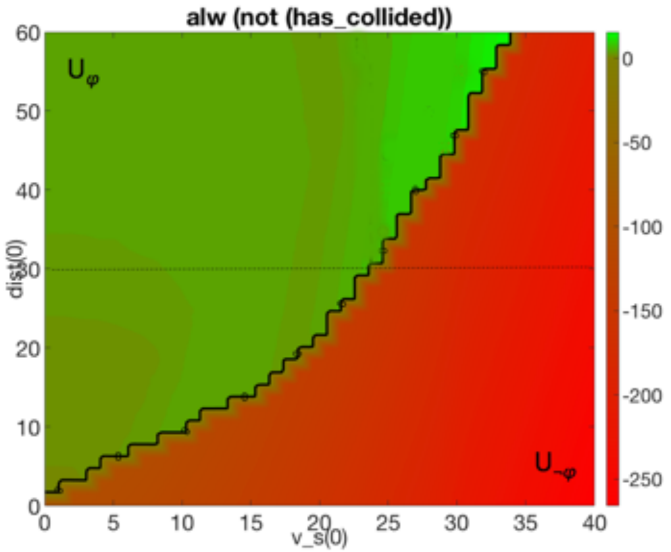
1. Analyze CPS gathering info on ML role
2. Use collected info to target ML
3. Compose CPS + ML narrowed input spaces
4. Perform targeted falsification



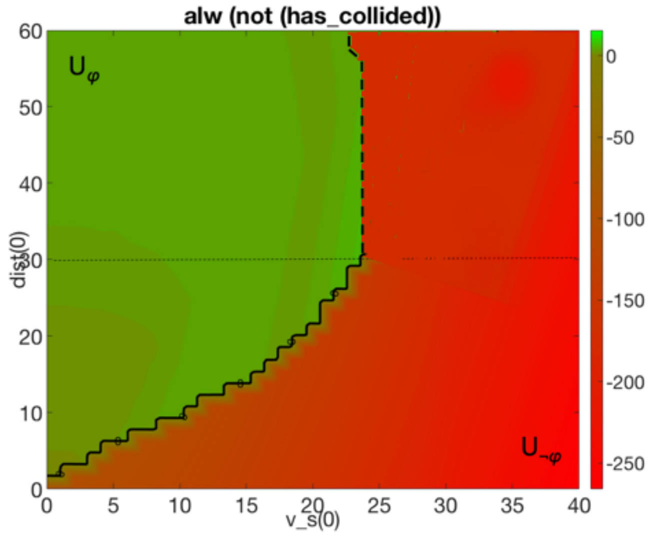
Compositional Falsification



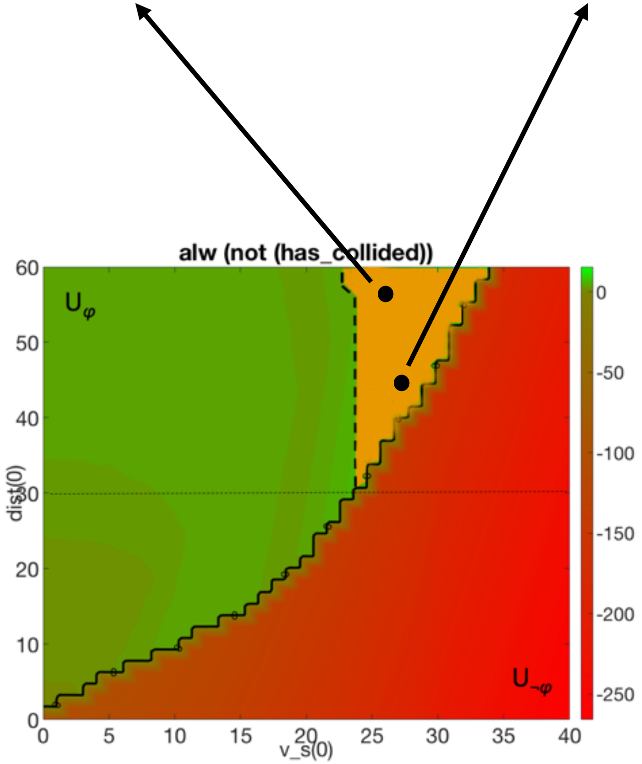
- Identifying regions of interest for AEBS
- Perform optimistic/pessimistic analyses of NN



ML correct



ML wrong



Potentially unsafe region
(depending on ML)

