



RESILIENCE AND SAFETY FOR AUTONOMOUS SYSTEMS

Siva Hari, 03/28/2021



AUTONOMOUS VEHICLES

Significant advancements are being made

Vehicles are becoming increasingly autonomous

92.7% of new vehicles in the U.S. as of May 2018 have at least one Advanced Driver Assistance Systems feature

E.g., adaptive cruise control, blind spot detection

61 companies hold permits to test AVs with a driver*

5 companies hold permits to test AVs without a driver*

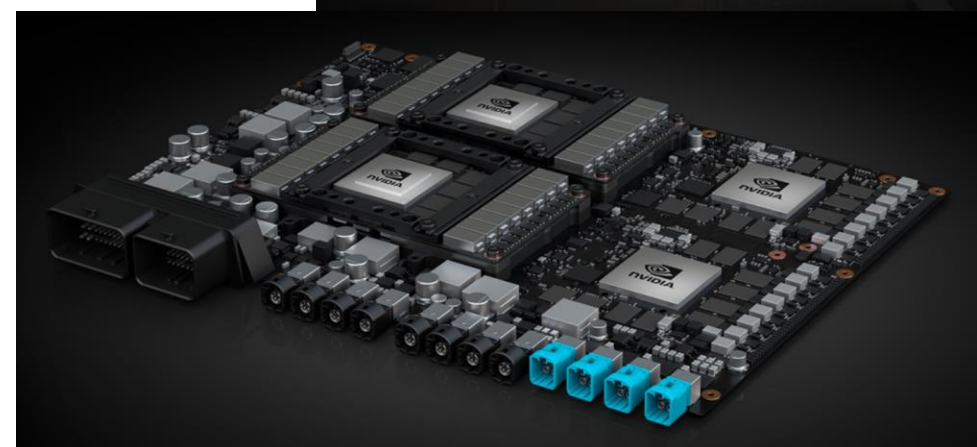
Significant advancements in sensor technology, processing (software), platforms (hardware), and simulation is driving the growth



LIDAR(s)
(Light Detection and Ranging)

