



# **Investigating the Impact of Simulation Fidelity on Automotive Decision-Making**

Master's thesis in Computer Science and Engineering

**ANTON GOLE  
HUNG LE**



MASTER'S THESIS 2025

# Investigating the Impact of Simulation Fidelity on Automotive Decision-Making

ANTON GOLE  
HUNG LE



UNIVERSITY OF  
GOTHENBURG



**CHALMERS**  
UNIVERSITY OF TECHNOLOGY

Department of Computer Science and Engineering  
CHALMERS UNIVERSITY OF TECHNOLOGY  
UNIVERSITY OF GOTHENBURG  
Gothenburg, Sweden 2025

Investigating the Impact of Simulation Fidelity on Automotive Decision-Making  
ANTON GOLE, HUNG LE

© ANTON GOLE, HUNG LE, 2025.

Supervisor: Felix Cherubini, Computer Science and Engineering  
Supervisor: Sergei Borzov, Volvo Cars  
Examiner: Ana Bove, Computer Science and Engineering

Master's Thesis 2025  
Department of Computer Science and Engineering  
Chalmers University of Technology and University of Gothenburg  
SE-412 96 Gothenburg  
Telephone +46 31 772 1000

Cover: Example image taken from inside our simulation in the game engine Unreal Engine 5.

Typeset in L<sup>A</sup>T<sub>E</sub>X  
Gothenburg, Sweden 2025

Investigating the Impact of Simulation Fidelity on Automotive Decision-Making

ANTON GOLE

HUNG LE

Department of Computer Science and Engineering

Chalmers University of Technology and University of Gothenburg

## Abstract

This thesis examines how *visual* rendering choices of a driving simulator changes how a camera-based Automated Emergency Braking (AEB) behaves. Scenario logic, perception, decision, and actuation are held fixed; only the visual rendering varies. One lower-detail baseline is compared with 39 higher-detail configurations that switch shadows on/off, change the sky, add environmental clutter (trees, grass, rocks), and vary the sun's height and direction across sixteen scenarios. The following metrics were used: AEB activation distance, Time-to-Collision at activation, and an activation indicator.

As a baseline validation, we first confirm that results are closely matched when advanced visual effects are turned off in two independent implementations of the simulator (typical differences about 2-3%). Within the higher-detail variants, effects are context-dependent rather than uniform. Low sun elevation ( $10^\circ$  above the horizon) consistently reduced safety margins: on average AEB triggers roughly 1 m closer and about 0.05 s later. Higher visual detail also exposed hidden failures: in 5 of 16 scenarios, at least one higher-detail run did not activate AEB while the baseline did (15/640 runs; 2.3% overall; per-scenario failure rates about 2-21%).

Analyses combined matched-pair comparisons, within-scenario models for continuous shifts, and logistic regression for failure analysis. Overall, visual setting *can* change AEB timing and occasionally prevent activation. Because each scenario has only one low-fidelity baseline run and failures are few, these findings are interpreted as descriptive signals: we recommend reporting visual settings and testing across varied lighting and scene complexity when using simulation for evaluation.

Keywords: simulation fidelity; visual rendering; Automated Emergency Braking (AEB); Advanced Driver-Assistance Systems (ADAS); closed-loop simulation; high-fidelity configurations.



## Acknowledgements

This master's thesis was carried out at Volvo Cars in Göteborg. We would like to express our sincere gratitude to our supervisors, Felix Cherubini (Department of Computer Science and Engineering) and Sergei Borzov (Volvo Cars), for their guidance, constructive feedback, and support throughout the project. We are also grateful to our examiner, Ana Bove (Department of Computer Science and Engineering), for her insightful comments and careful review.

We thank Volvo Cars for hosting the work and for providing the resources, data, and stimulating environment that made this thesis possible. The discussions with colleagues at Volvo Cars greatly helped refine our experimental design and interpretation of results related to simulation fidelity.

Anton Gole, Hung Le, Gothenburg, 2025-11-07



# Contents

<b>List of Acronyms</b>	<b>xiii</b>
<b>Nomenclature</b>	<b>xv</b>
<b>List of Figures</b>	<b>xvii</b>
<b>List of Tables</b>	<b>xix</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Background . . . . .	1
1.2 Testing & Simulation . . . . .	2
1.3 Problem Statement . . . . .	2
1.4 Objectives . . . . .	2
1.5 Scope and Limitations . . . . .	3
<b>2 Theory</b>	<b>5</b>
2.1 Simulation Fidelity . . . . .	5
2.2 ASAM OpenX Standards for Scenario-Based Simulation . . . . .	5
2.2.1 OpenDRIVE: Road-Network Geometry . . . . .	6
2.2.2 OpenSCENARIO XML: Dynamic Actors and Logic . . . . .	6
2.2.3 Open Simulation Interface (OSI) . . . . .	7
2.3 esmini: A Lightweight OpenSCENARIO Player . . . . .	7
2.4 Closed-Loop Testing Architecture . . . . .	8
2.5 Performance-Evaluation Metrics . . . . .	9
2.5.1 AEB Activation Time $t_{AEB}$ . . . . .	9
2.5.2 Distance to Lead Object at AEB Activation $d_{AEB}$ (longitudinal) .	10
2.5.3 Time-To-Collision at AEB Activation $TTC_{AEB}$ (longitudinal) .	10
2.5.4 Alignment with Safety Standards . . . . .	10
2.6 Core Perception Module: Zenseact DLOD . . . . .	11
2.6.1 Why FCOS3D? . . . . .	11
2.6.2 Key Design Choices . . . . .	11
2.6.3 Performance . . . . .	12
2.7 Game-Engine Rendering Pipelines . . . . .	12
2.7.1 Unity URP . . . . .	12
2.7.2 Unreal Engine 5 + Lumen/Path Tracing . . . . .	12
2.7.2.1 Why UE5 for the Fidelity Sweeps . . . . .	12

2.7.2.2	Nanite Virtualized Geometry . . . . .	13
2.7.2.3	Lumen Dynamic Global Illumination . . . . .	13
<b>3</b>	<b>Simulation System Design</b>	<b>15</b>
3.1	System Architecture Overview . . . . .	15
3.2	Synchronization & Communication Protocol . . . . .	16
3.2.1	Inter-Process Communication . . . . .	16
3.2.2	Synchronous Image Loop . . . . .	17
3.3	Implementation Details . . . . .	17
3.3.1	<code>esmini</code> Endpoint . . . . .	17
3.3.2	UE5 Endpoint . . . . .	18
3.3.3	Timing and Determinism . . . . .	18
3.3.4	Road-mesh Builder . . . . .	18
3.3.5	Procedural Generation . . . . .	19
3.3.6	Camera Setup . . . . .	19
<b>4</b>	<b>Experimental Methodology</b>	<b>21</b>
4.1	Game-Engine Alignment . . . . .	21
4.2	Experimental Design . . . . .	22
4.2.1	Rendering Fidelity Design . . . . .	22
4.2.2	NCAP-Derived Safety Scenarios . . . . .	24
4.3	Data Gathering and Processing . . . . .	25
4.3.1	Signals and Derived Metrics . . . . .	25
4.3.2	Scenario and Fidelity Labels . . . . .	26
4.4	Statistical Analysis Framework . . . . .	26
4.4.1	Decomposing the Research Questions . . . . .	26
4.4.2	RQ1: Baseline and Fidelity Analysis . . . . .	27
4.4.2.1	Baseline Agreement Analysis . . . . .	27
4.4.2.1.1	Rationale for approach. . . . .	27
4.4.2.2	Fidelity Effects on Continuous Outcomes . . . . .	28
4.4.2.2.1	Design and predictors. . . . .	28
4.4.2.2.2	Estimation and uncertainty. . . . .	29
4.4.2.2.3	Rationale. . . . .	29
4.4.2.2.4	Interpretation. . . . .	29
4.4.2.3	RQ2: Failure Exposure and Prediction . . . . .	29
4.4.3.1	Failure Exposure . . . . .	29
4.4.3.1.1	Caution about randomness. . . . .	30
4.4.3.2	Failure Predictors . . . . .	30
4.4.3.2.1	Rationale. . . . .	30
4.4.3.2.2	Predictors and coding. . . . .	30
4.4.3.2.3	Estimation and uncertainty. . . . .	30
4.4.3.2.4	Reporting and interpretation. . . . .	31
<b>5</b>	<b>Results</b>	<b>33</b>
5.1	RQ1: Baseline and Fidelity Analysis . . . . .	33
5.1.1	Baseline Agreement . . . . .	33
5.1.2	Fidelity Effects on Continuous Outcomes . . . . .	35

5.2 RQ2: Failure Exposure and Prediction . . . . .	36
5.2.1 Failure Exposure . . . . .	37
5.2.2 Failure Predictors . . . . .	38
<b>6 Discussion</b>	<b>39</b>
6.1 What the baseline validation establishes . . . . .	39
6.2 How fidelity changes AEB activation margins . . . . .	39
6.3 Higher fidelity exposes hidden non-activations . . . . .	39
6.4 Interpreting the failure model . . . . .	40
6.5 Practical implications for simulation-based ADAS testing . . . . .	40
6.6 Threats to validity and limitations . . . . .	40
6.7 Future work . . . . .	41
<b>Bibliography</b>	<b>43</b>
<b>A Comprehensive Results and AEB Metrics</b>	<b>I</b>
<b>B Source Code Listings</b>	<b>XIX</b>

## Contents

---

# List of Acronyms

<b>AAE</b>	Average Attribute Error
<b>ADAS</b>	Advanced Driver-Assistance Systems
<b>ADS</b>	Automated Driving Systems
<b>AEB</b>	Automated Emergency Braking
<b>API</b>	Application Programming Interface
<b>ASAM</b>	Association for Standardisation of Automation and Measuring Systems
<b>ATE</b>	Average Translation Error
<b>AOE</b>	Average Orientation Error
<b>ASE</b>	Average Scale Error
<b>AVE</b>	Average Velocity Error
<b>CCC</b>	Concordance Correlation Coefficient
<b>CI</b>	Confidence Interval
<b>CI/CD</b>	Continuous Integration / Continuous Delivery
<b>C2C</b>	Car-to-Car
<b>EPV</b>	Events per Variable
<b>DLOD</b>	Deep Learning Object Detection
<b>FCOS3D</b>	Fully Convolutional One-Stage 3D Detector
<b>FIFO</b>	First In, First Out
<b>GI</b>	Global Illumination
<b>GPU</b>	Graphics Processing Unit
<b>HDF5</b>	Hierarchical Data Format version 5
<b>HDR</b>	High Dynamic Range
<b>ICC</b>	Intraclass Correlation Coefficient
<b>ID</b>	Identifier

<b>ISO</b>	International Organization for Standardization
<b>JSON</b>	JavaScript Object Notation
<b>LBFGS</b>	Limited-memory Broyden-Fletcher-Goldfarb-Shanno
<b>LoA</b>	Limits of Agreement
<b>LiDAR</b>	Light Detection and Ranging
<b>LOD</b>	Level of Detail
<b>mAP</b>	mean Average Precision
<b>NDS</b>	nuScenes Detection Score
<b>NCAP</b>	New Car Assessment Programme
<b>OR</b>	Odds Ratio
<b>OSI</b>	Open Simulation Interface
<b>PCG</b>	Procedural Content Generation
<b>RMSE</b>	Root Mean Square Error
<b>SDF</b>	Signed Distance Field
<b>SOTIF</b>	Safety of the Intended Functionality
<b>TCP</b>	Transmission Control Protocol
<b>TTC</b>	Time-To-Collision
<b>UDP</b>	User Datagram Protocol
<b>UE5</b>	Unreal Engine 5
<b>URP</b>	Universal Render Pipeline
<b>VRU</b>	Vulnerable Road User
<b>XML</b>	Extensible Markup Language
<b>ZMQ</b>	ZeroMQ

# Nomenclature

## Variables

$\Delta t$	Fixed simulation step size (25 ms; logged as $2.5 \times 10^7$ ns)
$\Delta y_{is}$	Within-scenario change from baseline in outcome $y$ for run $i$ of scenario $s$
$\Delta D$	Change in decision output between fidelity conditions
$\dot{v}_{\text{ego}}$	Time-derivative of ego-vehicle speed ( $\text{m s}^{-2}$ )
$\dot{y}$	Lateral velocity ( $\text{m s}^{-1}$ )
$\hat{\mathbf{s}}(t)$	Longitudinal unit vector of the ego vehicle at time $t$
$\hat{p}_s$	Estimated failure rate for scenario $s$
$\mathbb{I}_{\text{no-AEB}}$	Indicator: 1 if no AEB activation observed, else 0
$\mathbf{p}_{\text{ego}}$	Position vector of the ego vehicle (m)
$\mathbf{p}_{\text{lead}}$	Position vector of the lead (front) vehicle (m)
$\mathbf{v}_{\text{ego}}$	Velocity vector of the ego vehicle ( $\text{m s}^{-1}$ )
$\mathbf{v}_{\text{lead}}$	Velocity vector of the lead vehicle ( $\text{m s}^{-1}$ )
$\mathcal{A}$	Actuator (vehicle-control) model in closed-loop diagram
$\mathcal{D}$	Decision-making module in closed-loop diagram
$\mathcal{P}$	Perception module in closed-loop diagram
$\mathcal{S}_{\text{logic}}$	Scenario-execution layer ( <code>esmini</code> ) in closed-loop diagram
TTC	Time-to-Collision computed from relative range and speed (s)
TTC <sub>AEB</sub>	Time-to-Collision at AEB activation (s)
$C_s$	Failure count in scenario $s$
$d_{\text{AEB}}$	Longitudinal distance to lead at AEB activation (m)
$R$	Rendering module that varies between fidelity conditions