Dreossi et. al, Compositional Falsification of Cyber-Physical Systems with Machine Learning Components, NFM 2017

ML Analyzer

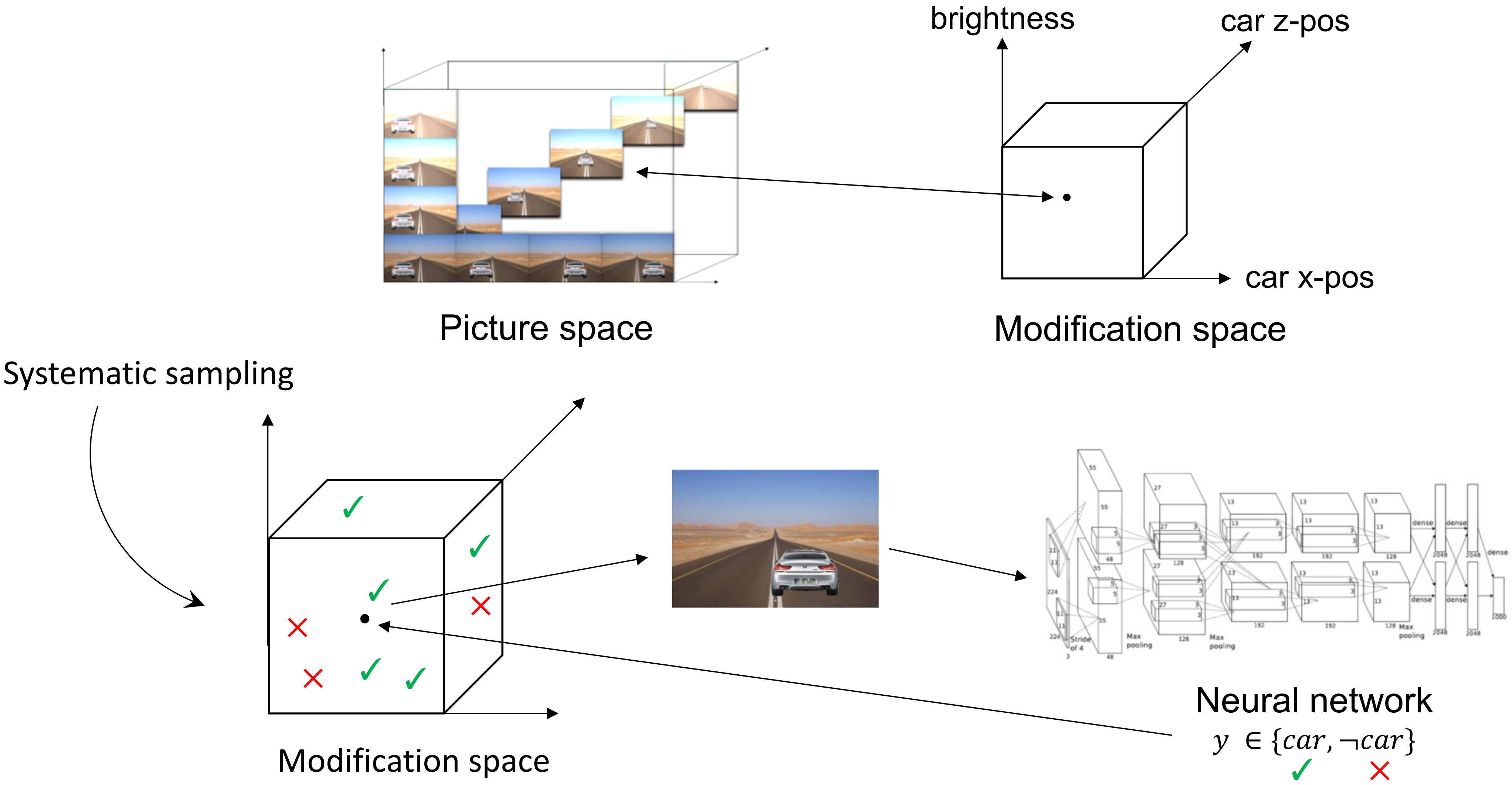
- How analyze ML feature space?
- E.g., image classifier: a lot of pictures to analyze
- Idea: Focus on semantic alterations

Plausible alterations

Dreossi et. al, Systematic Testing of Convolutional Neural Networks for Autonomous Driving, RMLW 2017