Cameras

Radars
Ultrasonic sensors



AUTONOMOUS VEHICLE SYSTEMS

Simple view

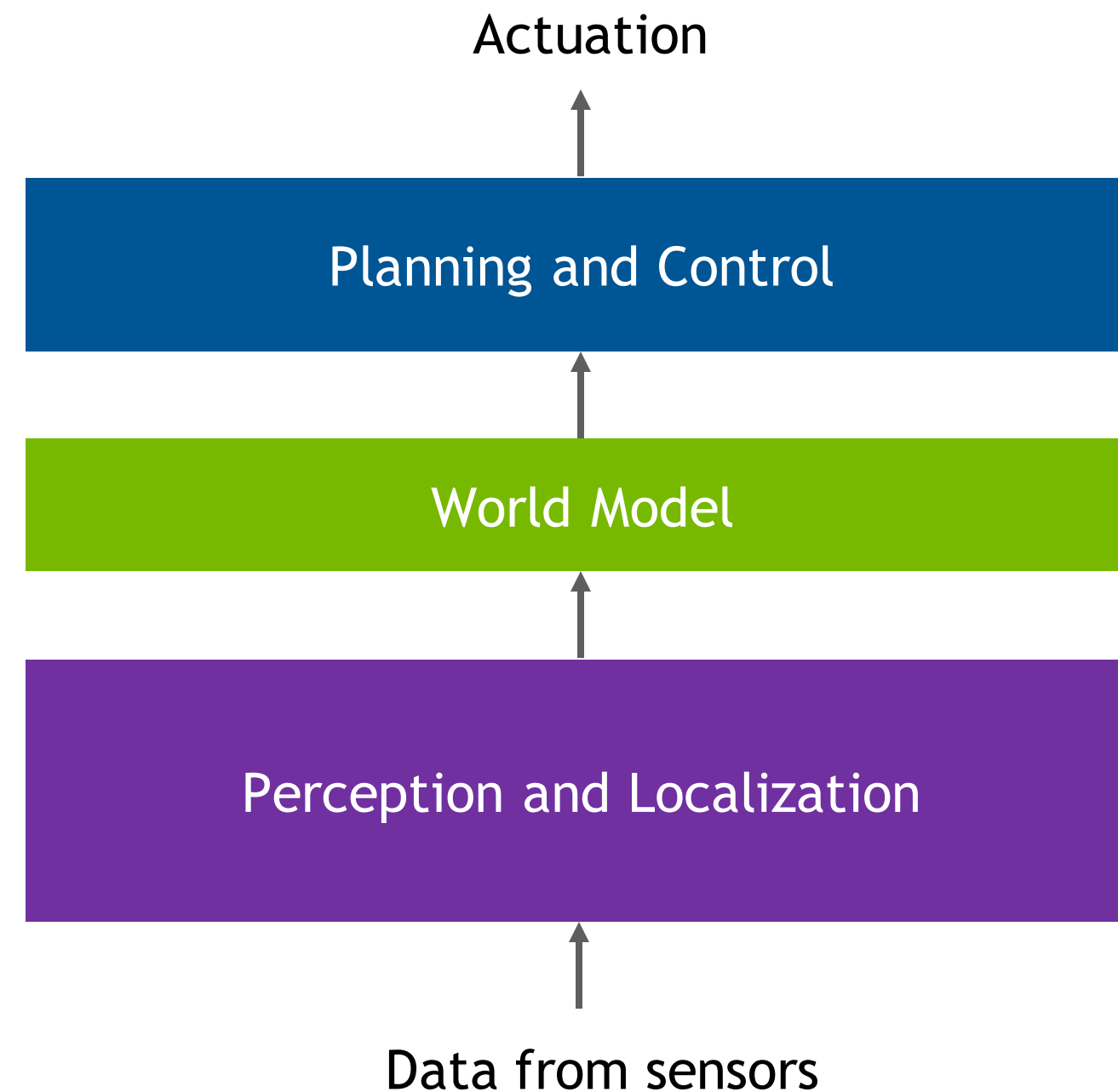
AV tasks

Perceive world

Localize

Plan

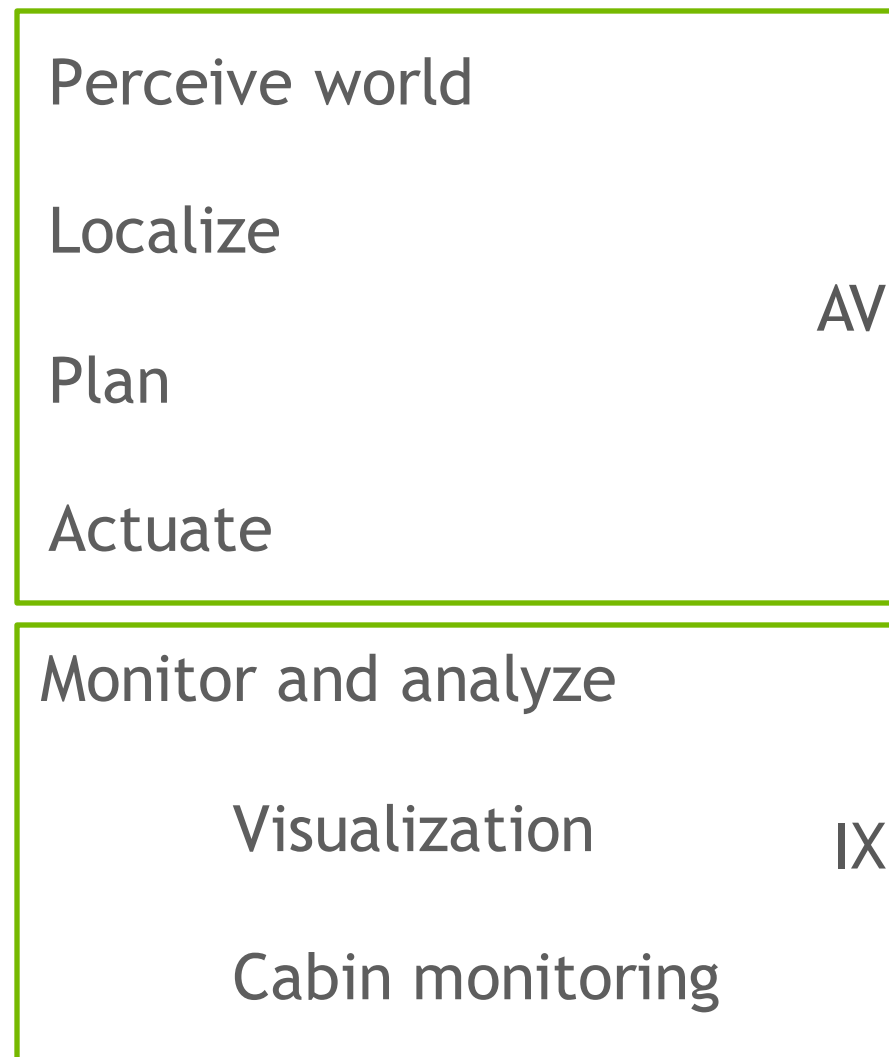
Actuate



AUTONOMOUS VEHICLE SYSTEMS

Not-so-simple view

AV tasks



AV SAFETY

Landscape

Safety is a key requirement for autonomous vehicles

What if something goes wrong

Hardware Systematic faults

Hardware Random faults (permanent and transient)

Functionality

Testing the functionality of based on the desired automation level

Sufficient computational and memory resources to meet real-time deadlines

This talk

ISO standards (21448 and 26262) have been established to guide the engineering effort

Several interesting research directions emerge



OUTLINE

Hardware Resilience

Functionality Testing

Designing Efficient and Safer System

HARDWARE RESILIENCE

Problem

Systems certified with ISO 26262 are required to be robust to single-point transient, intermittent, and permanent faults either by design or by coverage from safety procedures.

	ASIL	FiT	SPFM	LFM
	A	Irrelevant	Irrelevant	Irrelevant
	B	< 1000	> 90%	> 60%
	C	< 100	> 97%	> 80%
Desired	D	< 10	> 99%	> 90%

Source: An ISO 26262 Automotive Semiconductor Safety Primer, Optima

Different customers buy systems or components and would like the requirements to be taken care off by the vendors



CNN RESILIENCE EVALUATION

AVs

Simulation of hardware errors while running CNNs

Tools for evaluation (open-source):

NVBitFI: Dynamic GPU assembly instruction-level injector [DSN'21]

PyTorchFI: Inject errors in convolution outputs during inference [DSML'20]