$s$	Reference-line arc-length coordinate in OpenDRIVE $s$ - $t$ frame (m)
$t$	Lateral offset from the reference line in OpenDRIVE $s$ - $t$ frame (m)
$t_{AEB}$	AEB activation time (ms; logged in ns)
$v$	Vehicle speed ( $\text{m s}^{-1}$ )
$x$	Cartesian longitudinal position (m)
$y$	Cartesian lateral position (m)
$y_s^{(\text{base})}$	Scenario- $s$ baseline value (UE5 fidelity-off) for outcome $y$
$z$	Vertical position / altitude (m)
$z_{0.975}$	Standard-normal 97.5% quantile used in Wilson interval

### Special terms

Actor	Any entity in OpenSCENARIO able to perform actions (vehicles, pedestrians, etc.)
Closed-loop	Simulation where perception outputs steer the virtual vehicle each step
Dynamic actor	An actor with a scripted trajectory and behaviour
Ego vehicle	The vehicle of interest during simulation tests
esmini	Light-weight OpenSCENARIO/OpenDRIVE player used as scenario engine
GameInstance	Unreal Engine's top-level manager for a running application instance.
Lumen	Unreal Engine's dynamic global-illumination system
Nanite	Unreal Engine's virtualised geometry system
Path Tracer	Unreal Engine's offline unbiased renderer
PCG pipeline	Procedural-Content-Generation workflow for virtual environments
Perception stack	Sequence of detection, tracking and classification modules
Scenario	Complete description of roads, traffic and triggers used as an experiment
Time-to-Collision	Time needed to close the gap under current relative speed

# List of Figures

2.1	Reference-line-based $s$ - $t$ coordinate system with origin at the beginning of the road. . . . .	6
2.2	Screenshot of a 3D scene rendered by <code>esmini</code> . The minimal rendering style highlights lane-level and object's ground-truth structure, but lacks dynamic lighting and high-fidelity textures. This rendering is typically used for debugging or visualization, while image-based sensor testing requires an external engine such as Unity or Unreal. . . . .	8
2.3	Demonstration of Lumen global illumination in UE5. . . . .	13
3.1	Per-step synchronization between <code>esmini</code> , UE5, and the vision stack. <code>esmini</code> sends the step state (Tick), UE5 renders and conditionally ships the image to the vision stack, then acknowledges the same step (Tock). . . . .	17
4.1	Effect of enabling each of the three fidelity factors. . . . .	23
4.2	Effect of sun elevation angle on lighting and shadows, with sun fixed in the west. From left to right: 90° (zenith), 45°, and 10° above the horizon. Lower sun angles produce longer, softer shadows and warmer lighting effects. . . . .	23
5.1	Direct comparison (scatter): Unity values ( $x$ ) vs. UE5 (fidelity-off) values ( $y$ ) with the identity line $y = x$ . Points close to the identity line indicate strong baseline comparability. . . . .	35

List of Figures

# List of Tables

3.1	FIFOs used for synchronization. . . . .	16
4.1	Visual-fidelity factors explored in UE5 . . . . .	22
4.2	Sun-light directions tested when <i>shadows</i> = <i>On</i> . . . . .	23
4.3	NCAP scenarios and initial conditions . . . . .	25
5.1	Baseline alignment (Unity vs. UE5, fidelity-off): matched-pairs summaries. $\Delta = \text{UE5}_{\text{off}} - \text{Unity}$ . . . . .	33
5.2	Per-scenario <i>symmetric</i> absolute percentage differences for baseline alignment. Percentage difference is $100 \times \frac{2 m_s^{(\text{UE5\_off})} - m_s^{(\text{Unity})} }{ m_s^{(\text{UE5\_off})}  +  m_s^{(\text{Unity})} }$ . . . . .	34
5.3	Ridge linear model for $\Delta d_{\text{AEB}}$ (metres); scenario-cluster bootstrap 95% CI. . . . .	36
5.4	Ridge linear model for $\Delta \text{TTC}_{\text{AEB}}$ (seconds); scenario-cluster bootstrap 95% CI. . . . .	36
5.5	Scenarios where higher fidelity exposed AEB failures (baselines activated successfully). . . . .	37
5.6	Ridge-penalized logistic regression predicting AEB non-activation. Odds ratios (OR) with 95% scenario-cluster bootstrap CIs; baseline is <b>Shadow=off, PCG=off, Sky=white, Elevation=90°</b> . . . . .	38
A.1	AEB metrics for scenario: Car to Car front turn across path (ccftap_speed_20_45) . . . . .	II
A.2	AEB metrics for scenario: Car to Car front turn across path (ccf-tap_speed_20_60) . . . . .	III
A.3	AEB metrics for scenario: Car to Car Rear braking (ccrb_decel_6) .	IV
A.4	AEB metrics for scenario: Car to Car Rear braking (ccrb_decel_2) .	V
A.5	AEB metrics for scenario: Car to Car Rear moving (ccrm_speed_50)	VI
A.6	AEB metrics for scenario: Car to Car Rear moving (ccrm_speed_80)	VII
A.7	AEB metrics for scenario: Car to Car Rear stationary (ccrs_speed_50)	VIII
A.8	AEB metrics for scenario: Car to Car Rear stationary (ccrs_speed_80)	IX
A.9	AEB metrics for scenario: Car to Pedestrian Farside Adult (cpfa_speed_50)	X
A.10	AEB metrics for scenario: Car to Pedestrian Farside Adult (cpfa_speed_60)	XI
A.11	AEB metrics for scenario: Car to Pedestrian Longitudinal Adult (cpla_speed_50) . . . . .	XII
A.12	AEB metrics for scenario: Car to Pedestrian Longitudinal Adult (cpla_speed_80) . . . . .	XIII

## List of Tables

---

A.13 AEB metrics for scenario: Car to Pedestrian Nearside Adult (cpna_speed_50)	XIV
A.14 AEB metrics for scenario: Car to Pedestrian Nearside Adult (cpna_speed_60)	XV
A.15 AEB metrics for scenario: Car to Pedestrian Nearside Child Obstructed (cpnco_speed_50)	XVI
A.16 AEB metrics for scenario: Car to Pedestrian Nearside Child Obstructed (cpnco_speed_60)	XVII

# 1

## Introduction

Autonomous driving technologies, in their current state, are no longer experimental. Partial automation functions such as lane centering and Automated Emergency Braking (AEB) are now delivered in millions of production cars. As the scope of these so-called Advanced Driver-Assistance Systems (ADAS) expands, every software update must be evaluated for safety-relevant performance across a wide variety of traffic, weather, and lighting conditions. Exhaustive road testing is economically unfeasible, so industry practice has shifted toward large-scale, *closed-loop simulation* (the perception-decision outputs influence the simulated vehicle and world, which produce the next sensor inputs; see §2.4).

### 1.1 Background

Road transport remains an important but dangerous activity of modern society. According to the World Health Organization, in 2021 more than 1.19 million people were killed in traffic accidents. In 2019, road injuries were the leading cause of death for children and young people aged 5 to 29, and the 12th leading cause of death across all age groups, according to the same source [1]. Advanced Driver-Assistance Systems (ADAS) and their higher-level successors, Automated Driving Systems (ADS), have been introduced in many vehicles for this reason. These systems combine camera, radar, and Light Detection and Ranging (LiDAR) sensing with onboard intelligence to keep the vehicle in its lane, monitor blind spots, and even perform emergency manoeuvres [2].

Since the first production anti-lock braking system was introduced in 1978, vehicle automation has steadily increased in sophistication [3]. Modern vehicles offer Level 2 (partial automation) and even Level 3 (conditional automation) capabilities [2]. In parallel, the European Commission now requires the inclusion of features such as Collision Avoidance, Intelligent Speed Adaptation and Automated Emergency Braking in new vehicles [4]. Ensuring the safety of these increasingly complex, learning-based systems can be a challenge. As a result, the verification process has become a major bottleneck.

## 1.2 Testing & Simulation

Automotive testing traditionally occurs in both real-world conditions and controlled simulations. While real-world testing provides direct validation, it is expensive, time-consuming, and limited in scope. Simulation-based testing in virtual environments offers a scalable alternative, allowing manufacturers to evaluate decision-making systems under diverse scenarios and environmental conditions. Within such virtual worlds, *simulation fidelity*—the level of realism in geometry, lighting, shadows, materials, and environmental detail—becomes a critical variable. Depending on the software stack, graphic settings, and scene-generation techniques, fidelity can range from minimalist shading to high-fidelity real-time global illumination. Whether pushing to the upper end of the fidelity spectrum can affect an ADAS controller’s continuous behaviour remains an open question. Therefore in this thesis we adapt multiple higher-fidelity configurations that enable shadows, richer illumination, and procedural scene content; all other closed-loop components are held constant.

## 1.3 Problem Statement

Research to date has focused on frame-wise perception accuracy, yet little is known about how visual fidelity influences the controller’s decision-making. Does a photo-realistic scene have no impact, or does it influence the automated system’s decision-making, in particular the timing and conditions of Automated Emergency Braking, and to what extent? This thesis addresses the question by using both high- and low-fidelity scenes—under varying lighting, shadows, and environmental richness—to measure their impact on an industry-grade decision-maker. We further examine whether faults emerge only under higher-fidelity conditions—indicating that advanced rendering may expose failure modes otherwise hidden—and *explore* which fidelity factors are most plausibly associated with these failures.

Below are the two research questions (RQ) this thesis aims to answer:

- **RQ1:** Does increased simulation fidelity lead to changes in ADAS behaviour?
- **RQ2:** Can higher fidelity expose failure modes that remain hidden in low-fidelity simulation, and if so, which fidelity factors are most strongly associated with these failures?

## 1.4 Objectives

The work is organized around five objectives:

- **Adopt a More Capable Rendering Engine:** We migrate from the existing Unity baseline [5] to Unreal Engine 5 (UE5) for the *fidelity sweeps*. Two practical needs motivated this choice. First, UE5’s Nanite geometry (see section §2.7.2.2) plus the built-in Procedural Content Generation (PCG) framework let us populate dense roadside clutter (trees, grass, rocks) at scale. Second, UE5’s Lumen provides *dynamic, physically based* global illumination and soft

shadows out of the box, which made our sun-angle manipulations straightforward. To keep scenario logic, geometry, and materials consistent across engines, we export identical scenes to both renderers and then *verify* that Unity and UE5 with all fidelity features disabled produce closely aligned outputs (§4.1, §4.4). Because even minor baseline differences between engines could confound results, all subsequent experiments and analyses are performed entirely in UE5, using the configuration with all fidelity factors disabled as the reference to ensure that any behavioural changes can be attributed solely to rendering-fidelity modifications rather than engine differences.

- **Define and Implement Metrics:** Alongside pass/fail, extract continuous AEB metrics (Time-to-Collision (TTC), distance at activation) to capture subtle behavioural shifts, and compare them across fidelity levels using the statistical approach outlined in §4.4.
- **Automatic Generation of Scenes from Road Descriptions:** Create a tool for automatically generating scenes from road descriptions by incorporating high modularity, and leveraging UE5 capabilities for procedural generation.
- **Run Simulations in Varying Fidelity Levels:** For each simulated traffic scenario, execute 40 UE5 runs: one fidelity-off baseline and 39 fidelity-on variants created by toggling visual-fidelity factors: **Shadow** (off/on), PCG roadside clutter (off/on), **Sky** model (white vs. cloudy), and sun illumination (elevations 90°, 45°, 10°; directions north, east, south, west). Scenario logic, geometry, and materials are identical across variants; the UE5 fidelity-off run serves as the per-scenario reference.
- **Analyze differences:** Assess cross-simulator baseline alignment using matched pairs, visualize agreement with identity-line scatter plots, model within-scenario shifts in activation margins, *screen* per-scenario failure rates to see how often fidelity-on variants produce non-activation, and relate fidelity factors to non-activations.

## 1.5 Scope and Limitations

This thesis focuses on *image-based* (camera) tests; no radar or LiDAR sensing is evaluated. Thus, certain aspects of real-world simulation are excluded:

- **No Audio-based Perception:** No sound inputs (sirens, honking, road noise) are included. The thesis centers on visual-based sensor evaluations.
- **Rendering Constraints in UE5:** While increasing scene fidelity, certain camera imperfections (auto-exposure, high dynamic range (HDR)) or complex lighting effects such as lens flares are not simulated.

Because we vary only camera-based visual fidelity with a single perception stack and forward-camera setup, our results support claims about *sensitivity* (how behaviour can change with visual conditions), not general guarantees across stacks, sensors, or vehicle dynamics. Conclusions should therefore be read as evidence that fidelity can

## 1. Introduction

both reveal and mitigate issues in perception-driven AEB, rather than as absolute performance judgments.

# 2

## Theory

This chapter explains the pieces of our closed-loop simulation and how they connect. We explain how the road network, together with actors and their behaviour inside each scenario, are defined. We also go into detail about how scenario execution works by giving an overview of the full simulation architecture; from initial rendering and image capturing to feeding vehicle control signals back to the simulation at every time step. For the implementation details, see Chapter 3, especially the architecture overview (§3.1).

### 2.1 Simulation Fidelity

Liu, Macchiarella and Vincenzi describe simulation fidelity as “the degree to which a simulation reproduces the state and behaviour of a real-world system in a measurable or perceivable manner” [6]. They distinguish several overlapping dimensions—physical, visual-audio, equipment, motion and psychological-cognitive.

For the purposes of this thesis, the term *simulation fidelity* refers solely to the *visual-audio* dimension: geometric detail, materials, lighting, shadows, and atmospheric effects that feed camera pipelines. All other dimensions (vehicle dynamics, equipment interfaces, motion cueing and audio ambience) are omitted so that any behavioural differences can be attributed to changes in visual realism alone.