ML Analyzer

Systematically analyze modifications of interest



ML Analyzer

Sampling methods

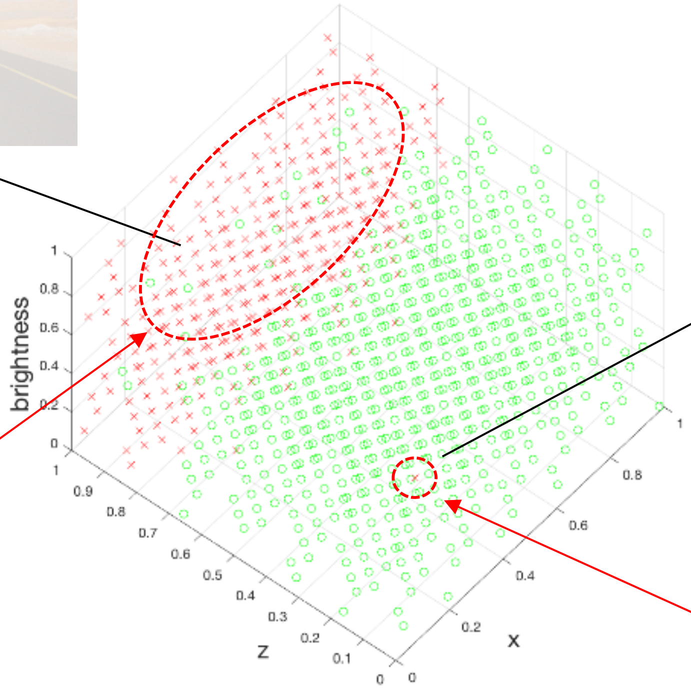
Method	Sampling speed	Diversity	Counterexample finding
Uniform random	✓	✕	✕
Uniform random + distance constraint	✕	–	✕
Low-discrepancy	✕	✓	–
Cross entropy	✕	✕	✓

H. Niederreiter, "*Random Number Generation and Quasi-Monte-Carlo Methods*", 1992
R. Y. Rubinstein et al., "*The Cross-Entropy Method, A Unified Approach to Combinatorial Optimization, Monte-Carlo Simulation, and Machine Learning*", 2004

Sample Results

AEBS

This misclassification may not be of concern



Misclassification cluster

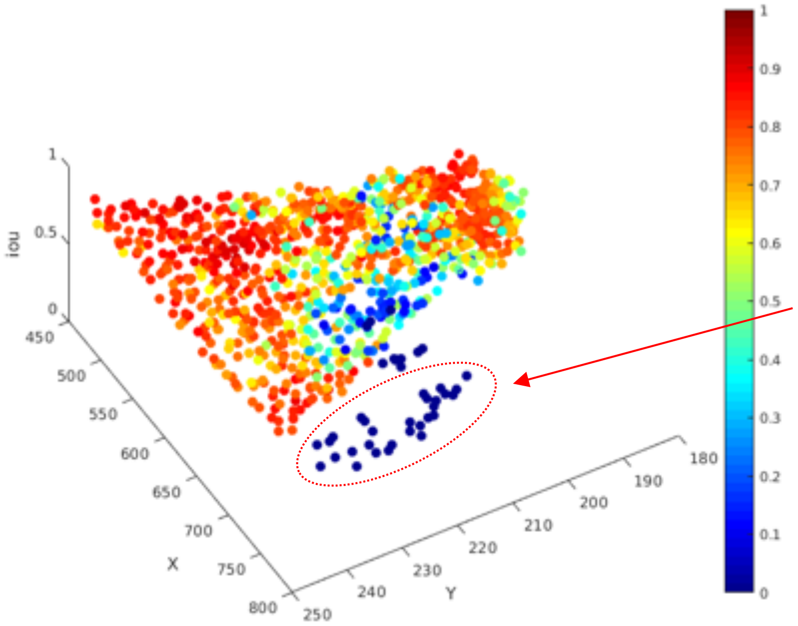
Corner case

But this one is a real hazard

Inception-v3 Neural Network
(pre-trained on ImageNet using TensorFlow)