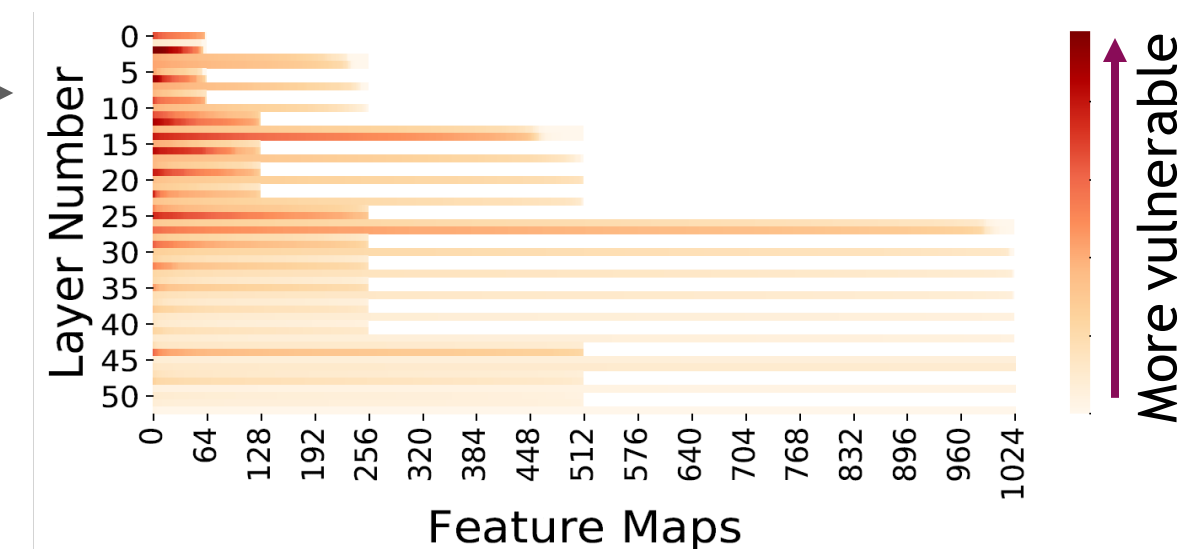
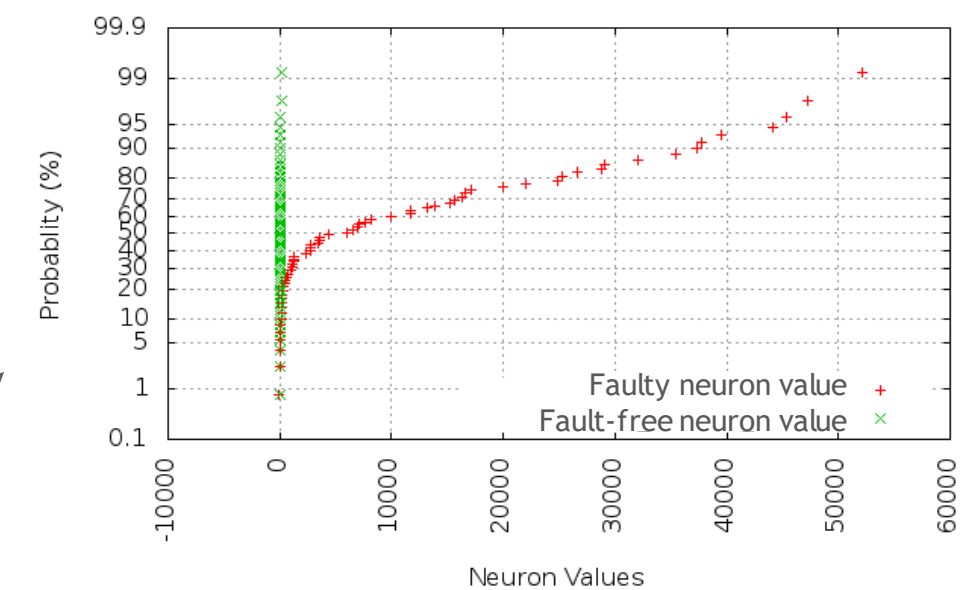
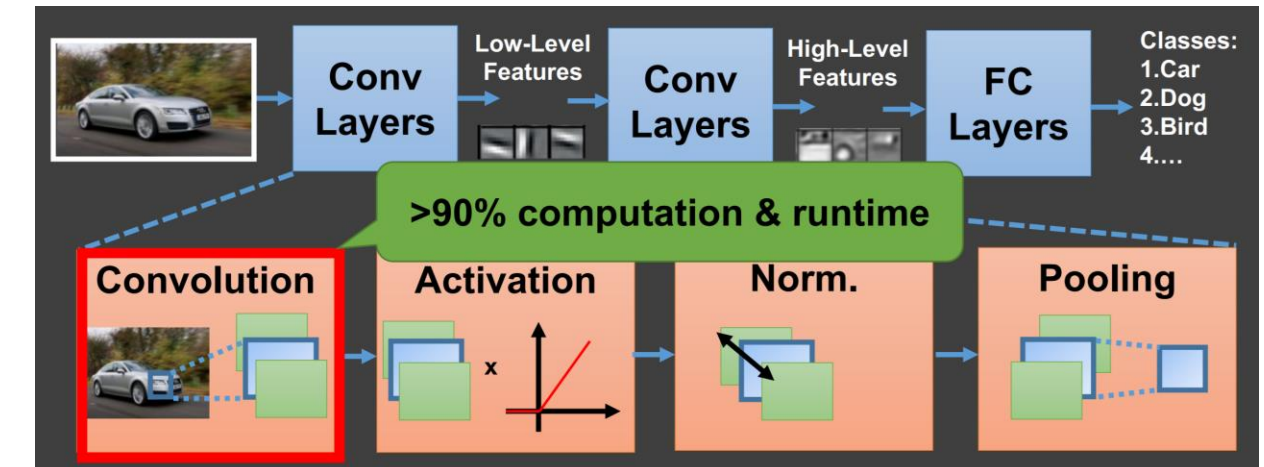
Key findings:

Large deviations in neurons is the main cause of output corruptions [SC'17]

Some feature maps are more vulnerable than others [SARA'20]

High confidence images are less vulnerable

Full AV-level evaluations show dependence on scenario [DSN'19]



ERROR MITIGATION METHODS

Research options to address the resilience challenge

Hardware-only

Selective latch hardening, ECC/parity for DNN accelerator components [SC'17]

CNN framework-level

Neuron range checks/clips (Clipped ReLU) [SC'17]

Selective fmap protection [SARA'20]

Algorithm-based error detection [TDSC'21]



Verify convolutions with checksums:

Use distributive property

Example: $a*b + a*c + a*d \rightarrow a * (b+c+d)$, eliminates 2 multiplies!

Applied it to convolutions


Only 6%-24% overhead << full duplication (100%)

Perception stack-level

Selective inference protection

Temporal perception results smoothing

Leverage diversity with robust sensor fusion



FUNCTIONAL TESTING: SCENARIO GENERATION AND CHARACTERIZATION

FUNCTIONAL TESTING FOR AV SAFETY

Scenario-based end-to-end AV testing

AVs need to be tested to ensure safe operation

Simulation plays a key role to augment on-road testing

Goal of testing:

- AV is following driving rules and etiquette

- Safety goals derived from the HARA process (part of ISO 26262)

- Pre-crash situations based on human accident statistics

- Corner cases that are challenging for a given AV but can theoretically be navigated safely

Next slide

Lots of interesting problems are yet to be solved!