### 2.2 ASAM OpenX Standards for Scenario-Based Simulation

Robust closed-loop testing of automated-driving functions depends on *interoperable* scene, map, and interface standards. The automotive industry has converged on the **ASAM** (Association for Standardisation of Automation and Measuring Systems) OpenX family [7], [8], [9], which allows different scenario players, traffic generators, sensor models, and control stacks to share a common digital ground-truth (the state of the virtual world). The three standards most relevant to this thesis are summarised below.

### 2.2.1 OpenDRIVE: Road-Network Geometry

ASAM OpenDRIVE describes the static road network—lane topology, elevation, superelevation, surface markings, signals, and other roadside objects—in a single `.xodr` (or compressed `.xodrz`) file [8]. Every element is parameterised along a one-dimensional reference line  $s$  with a lateral offset  $t$ . This curvilinear  $s$ - $t$  frame allows accurate reconstruction in any render or physics engine, and forms the basis of the road-aligned coordinate system used in most scenario definitions.

As shown in Figure 2.1, the  $s$ - $t$  system consists of two coordinate axes: the  $s$ -axis runs along the road’s reference line, while the  $t$ -axis points leftward orthogonal to it. This allows vehicle positions and lane elements to be defined in a road-relative frame, simplifying map-based localization and planning.

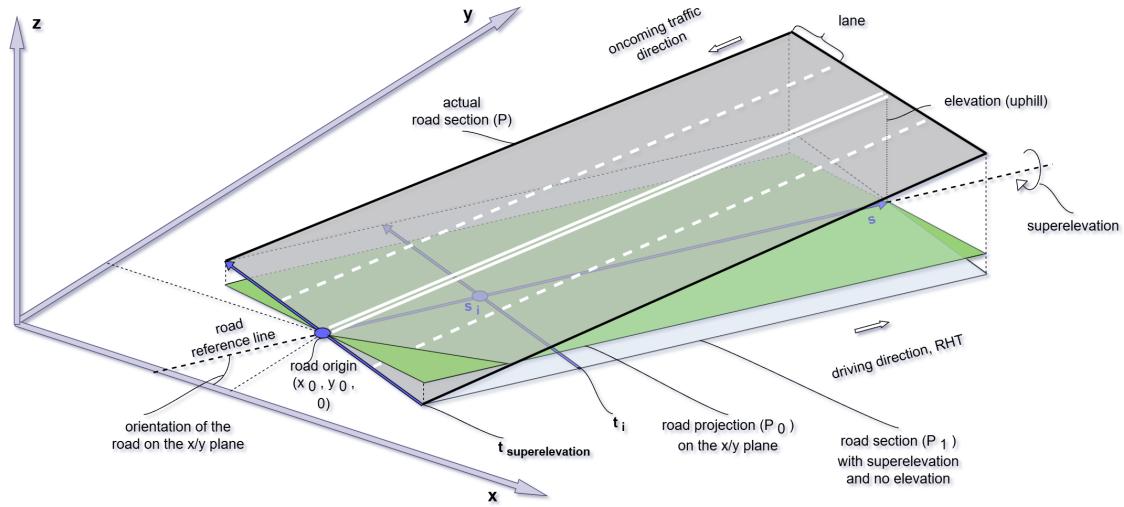


Figure 2.1: Reference-line-based  $s$ - $t$  coordinate system with origin at the beginning of the road.

### 2.2.2 OpenSCENARIO XML: Dynamic Actors and Logic

Where OpenDRIVE is static, ASAM OpenSCENARIO eXtensible Markup Language (XML) (`.xosc`) captures dynamic content—ego vehicle (the simulated car whose perception-planning stack is under test) and background entities, manoeuvres, trigger conditions, environmental parameters, and success criteria [7]. The language separates *what* happens (Actions inside Events) from *when and where* it happens (Start- and StopTriggers).

The behavioural model is hierarchical:

Story → Act → ManoeuvreGroup → Manoeuvre → Event → Action

A **Story** is a logical thread toward a goal. Each **Act** defines when behaviour is active (start/stop triggers). **ManeuverGroups** bind behaviour to actors. A **Maneuver** bundles one actor’s steps. An **Event** fires on a trigger and executes an **Action** (e.g., set speed, apply brake, lane change).

This explicit layering enables both sequential and parallel execution. For instance, a merging *ManoeuvreGroup* may contain two vehicles whose lane-change Actions are synchronised by a Condition that waits for a safe gap. The schema maps cleanly onto deterministic simulation back-ends such as **esmini** (§2.3).

### 2.2.3 Open Simulation Interface (OSI)

The Open Simulation Interface standardises run-time data exchange between simulation components via protocol-buffer messages [9]. OSI defines top-level messages for

- **GroundTruth**: ground-truth object list in a global frame,
- **SensorView** and **SensorData**: geometry clipped to a sensor field of view,
- **FeatureData**: mid-level perception features such as lane boundaries and object cues,
- **TrafficCommand/TrafficUpdate**: bidirectional control of traffic participants,
- **EgoControl**: actuation commands along the longitudinal (forward/reverse) and lateral (side-to-side) axes.

The message set allows both *modular* (component-in-the-loop) and *closed-loop* (software- or hardware-in-the-loop) evaluation by streaming sensor-level views into the stack and receiving actuation feedback.

## 2.3 esmini: A Lightweight OpenSCENARIO Player

**esmini** [10] is a minimalist yet capable scenario engine for OpenSCENARIO XML and OpenDRIVE. It can be used as a desktop viewer, a command-line batch runner, or a shared library embedded in custom pipelines.

At runtime, **esmini**

- parses OpenDRIVE road networks into an internal map representation,
- interprets OpenSCENARIO storyboards with deterministic fixed-step scheduling,
- exposes C/C++, Python, and C# application programming interfaces (APIs) for external controllers and observers,
- publishes OSI message streams (GroundTruth, SensorView, FeatureData) over User Datagram Protocol (UDP) or shared memory.

Entities such as the ego vehicle may therefore be driven by either script-defined Actions or an external closed-loop stack. Rendering is intentionally decoupled: **esmini** supplies ground-truth to Unity, Unreal, or custom renderers, keeping image generation outside the deterministic core. This architecture makes **esmini** a reproducible testbed for ADAS and ADS evaluation across a wide range of visual conditions.

`esmini`'s minimalist renderer (Figure 2.2) does not include detailed lighting or texture effects, making it insufficient for realistic image-based sensor simulation. In practice, it is often embedded in larger simulation frameworks where high-fidelity visuals are rendered by an external engine. This separation allows `esmini` to maintain deterministic behaviour and reproducibility while enabling advanced perception testing via more capable visual pipelines.

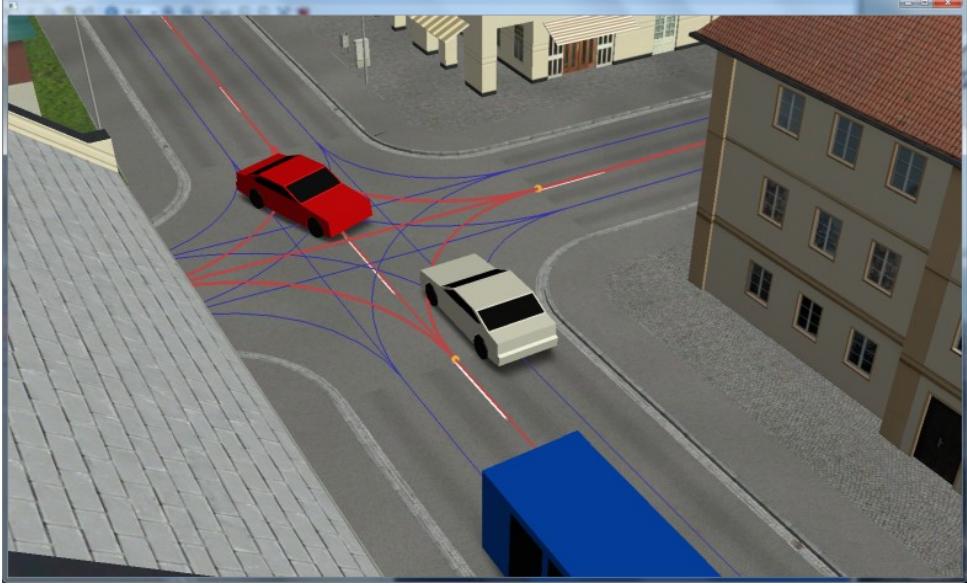


Figure 2.2: Screenshot of a 3D scene rendered by `esmini`. The minimal rendering style highlights lane-level and object's ground-truth structure, but lacks dynamic lighting and high-fidelity textures. This rendering is typically used for debugging or visualization, while image-based sensor testing requires an external engine such as Unity or Unreal.

## 2.4 Closed-Loop Testing Architecture

From a theory perspective, the sensor-simulation loop can be modelled as the composition

$$\underbrace{\mathcal{S}_{\text{logic}}}_{\text{esmini}} \rightarrow \underbrace{\mathcal{R}}_{\text{render pipeline}} \rightarrow \underbrace{\mathcal{P}}_{\text{perception}} \rightarrow \underbrace{\mathcal{D}}_{\text{decision-maker}} \rightarrow \underbrace{\mathcal{A}}_{\text{actuator model}}$$

Here,  $\mathcal{S}_{\text{logic}}$  refers to the scenario execution layer implemented in `esmini`. It ensures logical consistency while being independent of rendering realism. Only the rendering module  $\mathcal{R}$  varies between experimental conditions, enabling the isolation of visual fidelity as an independent variable.

All other blocks remain invariant, ensuring that any behavioural difference  $\Delta\mathcal{D}$  originates in render-side fidelity variations [11]. This closed-loop structure reflects how fidelity influences the full decision-making chain in an ADAS pipeline, from sensor input to actuation.

In a *closed-loop* setup the simulator, the perception-planning software, and the virtual vehicle continuously feed one another:

1. the simulator renders camera (and other sensor) frames for the current moment in the scene,
2. those frames are processed by the *real* vision stack and decision module exactly as they would be in a test car,
3. the resulting steering, throttle, and brake commands are sent back to the simulator, which moves the ego vehicle accordingly before generating the next set of frames.

Because every action produced by the software immediately influences what the sensors “see” next, the loop is *closed*. This contrasts with *open-loop* or “offline” replay, where recorded frames are simply analyzed on a desktop PC and the outputs are *not* applied to the vehicle model, so the software never experiences the consequences of its own decisions.

This thesis relies on the closed-loop property: only when the controller’s steering and braking commands are reapplied to the virtual car can observations be made about how different rendering qualities propagate through the whole perceive-decide-act chain and alter metrics of interest.

## 2.5 Performance-Evaluation Metrics

Closed-loop fidelity experiments require metrics that move *continuously* with perception quality, not merely pass/fail flags which could miss pinpointing subtle changes in results. Three variables—chosen for their prevalence in the AEB and lane-keeping literature and for the ease with which they can be extracted from our simulation tests—are the quantities we measure in all experiments:

$$\begin{aligned} t_{\text{AEB}} &: \text{Simulation time at AEB activation (ms; logged in ns)} \\ d_{\text{AEB}} &: \text{Distance to lead object at AEB activation (m)} \\ \text{TTC}_{\text{AEB}} &: \text{Time-to-Collision at AEB activation (s)} \end{aligned}$$

### 2.5.1 AEB Activation Time $t_{\text{AEB}}$

Euro New Car Assessment Programme (Euro NCAP; hereafter “NCAP”) defines AEB activation time ( $t_{\text{AEB}}$ ) as the instant the system initiates autonomous braking [12]. Formally, this is determined by identifying the last time point where the vehicle’s longitudinal (forward-direction) acceleration drops below  $-1 \text{ m/s}^2$ , and then tracing back to the point where it first crossed below  $-0.3 \text{ m/s}^2$ . This method captures the moment just before the braking action meaningfully begins.

The simulation advances with a fixed timestep  $\Delta t = 25 \text{ ms}$  ( $2.5 \times 10^7 \text{ ns}$  in logs). Activation time  $t_{\text{AEB}}$  is recorded at the first step where the AEB trigger is set and

*reported in milliseconds.*

### 2.5.2 Distance to Lead Object at AEB Activation $d_{\text{AEB}}$ (longitudinal)

To reflect the forward collision geometry, we define distance along the road/ego vehicle longitudinal axis. Let  $\hat{\mathbf{s}}(t) \in \mathbb{R}^3$  be the ego vehicle's longitudinal unit vector at time  $t$  (e.g., the road-tangent in the OpenDRIVE  $s$ -frame from §2.2.1, or equivalently the ego vehicle's forward heading in world coordinates). Define the signed longitudinal gap

$$g_{\parallel}(t) = (\mathbf{p}_{\text{lead}}(t) - \mathbf{p}_{\text{ego}}(t))^{\top} \hat{\mathbf{s}}(t).$$

where  $\mathbf{p}_{\text{lead}}(t)$  and  $\mathbf{p}_{\text{ego}}(t)$  denotes the position vectors (m) of the lead (front) and ego vehicles respectively. Here,  $(\cdot)^{\top}$  denotes vector transpose.

We report

$$d_{\text{AEB}} = g_{\parallel}(t_{\text{AEB}}).$$

When OpenDRIVE arc-length coordinates are available, this is equivalent to  $d_{\text{AEB}} = s_{\text{lead}}(t_{\text{AEB}}) - s_{\text{ego}}(t_{\text{AEB}})$ . In our scenarios the lead object is ahead at activation, so  $d_{\text{AEB}} \geq 0$ .