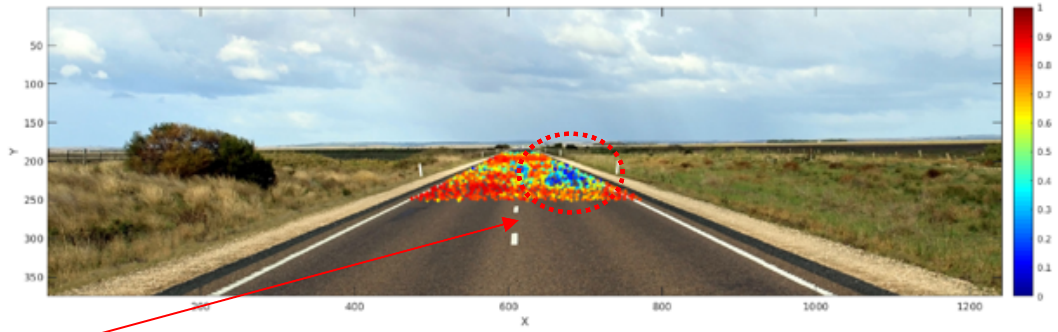
Sample Results

squeezeDet



squeezeDet
(trained on synthetic images)

Blind spot



Example of counterexamples

Counterexample-guided augmentation

- What to do with the generated counterexamples?

1. Analyze them and provide explanations (error tables)
2. Augment training sets

Misclassifications

Id	Car color	Background	Orientation
1	Red	Countryside	Front
2	Orange	Forest	Back
3	White	Forest	Front
4	Green	Forest	Back