GENERATING CORNER CASE SCENARIOS

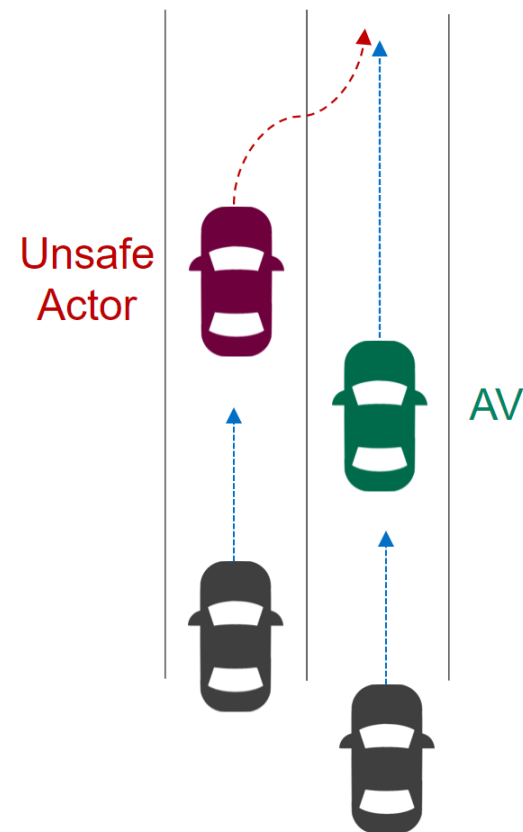
Two methods

Approach 1

Start from a random scenario; increase the probability of an accident

Actor(s) close to the AV violate safety for a limited time

Alter the unsafe actor's trajectory to create an accident with the AV



Approach 2

Employ genetic algorithm to find challenging scenarios

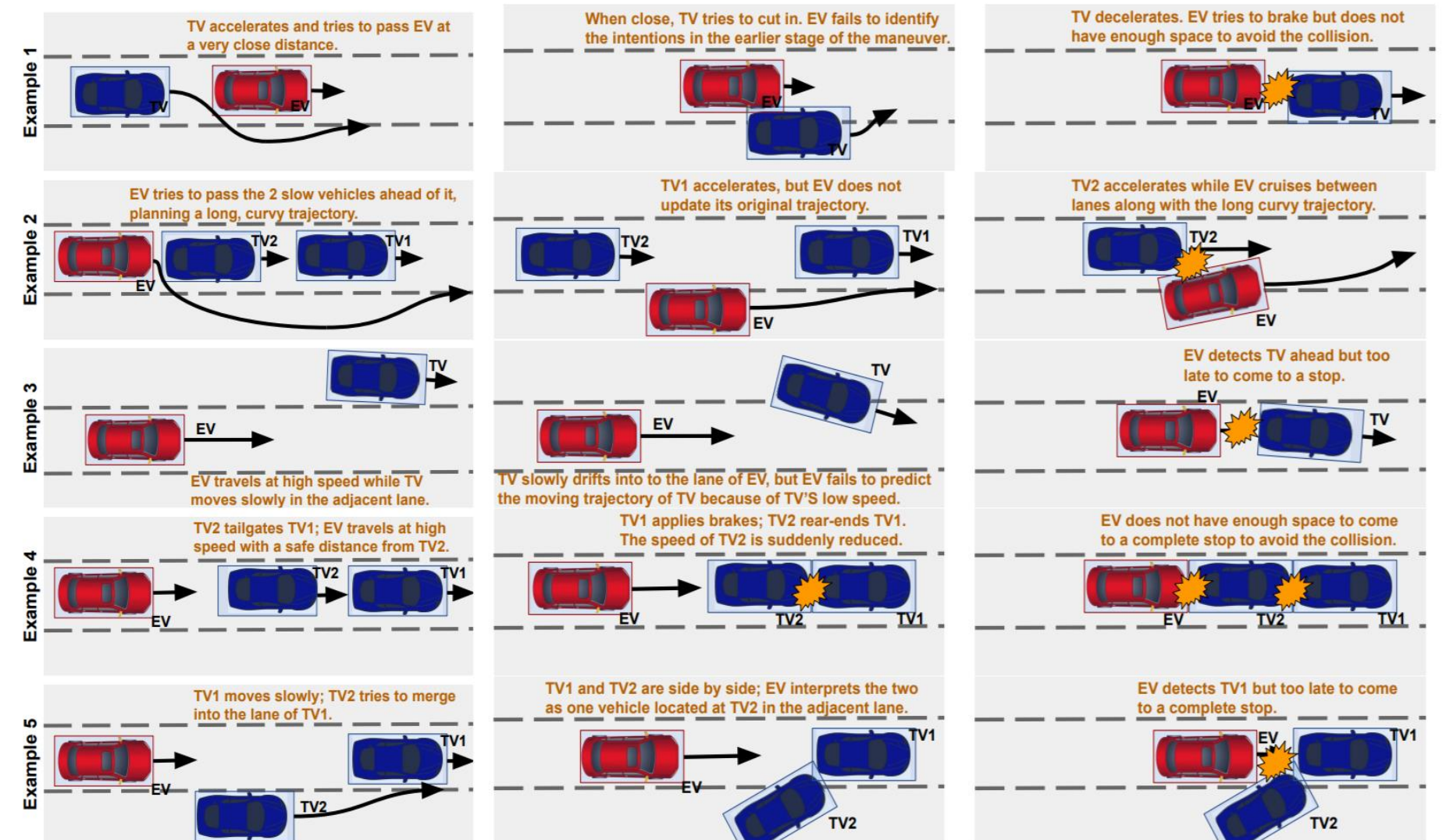


Figure 10: Examples of safety violation scenarios.

SCENARIO CHARACTERIZATION AND SELECTION

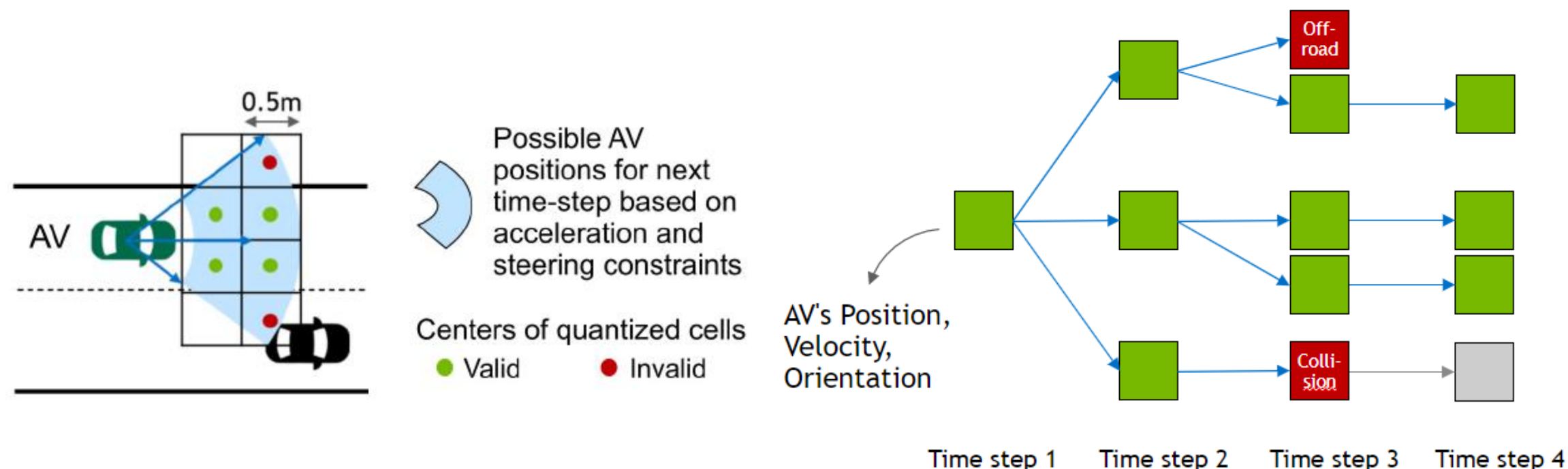
Selecting a diverse set for efficient safety testing

With advances in abstract scenario specification methods, scenario authoring and generation is becoming easier

Problem: Too many scenarios can be generated quickly. Distilling to a practical and diverse set is challenging.