### 2.5.3 Time-To-Collision at AEB Activation $\text{TTC}_{\text{AEB}}$ (longitudinal)

We likewise compute TTC using the *longitudinal closing speed*. Let

$$v_{\text{rel},\parallel}(t) = (\mathbf{v}_{\text{ego}}(t) - \mathbf{v}_{\text{lead}}(t))^{\top} \hat{\mathbf{s}}(t),$$

so that  $v_{\text{rel},\parallel} > 0$  indicates the ego vehicle is closing the longitudinal gap, where  $\mathbf{v}_{\text{ego}}(t)$  and  $\mathbf{v}_{\text{lead}}(t)$  denotes the velocity vectors ( $ms^{-1}$ ) of the ego and lead vehicles respectively. Then

$$\text{TTC}_{\text{AEB}} = \begin{cases} \frac{g_{\parallel}(t_{\text{AEB}})}{v_{\text{rel},\parallel}(t_{\text{AEB}})}, & \text{if } v_{\text{rel},\parallel}(t_{\text{AEB}}) > 0, \\ \text{undefined}, & \text{otherwise.} \end{cases}$$

With OpenDRIVE  $s$ -coordinates this reduces to the familiar 1-dimensional form  $\text{TTC}_{\text{AEB}} = \frac{s_{\text{lead}} - s_{\text{ego}}}{\dot{s}_{\text{ego}} - \dot{s}_{\text{lead}}}$  evaluated at  $t_{\text{AEB}}$ , defined only when  $\dot{s}_{\text{ego}} > \dot{s}_{\text{lead}}$ .

### 2.5.4 Alignment with Safety Standards

While this thesis primarily uses timing-based metrics such as AEB activation frame and Time-to-Collision (TTC) to evaluate system performance, these metrics are also relevant to broader safety validation frameworks.

The ISO/PAS 21448 standard, also known as Safety of the Intended Functionality (SOTIF), emphasizes the need to test not only for functional correctness but also

for system behaviour in insufficiently specified or unknown conditions [13]. This is particularly relevant for ADAS perception systems, which can behave unpredictably under unusual lighting, occlusion, or scene clutter.

In parallel, NCAP has outlined ambitions to incorporate simulation into future safety assessments [14]. Their Roadmap 2030 includes increased reliance on digital twins and scenario-based testing, where visual realism plays a crucial role in evaluating system robustness.

These developments support the importance of high-fidelity simulation as explored in this thesis. By evaluating system behaviour under varied visual conditions, simulation-based testing can help identify perception limitations earlier in the development cycle and contribute to safer ADAS deployment.

## 2.6 Core Perception Module: Zenseact DLOD

**Zenseact DLOD** (deep learning object detection) is a *single-frame* monocular detector built on the open-source Fully Convolutional One-Stage 3D (FCOS3D) framework [15]. In FCOS3D, every rendered frame is passed through DLOD to yield a list of 3D objects—class, position, size, yaw, depth and ground-plane velocity—that a downstream tracker smooths over time.

### 2.6.1 Why FCOS3D?

FCOS3D extends the 2D Fully Convolutional One-Stage (FCOS) family of anchor-free detectors [16]. Instead of sliding thousands of hand-tuned “anchor” boxes across the image to test for object presence, FCOS3D predicts object properties directly at each spatial location in the feature map. In other words, each pixel in the multi-scale feature pyramid can independently predict (or vote for) an object’s class, position, and size if it lies near that object’s projected centre. This technique, among others, has led FCOS3D to achieve 1st place out of all vision-only methods in the nuScenes 3D detection challenge of NeurIPS 2020 [15], making it a solid base for any perception-based tasks.

### 2.6.2 Key Design Choices

- **Anchor-free voting:** Each pixel directly predicts object attributes without using predefined anchor boxes. This design reduces the number of hyperparameters and improves recall for very small or very large objects compared with anchor-based approaches.
- **Scale-aware pyramid:** Objects are routed to the appropriate feature-pyramid level by simple 2D size rules.
- **Gaussian centre-ness.** A 2D Gaussian quality score down-weights low-confidence detections before non-maximum suppression [15].

- **Distance-based assignment:** When boxes overlap, the sample point is assigned to the object whose projected centre is nearest, boosting recall for large actors such as buses and trailers.

### 2.6.3 Performance

On the public nuScenes camera-only benchmark, FCOS3D reaches a *mean Average Precision* (mAP) of 0.358 and a *nuScenes Detection Score* (NDS) of 0.428. NDS is the headline metric for the dataset; it blends mAP with five true-positive quality terms—translation (ATE), scale (ASE), orientation (AOE), velocity (AVE) and attribute (AAE) errors—so a higher score means both good classification and well-shaped 3-D boxes. These figures exceed the open-source CenterNet baseline while requiring roughly one third of the training time [15].

Strengths include accurate yaw estimation and strong precision on small objects such as traffic cones and barriers. Weaknesses are depth over-estimation of distant large vehicles and missed detections under heavy occlusion.

## 2.7 Game-Engine Rendering Pipelines

This section gives a short description of the two game engines Unity and UE5 by mentioning core rendering technologies used.

### 2.7.1 Unity URP

Unity’s Universal Render Pipeline emphasises forward lighting and single-bounce shadows [5]. Frame time scales roughly linearly with pixel count; global illumination is approximated via pre-computed light-maps [17]. Such choices typically yield real-time performance even on embedded or mobile hardware [18], but omit multiple-scattering, spectral reflections and sensor high-dynamic-range artifacts.

### 2.7.2 Unreal Engine 5 + Lumen/Path Tracing

Unreal Engine 5 (UE5) introduces hardware-accelerated Nanite meshes (see section §2.7.2.2) and the Lumen real-time global-illumination (GI) system [19]. For offline export, the built-in Path Tracer provides unbiased lighting, area shadows and spectral reflections [20].

#### 2.7.2.1 Why UE5 for the Fidelity Sweeps

This thesis requires (i) scalable scene enrichment and (ii) controllable, physically-based lighting that can be varied per run without light baking. UE5 satisfies (i) via Nanite (virtualized geometry with fine-grained streaming and automatic level-of-detail (LOD)) together with the PCG framework for fast forest/roadside generation; and (ii) via Lumen, which supplies dynamic global illumination and soft shadowing.

### 2.7.2.2 Nanite Virtualized Geometry

Nanite encodes meshes as a hierarchy of cluster tiles that are streamed to the graphics processing unit (GPU) only when visible. This virtualised rasterisation removes manual LOD creation and keeps frame cost proportional to on-screen pixel coverage [21].

### 2.7.2.3 Lumen Dynamic Global Illumination

Lumen gathers direct lighting in screen space, injects it into a signed-distance-field surface cache, and traces diffuse and specular bounces using a hybrid of screen-space and hardware ray tracing, depending on GPU capability. The cache is updated each frame, enabling fully dynamic GI without requiring baked lightmaps [22].

As shown in Figure 2.3, Lumen is capable of simulating advanced lighting effects such as colour bleeding and soft indirect shadowing in real time. These effects enhance scene richness and photometric realism, contributing to more challenging and representative conditions for perception stacks.



Figure 2.3: Demonstration of Lumen global illumination in UE5.

## 2. Theory

---

# 3

## Simulation System Design

To answer our research questions we needed to replay the same traffic scenario while changing only the visual rendering. This required a closed-loop simulation setup with step-synchronous communication between the scenario player, the renderer, and the vision stack. This chapter describes that simulation loop and how the components communicate.

### 3.1 System Architecture Overview

A lightweight *orchestration container* starts three processes: `esmini`, UE5 and Zenseact’s vision stack. From this point on, we refer only to those programs themselves:

- **Fixed-step advancement:** `esmini` advances the scenario in fixed simulation steps. For each step it computes and updates the positions and rotations of moving objects based on the *previous step’s* control outputs from the vision stack,
- **State propagation to the renderer:** The updated actor states and the current simulation timestamp are provided to the game engine (UE5). The engine applies these states to the scene for the same step,
- **Image acquisition:** UE5 renders the camera frame corresponding to that step and forwards the image to the vision stack,
- **Perception and decision:** The vision stack analyses the frame and outputs control decisions (e.g., brake, maintain speed, change lane). These decisions constitute the control signals for the next simulation step,
- **Actuation application:** On the subsequent step, `esmini` applies the received control signals, advances the world state accordingly, and the cycle repeats,
- **Time synchronization:** All components are aligned by the step index and simulation timestamp supplied by `esmini`. Preserving this ordering and these timestamps ensures determinism and repeatability.

To realise the closed-loop and enable controlled fidelity variations, we implemented:

- **FIFO-based inter-process communication (IPC) and message schema:** A per-step request/acknowledgement protocol (*tick-tock*) between `esmini` and

UE5, with `TickMessage` (poses/IDs,  $t_k$ ,  $\Delta t$ ) and `TockMessage` (acknowledgement) (§3.2.1),

- **esmini runner:** The `EsminiRunner::Step()` loop that collects dynamic ground-truth, sends `TickMessages`, waits for the `TockMessage`, and advances the logical time (§3.3.1),
- **UE5 integration:** A `GameInstance`-based module with pre/post frame hooks to apply per-step state, capture and forward images to the vision stack, and send the `TockMessage` (§3.3.2),
- **Scene construction & configuration:** An OpenDRIVE road-mesh builder and a small configuration layer to toggle visual factors (e.g., procedural clutter) (§3.3.4, §3.3.5),
- **Camera capture & streaming:** Render-target readout and step-aligned frame streaming to the vision stack (§3.3.6, §3.2.2).

## 3.2 Synchronization & Communication Protocol

The closed-loop system requires precise synchronization between esmini, UE5, and the vision stack to ensure deterministic, repeatable simulations. This section describes the inter-process communication mechanisms (§3.2.1) and the three-party handshake protocol (§3.2.2) that coordinate state updates, rendering, and perception across simulation steps.

### 3.2.1 Inter-Process Communication

Inter-process communication (IPC) between esmini, UE5, and the vision stack is implemented through a combination of named pipes (FIFOs) and TCP sockets. A pair of FIFOs [23] establish a per-step request/acknowledgement protocol (*tick-tock*) between esmini and UE5 to ensure deterministic synchronization: *Tick* = step state from esmini to UE5 (updated poses + timestamp); *Tock* = UE5’s acknowledgement for the same step after rendering. esmini advances the simulation only after receiving the corresponding *Tock*.

Table 3.1: FIFOs used for synchronization.

FIFO path	Direction	Payload struct
/tmp/myfifo	esmini → UE5	<code>TickMessage</code>
/tmp/myfifo2	UE5 → esmini	<code>TockMessage</code>

In parallel, UE5 transmits rendered images and control signals to vision stack over a Transmission Control Protocol (TCP) socket, another form of IPC similar to FIFOs but operating over a network [24]. All exchanged data (actor states, images, control signals, logs) are bound to a unique simulation step index and timestamp. The C++ structs for these messages are detailed in Listing 1.

### 3.2.2 Synchronous Image Loop

Figure 3.1 illustrates the deterministic three-party handshake.

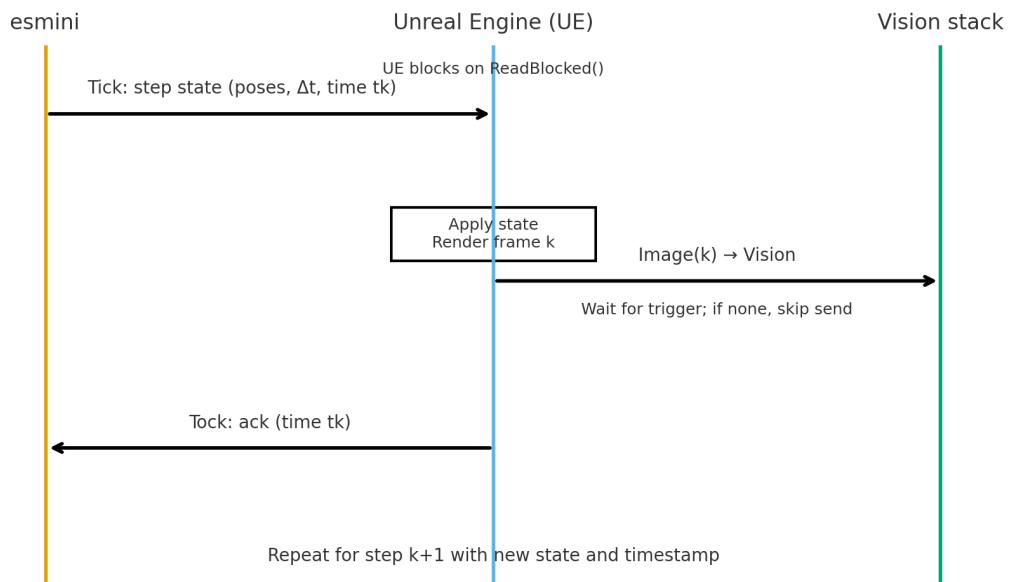


Figure 3.1: Per-step synchronization between `esmini`, UE5, and the vision stack. `esmini` sends the step state (Tick), UE5 renders and conditionally ships the image to the vision stack, then acknowledges the same step (Tock).

1. `esmini` sends updated actor positions/orientations together with the step index and timestamp to UE5 via `/tmp/myfifo`. UE5 blocks in `ReadBlocked()` until the state is available.
2. UE5 applies the received state, renders the camera frame for the same step, waits for the trigger byte on the socket, and—if triggered—sends the image to the vision stack. If the trigger does not arrive in time, sending is skipped for that step.
3. UE5 writes an acknowledgement with the (same) step index/timestamp to `/tmp/myfifo2`. `esmini` blocks in `ReadBlocked()` and reads the acknowledgement to confirm synchronization before advancing to the next step.

## 3.3 Implementation Details

This section describes the specific C++ implementations for the core simulation logic endpoints, scene generation, and sensor configuration.