Error table

Train

Test

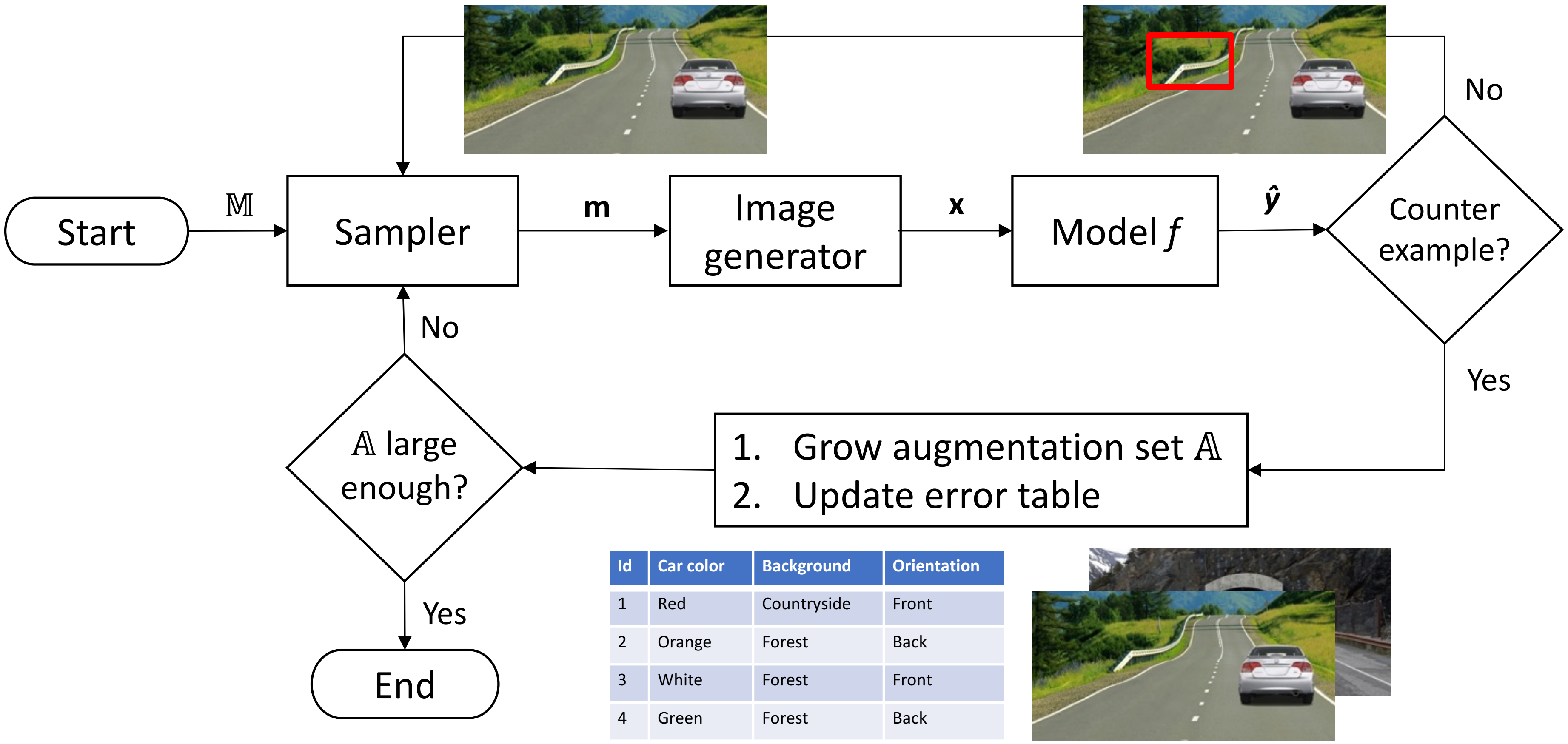
Test | Aug

Counterexamples

Dreossi et. al, Counterexample-Guided Data Augmentation, IJCAI 2018

Counterexample-guided augmentation

Find counterexamples and augment training set

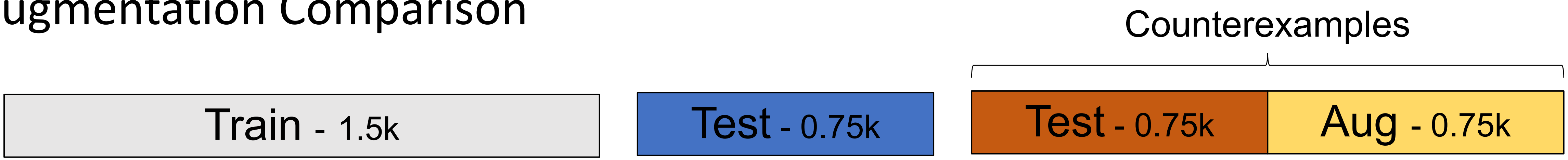


Id	Car color	Background	Orientation
1	Red	Countryside	Front
2	Orange	Forest	Back
3	White	Forest	Front
4	Green	Forest	Back

Dreossi et. al, Counterexample-Guided Data Augmentation, IJCAI 2018

Augmentation

Augmentation Comparison



Sampling methods comparison

Model	Precision	Recall	t (sec)
Original	.61	.75	
Standard augmentation	.69	.80	
Uniform random	.76	.87	~30
Constrain	.75	.86	~92
Low-discrepancy	.79	.87	~55
Cross-entropy	.78	.78	~70

Outline

1. Running CPSML example – Automatic emergency braking system
2. Specification
    - System- vs Module-level specification
3. System
    - Compositional falsification
    - ML input abstraction
    - Counterexample-guided augmentation
4. Environment
    - Scenic: Scenario description language
5. Conclusion

Environment Description

Idea: Use simulators to model environment (e.g., GTAV)

Problem

- Large and unstructured input space
- Generate meaningful scenes (for testing or training)

Car Model | Car Location | Car Orientation
Number of Cars | Reference | Scene Background
Car Color | Weather | Time of Day

Scenic

A Scenario Description Language

- Scenic: probabilistic programming language defining distributions over scenes
- Example: a badly parked car

```
from gta import Car, curb, roadDirection

ego = Car

spot = OrientedPoint on visible curb
badAngle = Uniform(1.0, -1.0) * (10, 20) deg
Car left of (spot offset by -0.5 @ 0),
    facing badAngle relative to roadDirection
```

Fremont et. al, Scenic: Language-Based Scene Generation, under review

Scenic Applications
Testing

Exploring the behavior of the system under different conditions:

Bright and clear weather

Dark and rainy weather

Scenic Applications

Training

Generate hard cases, e.g., one car partially occluding another:

```
from gta import Car

ego = Car with roadDeviation (-10, 10) deg

c = Car visible,
        with roadDeviation (-10, 10) deg

leftRight = Uniform(1.0, -1.0) * (1.25, 2.75)
Car beyond c by leftRight @ (4, 10),
    with roadDeviation (-10, 10) deg
```

Scenic Applications
Reasoning

Scenic Applications
Reasoning

Scenic Applications
Reasoning

Scenic makes it easy to generalize along different dimensions:

Add noise

Change car model

Change global position

Conclusion

Summary

- Framework for system-level counterexamples
- CNN analyzer (simulation based)
- Counter-example guided augmentation
- Scenic: Scenario description language

Future work

- Mix real/synthetic data
- Domain adaptation/randomization
- More complex data: lidar, radar, etc.