Goal: Characterize (score) scenarios based on safety-related metrics to select interesting scenarios

Quantize space → Create a tree representation (one level is for one time-step) → Metrics based tree characteristics



Higher metric value → more difficult the scenario

SafePathInv: Inverse of number of safe routes ($1 / \text{\#safepaths}$)

UnsafePercent: Percentage of routes leading to collision

AvgEffort: Average effort for all safe routes

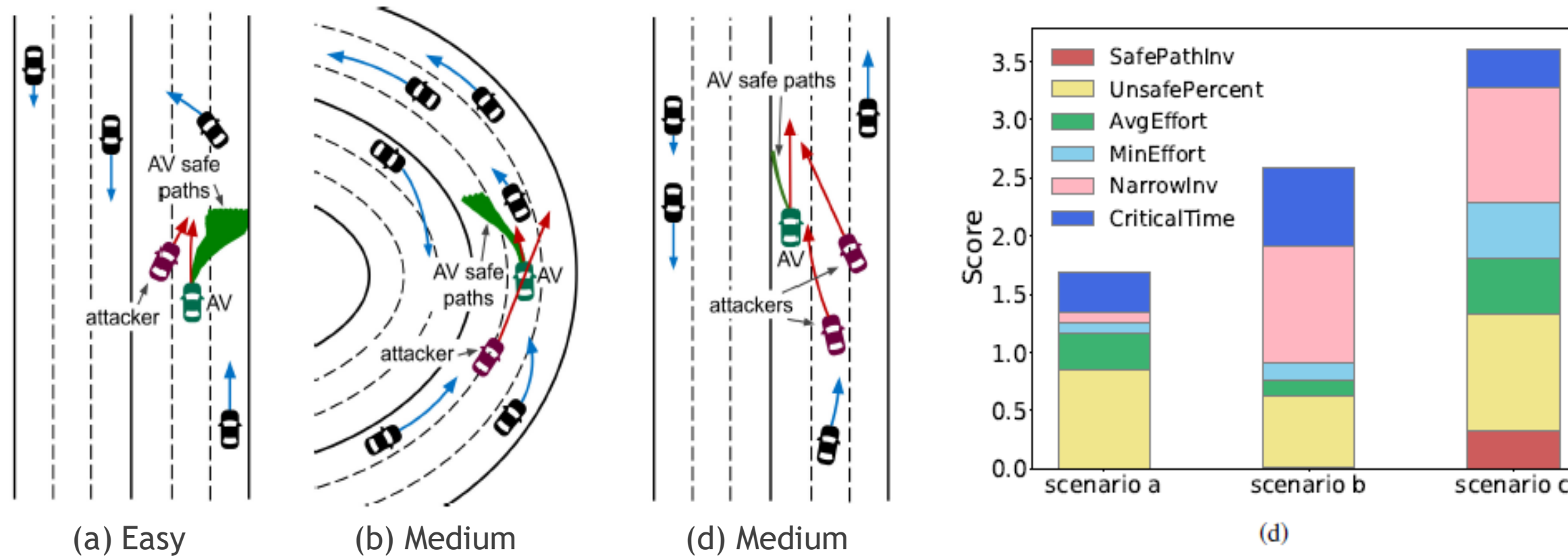
MinEffort: Minimum effort to navigate safely through a scenario

NarrowInv: Inverse of average narrowness for safe routes

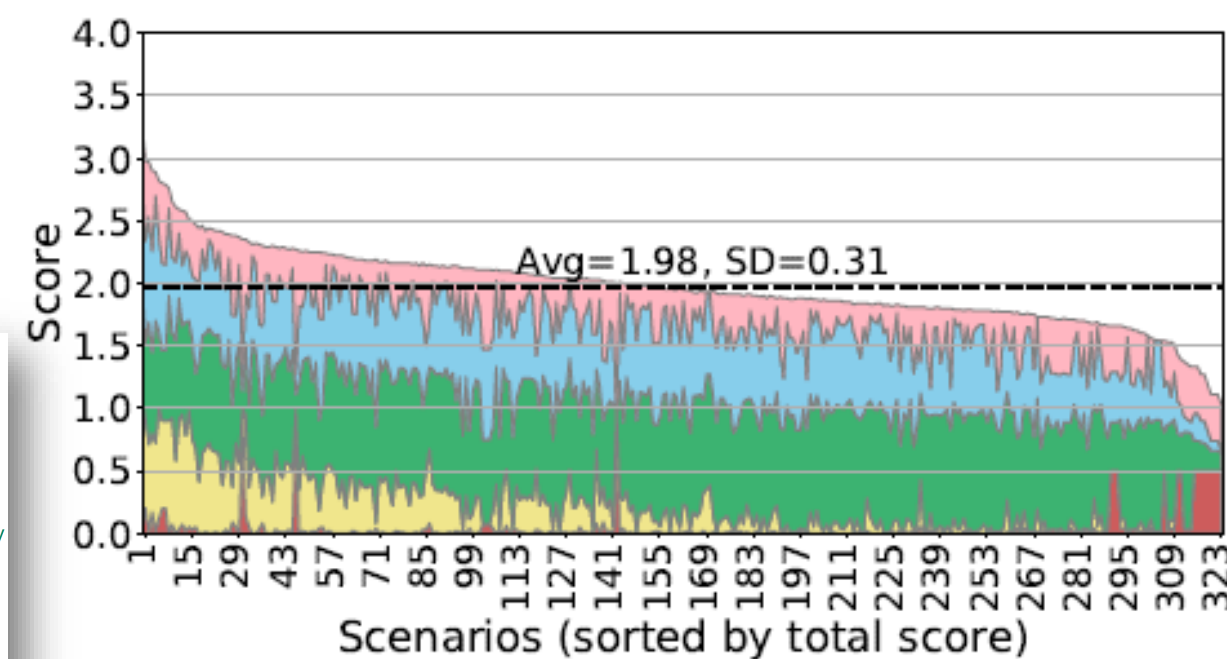
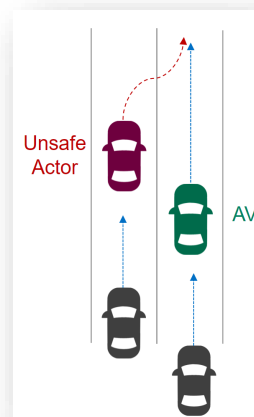
SCENARIO CHARACTERIZATION AND SELECTION