### 3.3.1 `esmini` Endpoint

`esmini` keeps track of the simulation steps. At each step, it collects ground-truth, computes updated positions and orientations for moving objects, sends this infor-

mation to UE5, and *blocks* until UE5 returns an acknowledgement for the same timestamp. After the acknowledgement, `esmini` advances the logical simulation time and applies the control outputs from the vision stack on the next step. This fixed-step, acknowledged progression (see Listing 2) ensures determinism and comparability across runs.

#### 3.3.2 UE5 Endpoint

The UE5 endpoint applies `esmini`'s step state to the scene, renders the corresponding camera frame, ships the image to the vision stack, and then acknowledges the step back to `esmini`. The `GameInstance` initializes the FIFOs during startup (Listing 3) and registers two frame-bracketing delegates: a pre-render hook (`OnPreTick`) to receive and apply transforms, and a post-render hook (`OnPostTick`) to dispatch the image and send the acknowledgement.

At startup, UE5 opens the FIFO endpoints (incoming step state, outgoing acknowledgement) and registers the pre/post frame delegates. This establishes the communication FIFOs used by the handshake, as shown in Listing 3.

Before rendering a frame for each step, UE5 *blocks* on the inbound FIFO, receives the step state from `esmini`, validates the step index/timestamp, and applies the provided transforms to scene actors (Listing 4). This guarantees that the rendered frame reflects exactly the world state computed by `esmini`.

After rendering the frame for the current step, UE5 forwards the image to the vision stack and then sends an acknowledgement back to `esmini` (Listing 5). The acknowledgement signals that image production/dispatch for the step was completed, allowing `esmini` to advance to the next step. Ordering and timestamp equality provide the synchronization guarantee.

#### 3.3.3 Timing and Determinism

UE5 waits for the vision stacks trigger byte *inside* the `OnPostTick` event, after the current frame has executed, so the presence of the vision stack cannot disturb determinism. Furthermore, there is a timeout mechanism for this process. When the time runs out, UE5 simply skips sending the data and proceeds to the next step, guaranteeing determinism.

#### 3.3.4 Road-mesh Builder

Road geometry is generated by the function in Listing 6. OpenDRIVE metre-based, right-handed coordinates  $(x, y, z)$  [m] are mapped to Unreal's left-handed, centimetre-based coordinates by  $(x, y, z)_{\text{UE5}} = (+100x, -100y, +100z)$ . (Note that these coordinates are scaled up by a factor of 100, metre-based to centimetre-based).

### 3.3.5 Procedural Generation

UE5's PCG system generates points for vegetation and rocks to be created at. The textures are Nanite-based, making it possible for the system to handle a big amount of them. These are generated inside a **BoxComponent** that is created around the road. Density of created vegetation and rocks is set at run-time, as shown in Listing 7.

### 3.3.6 Camera Setup

A Volvo JavaScript Object Notation (JSON) file lists every camera's local pose, render-target size, crop window and optical centre. The class **AEgoCarManager** responsible for storing information about the ego car creates one camera class (**AEgoCarCameraActor**) per defined camera for the specific ego car (Listing 8). Each camera capture follows the process in Listing 9, which involves capturing the scene to a render target, reading the pixels from the graphics processing unit, and sending this pixel data to the vision stack.

### 3. Simulation System Design

---

# 4

## Experimental Methodology

This chapter describes the experimental setup used to evaluate the impact of rendering fidelity and presents the statistical methods used to analyze the resulting data. Section §4.1 present measures taken in order to preserve alignment between the two game-engines. Section §4.2 details the fidelity factors tested and the NCAP-derived scenarios. Section §4.3 describes data collection and processing. Section §4.4 presents the statistical framework, decomposing the research questions into testable hypotheses and detailing the analysis methods for baseline alignment, fidelity effects, and failure mode detection.

### 4.1 Game-Engine Alignment

To increase the likelihood that any behavioural differences originate from the difference in visual fidelity between Unity and UE5, the following aspects are kept as identical as possible across both pipelines:

- **Road network and traffic logic:** Identical OpenDRIVE maps and OpenSCENARIO storyboards are parsed by `esmini` and streamed, step-for-step, to the two renderers. The pedestrians and vehicles are therefore kept in the same position and orientation at simulation step  $k$ .
- **Sensor model:** The virtual camera inherits the same resolution, crop window and fisheye lens distortion. Exposure adaptation is disabled in both engines.
- **Actor appearance:** The car and pedestrian meshes were imported unchanged, and their materials were manually recreated in UE5 using the same albedo, normal-map and metallic-roughness textures as in the Unity pipeline. Because the two engines employ different shading models, minor visual differences may remain, but every effort was made to match colors and gloss levels as closely as possible.
- **Simulation clock:** `esmini` remains the single source of time (§3.3.3), issuing a fixed  $\Delta t$  **Tick**. Unity and UE5 render a new frame only after receiving that Tick, thereby sampling the simulation on an identical timeline.
- **Image I/O path:** Images are transferred to the vision stack over an identical TCP socket and logged using the Hierarchical Data Format version 5 (HDF5) schema described in §4.3.

## 4. Experimental Methodology

---

By aligning these aspects, the renderer becomes the sole experimental variable. Any statistically significant change in closed-loop metrics should therefore reflect differences in visual fidelity alone.

To confirm that no inherent engine characteristics bias the results, we first compare Unity and UE5 with all fidelity features disabled and summarize paired differences per scenario. Once close agreement is established, we analyze fidelity factors—within UE5 only—to isolate the effects of lighting and scene complexity without cross-engine confounds.

## 4.2 Experimental Design

### 4.2.1 Rendering Fidelity Design

We use a partially crossed factorial design in UE5 to isolate how visual phenomena influence perception and control. Starting from the UE5 baseline (plain white sky, no shadows, no environmental enrichment), three primary fidelity factors are explored (see Table 4.1):

Table 4.1: Visual-fidelity factors explored in UE5

Factor	Levels
(1) Environmental clutter (PCG-scattered trees, saplings, tall grass & rocks)	Off
	On
(2) Skybox	Off (plain white, Unity baseline)
	On (blue sky with volumetric clouds)
(3) Shadows	Off
	On

Figure 4.1 illustrates the effect of enabling each of the three fidelity factors, using consistent scene geometry and camera angle. Shadows, when shown, use a zenith sun (90° elevation).

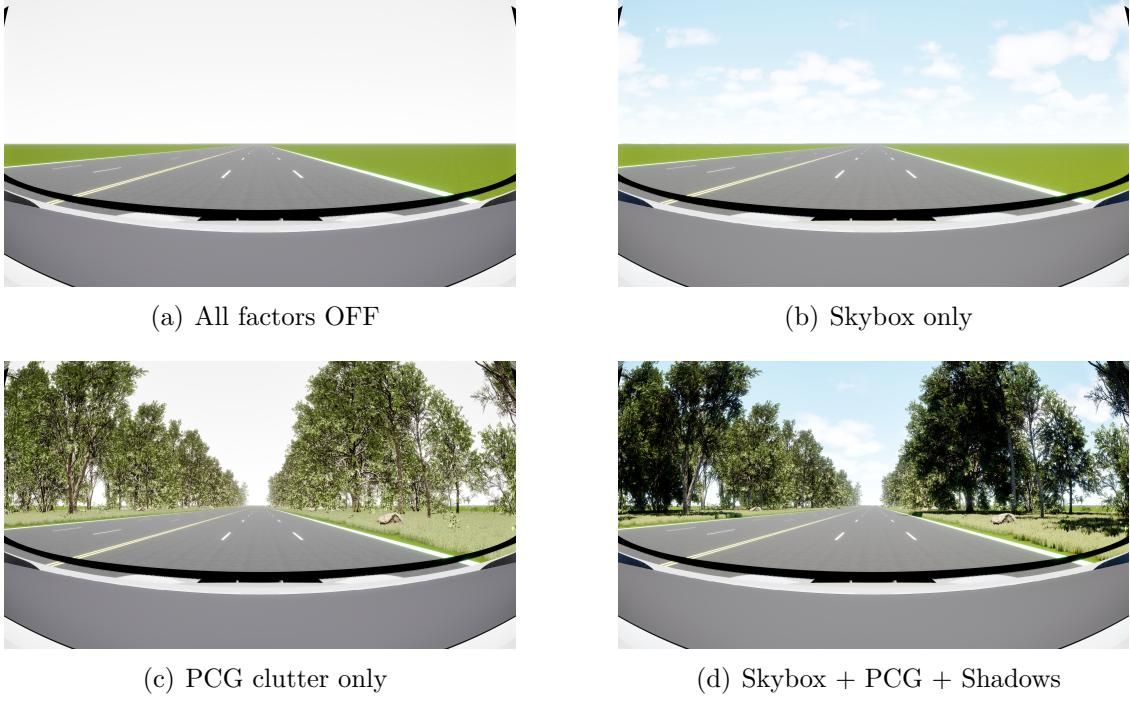


Figure 4.1: Effect of enabling each of the three fidelity factors.

In addition to toggling these three factors, we varied the illumination geometry when shadows were enabled. Lower sun elevations produce longer and more intrusive shadows and slightly warmer lighting. Figure 4.2 shows this effect for a single sun direction: from left to right the elevation is 90° (zenith), 45°, and 10°.



Figure 4.2: Effect of sun elevation angle on lighting and shadows, with sun fixed in the west. From left to right: 90° (zenith), 45°, and 10° above the horizon. Lower sun angles produce longer, softer shadows and warmer lighting effects.

Table 4.2: Sun-light directions tested when *shadows = On*

<b>Label</b>	<b>Cardinal direction</b>	<b>Elevation</b>
S <sub>0</sub>	Zenith (straight up)	90°
S <sub>1-S<sub>4</sub></sub>	N, E, S, W	45°
S <sub>5-S<sub>8</sub></sub>	N, E, S, W	10°

## 4. Experimental Methodology

---

The lower the sun is above the horizon, the more extreme the shadows cast by objects in the scene are. Two different elevations are therefore tested ( $45^\circ$ ,  $10^\circ$ ), each combined with all four cardinal directions, giving the 9 lighting directions listed in Table 4.2.

Testing all four cardinal directions at each elevation serves to reduce bias in the experimental design. Shadow orientation relative to the ego and lead vehicles can systematically affect detection depending on the scenario geometry and approach angle. By testing all four cardinal directions (N, E, S, W) at each elevation level, we ensure that observed elevation effects represent genuine responses to shadow length and lighting intensity rather than artifacts of a particular shadow orientation. This design choice increases the robustness and generalizability of our findings regarding sun elevation, though direction-specific effects are not modeled separately in the statistical analysis of fidelity effects on continuous outcomes (see §4.4.2.2).

Unity remains fixed in its default configuration and serves as a comparison for the UE5 baseline where all fidelity factors are off. The total number of UE5 test configurations is derived as follows:

- **Shadows = Off:** Only the environmental clutter and skybox toggles are relevant. The  $2 \times 2$  combinations of these two binary factors yield  $2^2 = 4$  configurations,
- **Shadows = On:** Lighting direction now enters. The 9 sun positions listed in Table 4.2 are each paired with the same four clutter/skybox settings, giving  $9 \times 2^2 = 36$  further configurations,
- **All factors off:** The UE5 baseline where all factors are off serves as our baseline, and is not counted towards the total, subtracting one configuration from the total.

$$4 \text{ [shadows off]} + 36 \text{ [shadows on]} - 1 \text{ [baseline]} = 39 \text{ UE5 configurations}$$

We compare the 39 UE5 configurations to one another and use the UE5 with all factors off as baseline reference.

### 4.2.2 NCAP-Derived Safety Scenarios

Each rendering configuration is run in eight traffic scenarios based on two Euro NCAP Automated Emergency Braking protocols: four Car to Car cases (*AEB Car to Car (C2C) v4.3.1* [12]) and four vulnerable-road-user cases (*AEB vulnerable road user (VRU) v4.5.1* [25]). Each scenario is presented with two different initial conditions. Table 4.3 lists the scenarios with their NCAP reference start conditions. All scenarios follow right-handed traffic conventions. The ego vehicle drives on the right side of the road, and pedestrian and vehicle interactions are oriented accordingly.

Table 4.3: NCAP scenarios and initial conditions

Short name	Initial conditions
Car to Car front turn across path	Ego 20 km/h, target 45 km/h
Car to Car front turn across path	Ego 20 km/h, target 60 km/h
Car to Car Rear braking	Ego 50 km/h, target 50 km/h ( decelerate 6 m/s <sup>2</sup> )
Car to Car Rear braking	Ego 50 km/h, target 50 km/h ( decelerate 2 m/s <sup>2</sup> )
Car to Car Rear moving	Ego 50 km/h, target 20 km/h
Car to Car Rear moving	Ego 80 km/h, target 20 km/h
Car to Car Rear stationary	Ego 50 km/h, target 0 km/h
Car to Car Rear stationary	Ego 80 km/h, target 0 km/h
Car to Pedestrian Farside Adult	Ego 50 km/h, target 8 km/h
Car to Pedestrian Farside Adult	Ego 60 km/h, target 8 km/h
Car to Pedestrian Longitudinal Adult	Ego 50 km/h, target 5 km/h
Car to Pedestrian Longitudinal Adult	Ego 80 km/h, target 5 km/h
Car to Pedestrian Nearside Adult	Ego 50 km/h, target 5 km/h
Car to Pedestrian Nearside Adult	Ego 60 km/h, target 5 km/h
Car to Pedestrian Nearside Child Obstructed	Ego 50 km/h, target 5 km/h
Car to Pedestrian Nearside Child Obstructed	Ego 60 km/h, target 5 km/h

Scenarios are treated as *fixed test conditions* rather than a random sample. The 16 scenarios correspond to the predefined AEB test cases available for this thesis; no additional scenarios were generated due to time and resource constraints. Consequently, these scenarios constitute the entire set of test cases considered in this analysis. Accordingly, all results describe performance for *these specific test cases* and are not intended to generalize beyond them.