Results

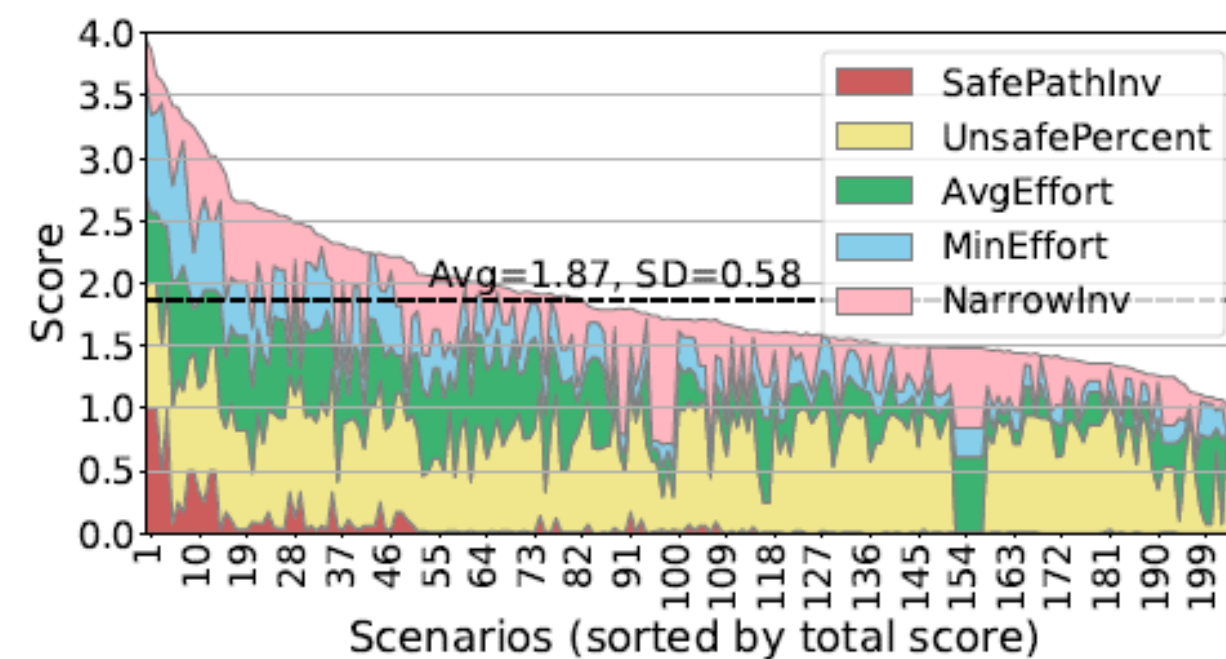
Three examples



Characterization for two data-sets



(a) Scenarios generated by our method



(b) Scenarios from NGSIM (US 101)



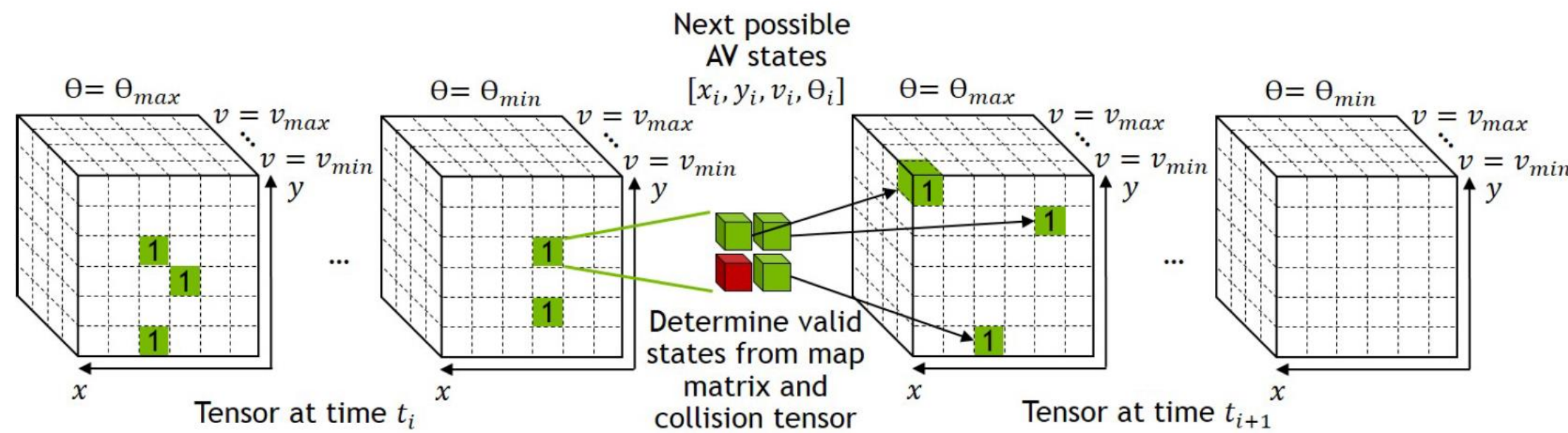
APPLICATIONS OF THE CHARACTERIZATION APPROACH