## 4.3 Data Gathering and Processing

Each closed-loop simulation run generates timestamped controller outputs logged in HDF5 format. This section describes the signals extracted from these logs, the metrics derived for analysis, and the conventions used to identify scenarios and map fidelity configurations.

### 4.3.1 Signals and Derived Metrics

For this thesis we extract the following signals (time-series) from the controller logs:

- Automated Emergency Braking (AEB) activation time,
- $TTC_{AEB}$  (s): Time-to-Collision (TTC) at AEB activation,
- $d_{AEB}$  (m): Distance to Lead Object at AEB activation,
- non-activation indicator  $\mathbb{1}_{\text{no-AEB}} \in \{0, 1\}$ .

Raw data are read with Python’s `h5py`, converted to NumPy arrays, and optionally visualised with `matplotlib`. From these signals we compute per-run metrics. The primary analysis (§4.4) uses Time-to-Collision at activation  $\text{TTC}_{\text{AEB}}$  (s), distance at activation  $d_{\text{AEB}}$  (m), and a non-activation indicator  $\mathbb{1}_{\text{no-AEB}}$  (defined as 1 if no trigger is observed, 0 otherwise).

### 4.3.2 Scenario and Fidelity Labels

Each run is identified by a compact scenario code that encodes the traffic configuration and a key parameter such as speed or deceleration:

`<scenario_code>_speed_<v>` or `<scenario_code>_decel_<a>`.

Examples include `ccrm_speed_50`, `ccrs_speed_80`, `ccrb_decel_2`, and `cpnco_speed_60`. These identifiers are consistent across simulators.

Each UE5 configuration can additionally be denoted by a binary fidelity label indicating whether it is the baseline configuration or not:

- **All factors off (“fidelity-off”):** no shadows, no procedural content, zenith sun, white sky.
- **Any factor on (“fidelity-on”):** any configuration enabling shadows, procedural content, non-zenith sun angles, or a cloudy sky.

This combined labeling scheme allows for uniquely specifying both the scenario and whether it is the baseline or not for each run.

## 4.4 Statistical Analysis Framework

This section details the statistical framework used to evaluate the research questions. It begins by dividing the research questions into components, and then details the specific statistical methods applied to each component.

### 4.4.1 Decomposing the Research Questions

To address the two research questions (RQ) formulated in §1.3, we decompose each into specific, testable analytical components.

#### RQ1: Does increased simulation fidelity lead to changes in ADAS behaviour?

Before testing fidelity effects, we first verify that Unity and UE5 produce equivalent baselines when all fidelity features are disabled. This baseline-alignment step ensures that any behavioural differences observed later can be attributed to rendering fidelity rather than engine discrepancies. Only after confirming close baseline agreement do we quantify within-UE5 fidelity effects on continuous AEB outcomes.

Accordingly, RQ1 is operationalized through two analytical components:

1. **Baseline alignment (validation step):** Verify that Unity and UE5 (*fidelity-off*) yield comparable baseline metrics ( $d_{AEB}$ ,  $TTC_{AEB}$ ) across scenarios.
2. **Fidelity effects:** Quantify how specific fidelity factors within UE5 shift  $d_{AEB}$  and  $TTC_{AEB}$  relative to the verified UE5 baseline.

**RQ2:** Can higher fidelity expose failure modes that remain hidden in low-fidelity simulation, and if so, which fidelity factors are most strongly associated with these failures?

1. **Failure exposure:** Identify scenarios where the baseline activates but at least one UE5 *fidelity-on* run does not.
2. **Failure predictors:** Model the association between fidelity factors and non-activation.

#### 4.4.2 RQ1: Baseline and Fidelity Analysis

We assess baseline alignment using paired scenario-level comparisons with descriptive statistics and visualization (§4.4.2.1). Fidelity effects are quantified via ridge linear regression with cluster bootstrap uncertainty estimation (§4.4.2.2).

##### 4.4.2.1 Baseline Agreement Analysis

Our goal is to verify that Unity and UE5 with fidelity disabled yield comparable baselines so that any effects observed later can be attributed to fidelity changes rather than to simulator differences. Accordingly, we conduct a paired, scenario-level agreement analysis. For each scenario  $s$  and metric  $m$  we form the paired difference

$$\Delta_{s,m} = m_s^{(\text{UE5\_off})} - m_s^{(\text{Unity})}.$$

For each metric we report: (i) the *mean paired difference*  $\bar{\Delta}_m$  with a *95% t-confidence interval*, (ii) the *root mean square error (RMSE)* of the differences, and (iii) the *maximum absolute difference*. To provide a scale-free sense of magnitude, we also report the symmetric percentage difference

$$100 \times \frac{2 \left| m_s^{(\text{UE5\_off})} - m_s^{(\text{Unity})} \right|}{\left| m_s^{(\text{UE5\_off})} \right| + \left| m_s^{(\text{Unity})} \right|}.$$

To complement these numeric summaries, agreement was also checked visually using direct comparison (scatter plots).

**4.4.2.1.1 Rationale for approach.** The same scenario is executed in both simulators, so a *paired* comparison by scenario is the appropriate unit of analysis. Correlation alone can be high even with systematic offsets and therefore does not assess agreement. We summarize *paired differences* because they quantify bias and spread directly on the measurement scale. Given  $n=16$  scenario-level pairs, the mean difference with a *t*-interval is a standard summary; the RMSE of differences complements the mean by reflecting typical magnitude.

#### 4.4.2.2 Fidelity Effects on Continuous Outcomes

To quantify how visual-fidelity factors shift continuous AEB outcomes when activation occurs, we analyze within-scenario differences relative to the per-scenario UE5 (*fidelity-off*) baseline. For each scenario  $s$  and run  $i$ :

$$\Delta y_{is} = y_{is} - y_s^{(\text{base})}, \quad y \in \{d_{\text{AEB}} (\text{m}), \text{TTC}_{\text{AEB}} (\text{s})\}.$$

Where  $y_s^{(\text{base})}$  denotes the UE5 (*fidelity-off*) run for scenario  $s$ . This within-scenario difference removes time-invariant scenario effects before modeling (fixed-effects logic) [26].

We then fit ridge ( $\ell_2$ ) linear models for  $\Delta d_{\text{AEB}}$  and  $\Delta \text{TTC}_{\text{AEB}}$ , quantifying uncertainty with scenario-level cluster bootstrapping; activation is modeled separately via the failure predictors analysis (§4.4.3.2) to avoid undefined continuous metrics.

**4.4.2.2.1 Design and predictors.** We include UE5 runs with successful AEB activation only (continuous metrics are undefined otherwise). This creates a *two-part* analysis: (i) a logistic model for the probability of *any* activation (§4.4.3.2) and (ii) a linear model for the *magnitude* of the activation margins *conditional on activation* (this section). The split avoids mixing undefined values into the continuous outcomes while still accounting for non-activations in a dedicated failure analysis. Binary fidelity factors are encoded as  $\{0, 1\}$ : `Shadow` (on/off), `PCG` (on/off), and `Sky` (cloudy vs. white). Sun elevation is categorical with  $90^\circ$  as the reference via indicators for  $45^\circ$  and  $10^\circ$ . As stated in §4.2.1, the four cardinal sun directions were included in the rendered data to reduce potential bias and ensure that elevation effects reflect an average over different shadow orientations. Accordingly, sun direction is omitted as a predictor for this analysis. This allows the estimated elevation effects to represent directionally balanced lightning conditions while avoiding unnecessary model complexity and collinearity with other fidelity factors. The linear mean model for each outcome is

$$\begin{aligned} \Delta y_{is} = & \beta_0 + \beta_1 \text{Shadow}_{is} + \beta_2 \text{PCG}_{is} + \beta_3 \text{Sky}_{is} \\ & + \beta_4 \mathbb{1}\{\text{Elev} = 45^\circ\}_{is} + \beta_5 \mathbb{1}\{\text{Elev} = 10^\circ\}_{is} \\ & + \varepsilon_{is}. \end{aligned}$$

where  $\beta_0$  is the intercept, representing the expected change when all factors are at reference levels (shadows off, PCG off, white sky,  $90^\circ$  sun). Because the baseline configuration is included in the regression with  $\Delta y = 0$  by construction,  $\beta_0$  is expected to be near zero. Coefficients  $\beta_1$  through  $\beta_5$  quantify the average within-scenario shift associated with enabling shadows, PCG, cloudy sky,  $45^\circ$  elevation, and  $10^\circ$  elevation respectively (each relative to its reference level), and  $\varepsilon_{is}$  represents residual error. Each coefficient estimates the change in outcome ( $\Delta d_{\text{AEB}}$  or  $\Delta \text{TTC}_{\text{AEB}}$ ) when the corresponding factor is present, holding other factors constant.

**4.4.2.2.2 Estimation and uncertainty.** We fit ridge linear regression ( $\ell_2$ ) separately for each outcome to stabilize estimates under correlated binaries and modest signal [27]. The penalty is parameterized as  $\alpha = 1/C$  with  $C = 5$  (intercept unpenalized). To accommodate within-scenario dependence, we compute scenario-level cluster bootstrap percentile intervals: with  $K = 16$  scenarios and  $B = 2000$  replicates, each bootstrap sample draws  $K$  scenarios with replacement, pools all rows from the drawn scenarios, refits the ridge model, and stores coefficients. The 95% confidence intervals (CI) are the 2.5th and 97.5th percentiles of the bootstrap coefficient distribution [28], [29]. Given the small number of scenarios ( $K = 16$ ), coverage may be imperfect and intervals are interpreted cautiously.

**4.4.2.2.3 Rationale.** Within-scenario differencing removes time-invariant scenario effects and avoids the need for mixed-effects structure at small  $K$ . The fidelity factors are binary and partially co-occur (e.g., shadows often with non-zenith elevations), creating collinearity; ridge ( $\ell_2$ ) stabilizes estimates under such correlation and modest signal-to-noise. Scenario-level cluster bootstrapping respects within-scenario dependence by resampling entire scenarios, which is preferable to IID resampling when multiple runs nest within-scenario.

**4.4.2.2.4 Interpretation.** Because we first condition on activation and then model  $\Delta$  relative to the per-scenario baseline, coefficients are *average within-scenario shifts* when a factor is “on” versus “off”, holding other factors fixed. By convention here, negative  $\Delta d_{AEB}$  or  $\Delta \text{TTC}_{AEB}$  implies activation occurs *closer to impact* (riskier). Estimates are associational (not causal) and conditional on activation. Because analyses are restricted to runs with activation, estimates are conditional on activation status; if the fidelity factors also influence activation, conditioning can induce selection (collider) bias. Accordingly, effects are interpreted as descriptive within-activation associations rather than population-average causal effects.

### 4.4.3 RQ2: Failure Exposure and Prediction

Failure exposure is evaluated through scenario specific non-activation rates with Wilson score confidence intervals (§4.4.3.1). Failure predictors are identified using ridge-penalized logistic regression with scenario-level cluster bootstrap (§4.4.3.2).

#### 4.4.3.1 Failure Exposure

For each scenario with Unity activation, we compute the UE5 (*fidelity-on*) failure count and rate

$$C_s = \sum_{i=1}^{n_{\text{on},s}} \mathbb{1}_i^{(\text{on})}, \quad \hat{p}_s = \frac{C_s}{n_{\text{on},s}}.$$

Uncertainty is summarized by the **Wilson score** 95% confidence interval [30]

$$\text{CI}_{\text{Wilson}} = \frac{1}{1 + \frac{z^2}{n}} \left( \hat{p} + \frac{z^2}{2n} \pm z \sqrt{\frac{\hat{p}(1 - \hat{p})}{n} + \frac{z^2}{4n^2}} \right),$$

with  $z = 1.96$ . We summarize uncertainty with Wilson CIs; exact tests are uninformative when any nonzero count makes p trivially small.

**4.4.3.1.1 Caution about randomness.** With one Unity baseline and finite UE5 runs per scenario, we cannot rule out that some UE5 non-activations reflect random variation rather than a systematic fidelity effect. The Wilson intervals shown quantify binomial sampling uncertainty for the observed non-activation rates, but they do not by themselves establish causation. We therefore treat these findings as *screening signals* that motivate targeted follow-up with additional replications.

#### 4.4.3.2 Failure Predictors

We model the probability of an AEB non-activation in UE5 runs as a function of visual-fidelity factors using a ridge-penalized logistic regression with scenario-level cluster bootstrap for uncertainty quantification. Ridge logistic regression is well-suited for sparse events because  $\ell_2$  shrinkage stabilizes estimates and reduces variance [31]. We motivate penalization further by the low events-per-variable (EPV) setting [32] and use bootstrap resampling to obtain uncertainty intervals [28].

**4.4.3.2.1 Rationale.** The logistic model complements the descriptive by-factor failure rates by providing an adjusted association across factors. Failures are rare (15/640; low EPV), so we apply  $\ell_2$ -penalization to reduce variance and use scenario-level cluster bootstrap to reflect within-scenario dependence. Because the model operates in a low-signal, low-EPV regime, we treat estimates as exploratory and report them alongside the simpler rate summaries.