Generate new scenarios with desired characteristics

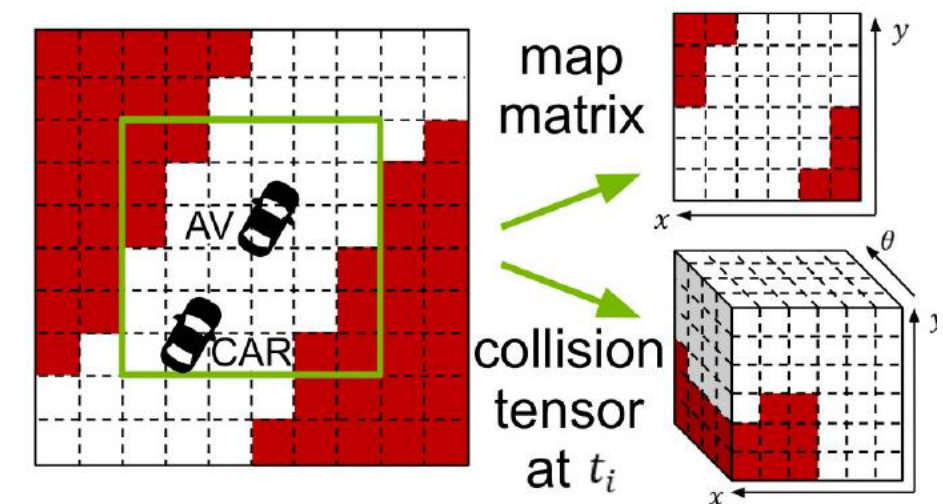
Suggest a safe path in case of a collision detection (by NVIDIA SFF, for example)

Employ in real-time to ensure that the AV's path is safe, if the characterization algorithm is fast

Developed a tensor-based method for an order-of-magnitude faster analysis



(a) AV state transition from time i to $i+1$, by computing next possible AV states for each valid state at time i



(b) Pre-calculation of map matrix and collision tensor

A complex network diagram with white and yellow nodes connected by thin lines on a dark background. The nodes are distributed across the frame, with a higher density in the upper right and lower left areas. The lines represent connections between the nodes, creating a web-like structure.

DESIGNING EFFICIENT AND SAFER AV SYSTEMS

PERCEPTION QUALITY

Scenario Dependence

Many AV algorithms are resource demanding, especially Perception

Perception requirements can be scenario dependent. For example:

AV traveling at slow speeds with no object near it can tolerate low perception quality

AV going through an intersection can benefit from elevated perception for cross-traffic

Future AV's are expected to employ many high-resolution cameras, radars, and possibly more than one LiDARs

Example: NIO Autonomous Driving system deploys many high-performance sensors



RESEARCH QUESTIONS

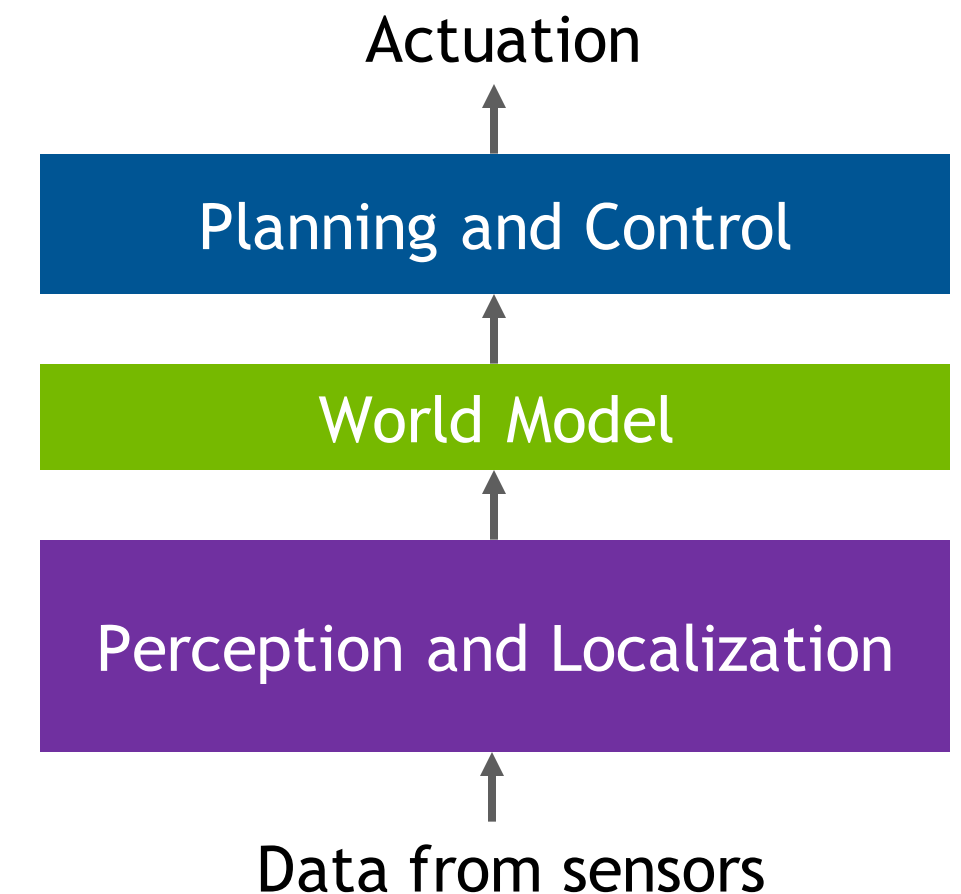
Designing for safety and efficiency

What perception quality is required for safe operation under different conditions?

How to design and optimize the perception stack?

Several choices/trade-offs: Camera resolution, camera FPS, CNN model accuracy (quantization, pruning), fusion with multiple sensors

What set of scenarios should be considered to derive requirements?