**4.4.3.2.2 Predictors and coding.** Binary factors are coded  $\{0, 1\}$ : `Shadow` (on/off), `PCG` (on/off), and `Sky` (cloudy vs. white). Sun elevation is treated categorically with  $90^\circ$  as the reference level via two indicator variables for  $45^\circ$  and  $10^\circ$ . The linear predictor is

$$\text{logit } P(Y = 1 \mid \mathbf{x}) = \beta_0 + \beta_1 \text{Shadow} + \beta_2 \text{PCG} + \beta_3 \text{Sky} + \beta_4 \mathbb{1}\{\text{Elev} = 45^\circ\} + \beta_5 \mathbb{1}\{\text{Elev} = 10^\circ\},$$

Here,  $\beta_0$  is the log-odds of failure for the reference configuration (no shadows, no PCG, white sky, sun at  $90^\circ$ ). Coefficients  $\beta_1 - \beta_5$  represent the change in log-odds of failure when each factor is present, relative to its reference level, holding other factors constant. For example,  $\exp(\beta_1)$  is the odds ratio for failure when shadows are enabled versus disabled.

**4.4.3.2.3 Estimation and uncertainty.** We fit a ridge-penalized logistic regression ( $\ell_2$ ) to stabilize coefficient estimates under sparse events and correlated predictors [31]. With only 15 failures across 5 predictors (events-per-variable  $\approx 3$ ), penalization helps reduce variance and improve numerical stability. The inverse regularization strength is set to  $C = 5.0$  (larger  $C$  implies weaker penalty), with the intercept left unpenalized. To mitigate class imbalance, we apply a weighted likelihood equivalent to `class_weight="balanced"` in scikit-learn's

`LogisticRegression(penalty=12, solver=lbfgs)`, assigning approximate weights  $w_1 \approx 21.3$  to failure cases and  $w_0 \approx 0.51$  to activations [33].

Uncertainty is quantified using a scenario-level cluster bootstrap analogous to that applied in the continuous-outcome model. With  $K = 16$  observed scenarios and  $B = 2000$  bootstrap replicates, each sample resamples  $K$  scenarios with replacement, refits the ridge logistic model, and stores coefficients. Replicates with no variation in  $Y$  are discarded. The 95% confidence intervals (CI) are the 2.5th and 97.5th percentiles of the bootstrap coefficient distribution. Because entire scenarios are resampled, the bootstrap captures within-scenario dependence, though coverage may be imperfect given the limited number of scenarios.

**4.4.3.2.4 Reporting and interpretation.** We report effects as odds ratios  $\text{OR}_j = \exp(\beta_j)$  with bootstrap 95% CIs computed as  $[\exp(\beta_{j,0.025}^{\text{boot}}), \exp(\beta_{j,0.975}^{\text{boot}})]$ , applying the exponential transform to coefficient percentiles. An  $\text{OR} < 1$  indicates the factor is associated with *lower* failure odds;  $\text{OR} > 1$  indicates *higher* odds. All effects are *associations* conditional on the model structure and observed data; they are not causal claims. Because estimates are both regularized and obtained from a weighted likelihood, bootstrap CIs reflect cluster resampling uncertainty but do not correct shrinkage bias.

#### 4. Experimental Methodology

# 5

## Results

This chapter presents results addressing the two research questions defined in §4.4. Section §5.1 addresses RQ1 by first presenting the baseline agreement analysis (§5.1.1) and then quantifying the impact of fidelity factors on continuous AEB outcomes (§5.1.2). Section §5.2 addresses RQ2 by reporting on failure exposure (§5.2.1) and identifying failure predictors (§5.2.2).

### 5.1 RQ1: Baseline and Fidelity Analysis

This section addresses the first research question by validating the baseline agreement between Unity and the UE5 (*fidelity-off*) configuration, and then quantifying the effect of fidelity factors on continuous AEB activation metrics.

#### 5.1.1 Baseline Agreement

To confirm that any observed differences were due to fidelity and not engine-specific characteristics, we first analyzed the agreement between the Unity baseline and the UE5 (*fidelity-off*) baseline, as per the methodology in §4.4.2.1.

Table 5.1: Baseline alignment (Unity vs. UE5, fidelity-off): matched-pairs summaries.  $\Delta = \text{UE5}_{\text{off}} - \text{Unity}$ .

Metric	Unit	$n$	Mean $\Delta$	95% t-CI	RMSE	Max $ \Delta $
$d_{\text{AEB}}$	m	16	0.4272	[0.0495, 0.8049]	0.8084	1.711
$\text{TTC}_{\text{AEB}}$	s	16	0.0225	[0.0005295, 0.04447]	0.04583	0.1

Across scenarios, UE5 (fidelity-off) yields slightly larger values on average, but the shifts are small in magnitude. The symmetric absolute percentage differences average 2.59% for  $d_{\text{AEB}}$  and 2.87% for  $\text{TTC}_{\text{AEB}}$ , with maxima of 7.35% and 8.70%, respectively (Table 5.2). Overall, typical cross-simulator differences are small (approximately 2–3%).

The direct comparison (scatter) plots (figure 5.1) places Unity values on the  $x$ -axis and UE5 (*fidelity-off*) values on the  $y$ -axis, with the identity line  $y = x$  indicating perfect agreement; points close to this line denote strong alignment between base-

## 5. Results

---

Table 5.2: Per-scenario *symmetric* absolute percentage differences for baseline alignment. Percentage difference is  $100 \times \frac{2|m_s^{(\text{UE5\_off})} - m_s^{(\text{Unity})}|}{|m_s^{(\text{UE5\_off})}| + |m_s^{(\text{Unity})}|}$ .

Scenario	$d_{\text{AEB}} (\%)$	$\text{TTC}_{\text{AEB}} (\%)$
ccftap_speed_20_45_variants	0.10	0.00
ccftap_speed_20_60_variants	5.36	2.82
ccrb_decel_2_variants	0.63	0.00
ccrb_decel_6_variants	0.11	0.00
ccrm_speed_50_variants	0.02	2.06
ccrm_speed_80_variants	1.69	1.44
ccrs_speed_50_variants	0.83	1.57
ccrs_speed_80_variants	1.57	2.86
cpfa_speed_50_variants	5.03	6.67
cpfa_speed_60_variants	5.28	6.67
cpla_speed_50_variants	0.00	0.00
cpla_speed_80_variants	1.63	1.46
cpna_speed_50_variants	0.02	0.00
cpna_speed_60_variants	7.35	8.70
cpnco_speed_50_variants	5.70	5.61
cpnco_speed_60_variants	6.08	6.06
<b>Mean</b>	<b>2.59</b>	<b>2.87</b>
<b>Max</b>	<b>7.35</b>	<b>8.70</b>

lines. Inspecting the plots, points show clear clustering around the identity lines for both  $d_{AEB}$  and  $TTC_{AEB}$ , indicating close agreement.

Taking the results of these agreement checks into account, we regard the two simulators as providing sufficiently comparable baselines, and use the UE5 (*fidelity-off* run as the per-scenario reference for the analysis in §5.1.2.

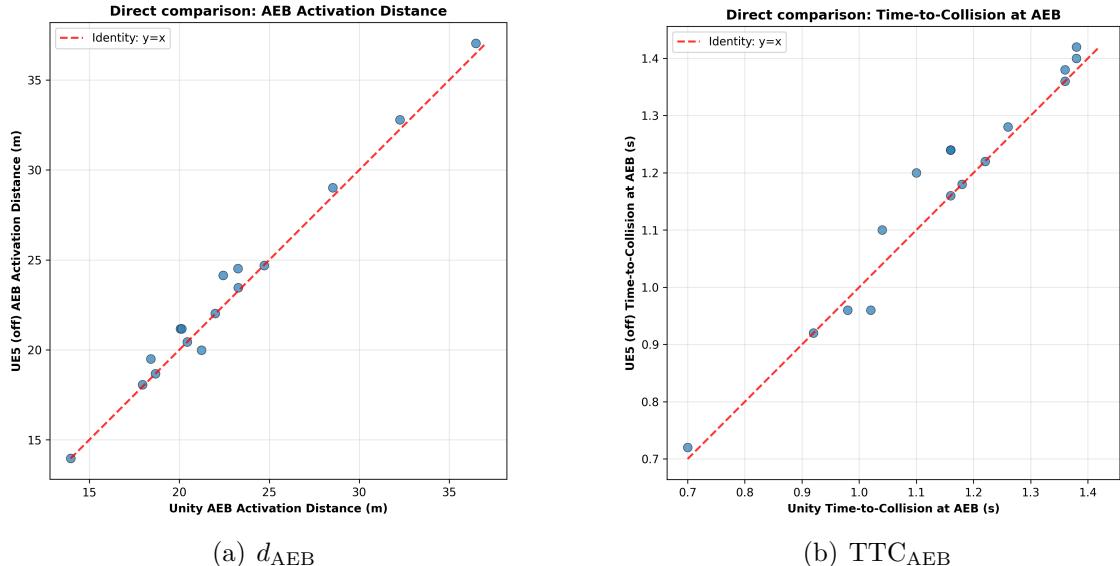


Figure 5.1: Direct comparison (scatter): Unity values ( $x$ ) vs. UE5 (fidelity-off) values ( $y$ ) with the identity line  $y = x$ . Points close to the identity line indicate strong baseline comparability.

### 5.1.2 Fidelity Effects on Continuous Outcomes

After establishing baseline agreement, we fit the ridge linear regression models described in §4.4.2.2 to quantify the association between fidelity factors and continuous AEB outcomes. This analysis used all UE5 runs that resulted in a successful activation.

Across 16 scenarios, there were  $N = 625$  activated UE5 runs used for the continuous-outcome analyses. Coefficients and scenario-cluster bootstrap 95% CIs are reported in Tables 5.3 and 5.4. By our sign convention, negative  $\Delta d$  or  $\Delta TTC$  indicates reduced margin. Uncertainty reflects a scenario-level cluster bootstrap with  $B = 2000$  replicates (degenerate draws discarded); given  $K = 16$  scenarios, intervals are interpreted cautiously.

A consistent pattern emerges for low sun elevation ( $10^\circ$ ):

- $\Delta d_{AEB}$  (m):  $-0.963$   $[-1.789, -0.349]$
- $\Delta TTC_{AEB}$  (s):  $-0.0535$   $[-0.0936, -0.0214]$

Both intervals exclude zero and point in the same “closer” (reduced margin) direction, indicating that low sun shrinks activation margins.

## 5. Results

---

Table 5.3: Ridge linear model for  $\Delta d_{AEB}$  (metres); scenario-cluster bootstrap 95% CI.

Feature	Effect (CI)
Shadow (on)	-0.1676 [-0.5332, 0.1559]
PCG (on)	-0.2671 [-0.5938, 0.0711]
Sky (cloudy vs. white)	0.0773 [-0.0426, 0.2020]
Elevation 45° (vs. 90°)	-0.1311 [-0.4015, 0.0878]
Elevation 10° (vs. 90°)	-0.9628 [-1.7893, -0.3488]

Table 5.4: Ridge linear model for  $\Delta \text{TTC}_{AEB}$  (seconds); scenario-cluster bootstrap 95% CI.

Feature	Effect (CI)
Shadow (on)	-0.0113 [-0.0352, 0.0081]
PCG (on)	-0.0163 [-0.0365, 0.0012]
Sky (cloudy vs. white)	0.0045 [-0.0030, 0.0125]
Elevation 45° (vs. 90°)	-0.0061 [-0.0173, 0.0043]
Elevation 10° (vs. 90°)	-0.0535 [-0.0936, -0.0214]

For other factors, intervals generally include zero and effects are modest:

- **PCG(on)**: shifts toward smaller distance/TTC (e.g.,  $\Delta d = -0.267$  m [-0.594, 0.071];  $\Delta \text{TTC} = -0.0163$  s [-0.0365, 0.0012]) but with CIs overlapping zero.
- **Shadow(on)** and **Sky(cloudy)**: small, imprecise changes around zero across outcomes.
- **Elevation 45°**: near-null with wide CIs.

Overall, the continuous-outcome analysis suggests that *low sun elevation* is a salient fidelity condition that reduces the time/space margin at AEB activation (lower  $\text{TTC}_{AEB}$  and  $d_{AEB}$ ), while other visual-fidelity features exhibit at most modest, uncertain shifts. These findings are conditional on activation and complement the failure-probability analysis in §5.2.2.

## 5.2 RQ2: Failure Exposure and Prediction

This section addresses the second research question by identifying new failure modes (non-activations) exposed by high-fidelity configurations and modeling which fidelity factors are the strongest predictors of these failures.

Across all 656 runs (16 Unity, 640 UE5), we observed 15 non-activations. All 16 Unity baseline runs and all 16 UE5 (*fidelity-off*) baseline runs resulted in successful AEB activation. The 15 observed failures (2.3% of UE5 runs) all occurred in fidelity-on configurations, indicating that increased fidelity can expose failure modes hidden in baseline simulations.

### 5.2.1 Failure Exposure

As detailed in §4.4.3.1, we first analyzed the distribution of these 15 failures across the 16 scenarios. Failures were not distributed evenly; they were concentrated in a few specific scenarios. Table 5.5 lists these five scenarios, along with failure rates and Wilson score 95% confidence intervals.

Table 5.5: Scenarios where higher fidelity exposed AEB failures (baselines activated successfully).

Scenario	Failures	Rate (%)	95% CI (%)
ccrb_decel_2	1/39	2.6	[0.5, 13.2]
ccrm_speed_50	2/39	5.1	[1.4, 16.9]
cpla_speed_50	8/39	20.5	[10.8, 35.5]
cpnco_speed_50	1/39	2.6	[0.5, 13.2]
cpnco_speed_60	3/39	7.7	[2.7, 20.3]

Higher visual fidelity exposed failures in 5 of 16 test scenarios (31.3%), indicating that AEB non-activations occurred selectively under increased rendering realism. Within these affected scenarios, failure rates among fidelity-on variants ranged from 2.6% to 20.5% (Table 5.5). The Wilson score 95% confidence intervals indicate the plausible range of true failure rates for each scenario. In all five cases, the baseline Unity run activated successfully, indicating that rendering fidelity *can expose* failure modes that remain hidden in low-fidelity simulation.