PRELIMINARY STUDY

System's tolerance to perception degradation in different driving conditions

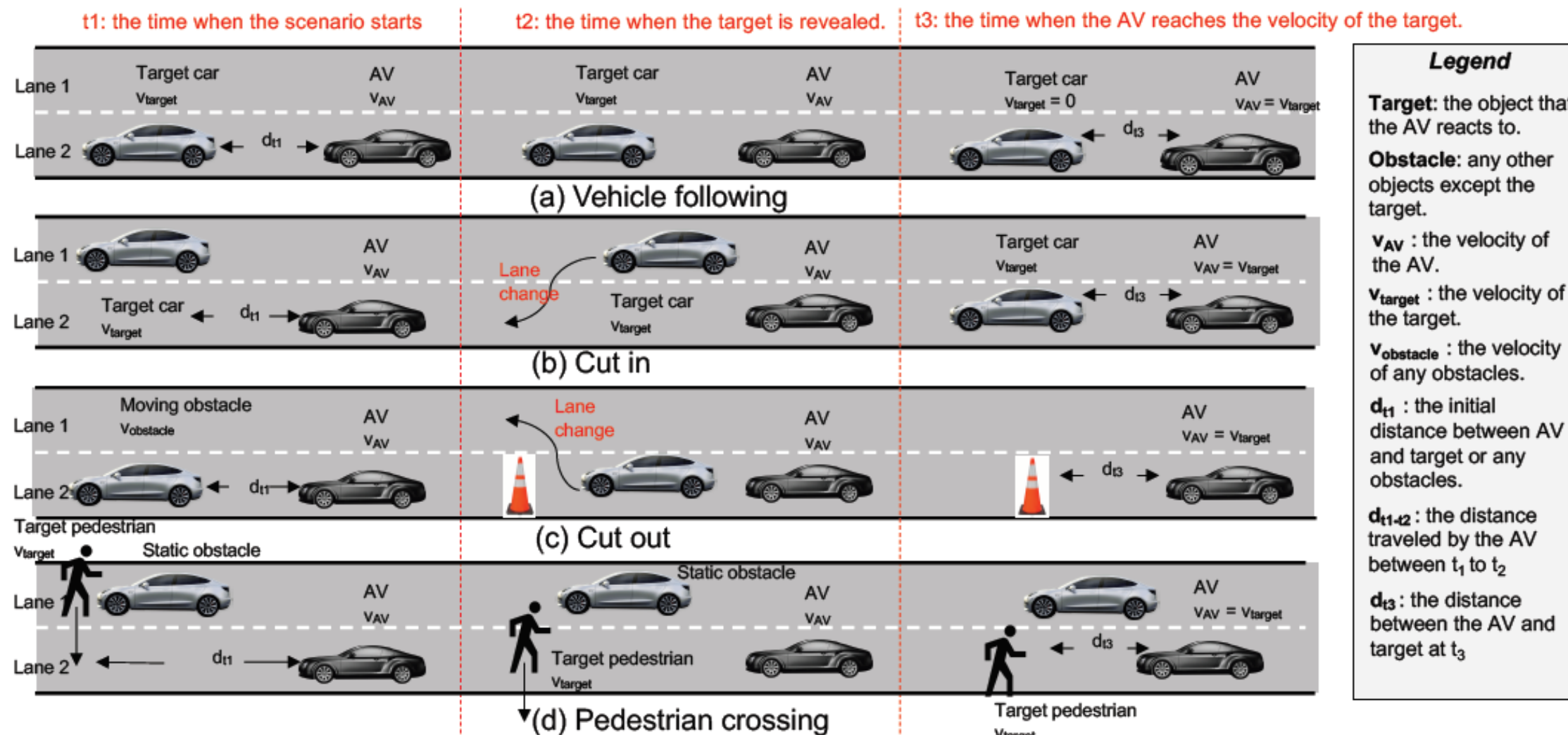
Configure system with different perception settings; run each version across the scenario suite

Quantify tolerance for each of the settings in different scenarios

Setup: Adaptive Cruise Control and Lane Keep; Single camera

Varied camera frames per second (FPS) and perception model precision

Scenarios:



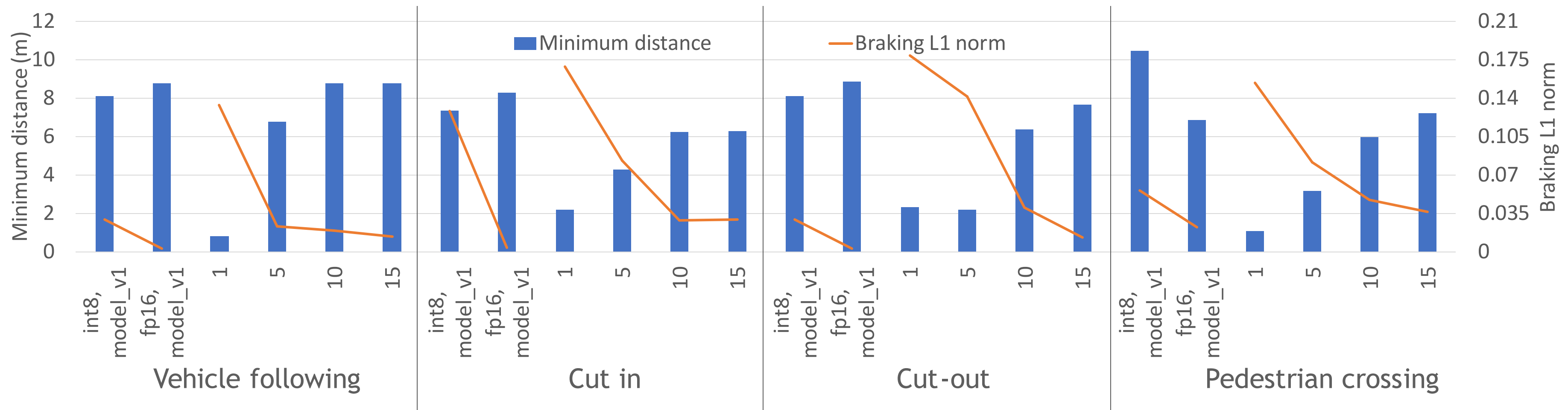
Average acceleration applied by the system:

- > 0.5 x g: Hard
- > 0.25 x g: Moderate
- < 0.25 x g: Easy (Ignore)

PRELIMINARY RESULTS

Planning algorithm adapted to tolerate delayed perception

For the tested hard/moderate scenarios, lowering FPS by 2-3x did not result in significant safety issue



Also studied injecting noise and delay directly to the perceived world model - observed significant tolerance

FUTURE RESEARCH

Designing efficient perception stack for a safer AV

Efficient methods to explore trade-offs offered by various perception parameter setting combinations

Quantify benefits of adding sensors diversity under different scenarios

Architecting the system to tolerate component-level failures (e.g., camera malfunction or blockage)

Dynamic perception boosting for the areas around the AV (or for objects) that impact safety most

Efficient on-line verification system

SUMMARY

Vehicles are becoming increasingly autonomous

Safety is a key requirement for autonomous vehicles

Discussed the following factors that affect safety

- Hardware errors: Evaluating the effects of hardware errors; Error mitigation strategies

- Scenarios: Driving scenario generation for testing; Characterization for selecting a diverse set

- Perception: Effects of degrading perception quality on safety; design exploration for efficient and safe system

Several research problems are yet to be addressed