Two scenarios account for most of these events. In `cpla_speed_50` (Car-to-Pedestrian Longitudinal Adult, 50 km/h), 8 of 39 fidelity-on runs failed to trigger AEB (20.5%, Wilson 95% CI  $\approx$  [10.8%, 35.5%]), which is far above the overall UE5 failure rate ( $15/640 \approx 2.3\%$ ). Likewise, `cpnco_speed_60` (Car-to-Pedestrian Nearside Child Obstructed, 60 km/h) failed 3 of 39 times (7.7%, Wilson 95% CI  $\approx$  [2.6%, 20.3%]). That the misses are concentrated in Vulnerable Road User (VRU) cases, and that neither Unity nor UE5 (*fidelity-off*) missed these same scenarios, suggests that added realism (shadows, richer background structure, more competing edges) can make pedestrian rendering harder for the perception stack, so that small visual changes flip the outcome from activation to non-activation.

Each scenario has only one Unity run and one UE5 (*fidelity-off*) run, while the fidelity-on condition has about 39 variants per scenario. The Wilson intervals therefore quantify uncertainty only around the observed UE5 fidelity-on rates; they do not provide a formal comparison against Unity or the UE5 baseline. Our claim is thus descriptive: some fidelity-on renderings produced non-activations that did not occur in the corresponding low-fidelity baselines. To determine whether this reflects systematic sensitivity to fidelity (rather than sampling fluctuation within the 39 runs) we would need repeated runs of identical visual conditions or a hierarchical analysis with replication.

### 5.2.2 Failure Predictors

To understand the association between specific fidelity factors and the 15 observed failures, we fit the ridge-penalized logistic regression model described in §4.4.3.2.

The resulting odds ratios (OR) and 95% cluster bootstrap confidence intervals are presented in Table 5.6.

Table 5.6: Ridge-penalized logistic regression predicting AEB non-activation. Odds ratios (OR) with 95% scenario-cluster bootstrap CIs; baseline is `Shadow=off`, `PCG=off`, `Sky=white`, `Elevation=90°`.

Factor	OR (95% CI)	Significant
Shadow (on/off)	2.75 [0.00, 10.90]	False
PCG (off/on)	0.17 [0.00, 0.76]	True
Sky (cloudy vs. white)	1.68 [0.14, 301.53]	False
Elevation 45° (vs. 90°)	0.19 [0.07, 18.27]	False
Elevation 10° (vs. 90°)	0.17 [0.02, 49.95]	False

OR = odds ratio; CIs from scenario-level cluster bootstrap (2,000 replicates).

In this dataset, the interval for PCG (on/off) does not cover 1, indicating an association with *lower* odds of failure (OR < 1). Intervals for `Shadow` (on/off), `Sky` (cloudy vs. white), and the elevation indicators are wide and overlap 1. Given the small number of failures (15/640) and within-scenario dependence, these are exploratory associations rather than definitive effects. Odds ratios are regularized (ridge), which shrinks estimates toward 1; CIs reflect cluster resampling and do not correct shrinkage bias.

# 6

## Discussion

This chapter interprets the findings of the experimental analysis and reflects on their significance for simulation-based ADAS testing. We focus on what the results imply for using high-fidelity rendering as a stressor, where the current design is limited, and how the approach could be extended.

### 6.1 What the baseline validation establishes

The baseline comparison in §5.1.1 showed that Unity and UE5 (*fidelity-off*) produce closely aligned AEB activation metrics across the 16 NCAP-derived scenarios, with only small per-scenario differences. This supports the key modelling choice in §4.4.2.2 to treat the UE5 *fidelity-off* run as the per-scenario reference when analysing fidelity effects. A residual simulator bias cannot be ruled out completely (different shading/tone mapping pipelines), but the evidence is consistent with subsequent differences being driven mainly by the fidelity manipulations.

### 6.2 How fidelity changes AEB activation margins

The within-scenario delta models in §4.4.2.2 and the estimates reported in §5.1.2 point to a simple pattern: illumination geometry matters more than generic scene enrichment. Runs rendered with a low sun elevation ( $10^\circ$ ) activated closer in space and time than the same scenario rendered with the baseline lighting, while shadows, sky and PCG showed at most modest, uncertain shifts. Because these models are conditional on activation, this should be read as: *when the AEB does activate, low sun tends to reduce the available margin*. This is a useful complement to the failure analysis, because it shows that fidelity does not only produce rare hard misses but can also shrink safety margins in otherwise successful runs.

### 6.3 Higher fidelity exposes hidden non-activations

The failure exposure results in §5.2.1 showed that non-activations did not occur randomly across all 16 scenarios but were concentrated in a small subset of VRU scenarios, in particular `cpla_speed_50` and `cpnco_speed_60`. Both of these activated in Unity and in UE5 (*fidelity-off*), so the simplest interpretation is that

added visual realism made an otherwise solvable perception task fragile. As noted in §5.2.1, each scenario has only one Unity and one UE5 baseline run, whereas the fidelity-on condition has many more variants; the Wilson intervals therefore describe uncertainty *within* UE5 (*fidelity-on*), not a formal test against Unity. Our claim is therefore descriptive: some fidelity-on renderings produced non-activations that did not occur in the corresponding low-fidelity baselines. That is already enough to justify including fidelity sweeps in regression test suites, because it shows that “clean” renders can under-expose risk.

## 6.4 Interpreting the failure model

The ridge-penalised logistic regression in §4.4.3.2 / §5.2.2 was designed to adjust for multiple fidelity factors in a low-events setting, not to deliver definitive odds ratios. Within that exploratory framing, the only factor whose interval did not include 1 was PCG, i.e. scenes with additional structure were associated with *lower* failure odds. Other factors were inconclusive. This is a plausible outcome: extra structure can add cues for the perception stack, but with only 15 failures we cannot separate helpful structure from structure that also introduced the failure at a finer level. Follow-up experiments would have to hold lighting fixed and vary structure (or vice versa) to identify the mechanism.

## 6.5 Practical implications for simulation-based ADAS testing

Two practical conclusions follow from the above:

- **Test beyond the clean baseline:** Since some scenarios only showed non-activations once visual realism was increased, sticking to a single low-fidelity render can give a false sense of robustness,
- **Exercise scenario diversity:** Effects were scenario-dependent and strongest for VRU scenarios. It is therefore more informative to cover different geometries, occlusions and approach speeds than to run many repeats of a single clean scene.

## 6.6 Threats to validity and limitations

Three limitations are especially relevant. First, the analysis is tied to one perception stack and a camera-only setup; other stacks or sensor suites might react differently to the same fidelity changes. Second, the fidelity space we explored was narrow (shadows, sky, PCG, sun elevation); other visual phenomena such as exposure control, weather or night lighting could be stronger failure triggers. Third, the number of scenarios (16) is small, so scenario-level cluster bootstraps must be read cautiously. These limitations mean that the study demonstrates the *existence* of fidelity-induced failures, not their prevalence in a broader operational design domain.

## 6.7 Future work

Future work should therefore (i) widen the scenario and sensor coverage, (ii) vary additional rendering dimensions, especially those that affect exposure/contrast, and (iii) replicate the specific VRU scenarios that showed the highest non-activation rates, but with tightly controlled lighting, so that systematic and random components can be separated.

## 6. Discussion

# Bibliography

- [1] *Global Status Report on Road Safety 2023*, en, 1st ed. Geneva: World Health Organization, 2023, ISBN: 978-92-4-008651-7.
- [2] J. Nidamanuri, C. Nibhanupudi, R. Assfalg, and H. Venkataraman, “A progressive review: Emerging technologies for adas driven solutions,” *IEEE Transactions on Intelligent Vehicles*, vol. 7, no. 2, pp. 326–341, 2022. DOI: 10.1109/TIV.2021.3122898.
- [3] M. Ellims, “Brake systems: A mind of their own,” *Digital Evidence & Elec. Signature L. Rev.*, vol. 18, p. 27, 2021.
- [4] *New Vehicle General Safety Regulation*, sv, Text. Accessed: Apr. 18, 2025. [Online]. Available: [https://ec.europa.eu/commission/presscorner/detail/en/ip\\_22\\_4312](https://ec.europa.eu/commission/presscorner/detail/en/ip_22_4312).
- [5] Unity Technologies, *Universal Render Pipeline Overview*, <https://docs.unity3d.com/Packages/com.unity.render-pipelines.universal@16.0/manual/index.html>, Accessed: 2025-04-23.
- [6] D. Liu, J. Yu, N. D. Macchiarella, and D. A. Vincenzi, “Simulation fidelity,” in *Human factors in simulation and training*, CRC Press, 2008, pp. 91–108.
- [7] ASAM e.V., *ASAM OpenSCENARIO XML Standard*, <https://www.asam.net/standards/detail/openscenario-xml/>, Accessed: 2025-04-22.
- [8] ASAM e.V., *ASAM OpenDRIVE Standard*, <https://www.asam.net/standards/detail/opendrive/>, Accessed: 2025-04-22.
- [9] ASAM e.V., *ASAM OSI Standard*, <https://www.asam.net/standards/detail/osi/>, Accessed: 2025-04-22.
- [10] M. Laukvik and contributors, *esmini — Essential mini openscenario player*, <https://github.com/esmini/esmini>, Accessed: 2025-04-22.
- [11] X. Yang et al., “Drivearena: A closed-loop generative simulation platform for autonomous driving,” *arXiv preprint arXiv:2408.00415*, 2024.
- [12] Euro NCAP, *Euro NCAP AEB C2C Test Protocol - v4.3.1*, <https://www.euroncap.com/media/80155/euro-ncap-aeb-c2c-test-protocol-v431.pdf>, Accessed: 2025-06-14.
- [13] International Organization for Standardization, *Road vehicles – Safety of the intended functionality (SOTIF)*, <https://www.iso.org/standard/77490.html>, ISO 21448:2022, 2022.
- [14] Euro NCAP, *Vision 2030: A Safer Future for Mobility*, <https://cdn.euroncap.com/media/74468/euro-ncap-roadmap-vision-2030.pdf>, Euro NCAP roadmap, published 9November2022, Nov. 2022.

- [15] T. Wang, X. Zhu, J. Pang, and D. Lin, “Fcov3d: Fully convolutional one-stage monocular 3d object detection,” in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 913–922.
- [16] Z. Tian, C. Shen, H. Chen, and T. He, “Fcov: Fully convolutional one-stage object detection,” in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 9627–9636.
- [17] Unity Technologies, *Introduction to Lightmaps and Baking*, <https://docs.unity3d.com/Manual/Lightmappers.html>, Accessed: 2025-04-23.
- [18] Unity Technologies, *Optimize Your Mobile Game Performance: Expert Tips on Graphics and Assets*, <https://unity.com/blog/games/optimize-your-mobile-game-performance-expert-tips-on-graphics-and-assets>, Accessed: 2025-04-23.
- [19] Epic Games, *Unreal Engine 5.0 Release Notes*, [https://dev.epicgames.com/documentation/en-us/unreal-engine/unreal-engine-5.0-release-notes?application\\_version=5.0](https://dev.epicgames.com/documentation/en-us/unreal-engine/unreal-engine-5.0-release-notes?application_version=5.0), Accessed: 2025-04-23.
- [20] Epic Games, *Path Tracer in Unreal Engine*, <https://dev.epicgames.com/documentation/en-us/unreal-engine/path-tracer-in-unreal-engine/>, Accessed: 2025-04-23.
- [21] Epic Games, *Nanite Virtualized Geometry in Unreal Engine*, <https://dev.epicgames.com/documentation/en-us/unreal-engine/nanite-virtualized-geometry-in-unreal-engine/>, Accessed: 2025-04-23.
- [22] Epic Games, *Lumen Global Illumination and Reflections in Unreal Engine*, <https://dev.epicgames.com/documentation/en-us/unreal-engine/lumen-global-illumination-and-reflections-in-unreal-engine>, Accessed: 2025-04-23.
- [23] Canonical, *Pipe - overview of pipes and FIFOs*, <https://manpages.ubuntu.com/manpages/focal/man7/pipe.7.html>, Accessed: 2025-05-18.
- [24] Oracle Corporation, *Overview of sockets*, Oracle Solaris 11.1 Programming Interfaces Guide, 2012. [Online]. Available: [https://docs.oracle.com/cd/E26502\\_01/html/E35299/sockets-42163.html](https://docs.oracle.com/cd/E26502_01/html/E35299/sockets-42163.html).
- [25] Euro NCAP, *Euro NCAP AEB LSS VRU Test Protocol - v4.5.1*, <https://www.euroncap.com/media/80156/euro-ncap-aeb-lss-vru-test-protocol-v451.pdf>, Accessed: 2025-06-14.
- [26] J. M. Wooldridge, *Econometric analysis of cross section and panel data*. MIT press, 2010.
- [27] A. E. Hoerl and R. W. Kennard, “Ridge regression: Biased estimation for nonorthogonal problems,” *Technometrics*, vol. 12, no. 1, pp. 55–67, 1970.
- [28] R. J. Tibshirani and B. Efron, “An introduction to the bootstrap,” *Monographs on statistics and applied probability*, vol. 57, no. 1, pp. 1–436, 1993.
- [29] A. C. Cameron and D. L. Miller, “A practitioners guide to cluster-robust inference,” *Journal of human resources*, vol. 50, no. 2, pp. 317–372, 2015.
- [30] E. B. Wilson, “Probable inference, the law of succession, and statistical inference,” *Journal of the American Statistical Association*, vol. 22, no. 158, pp. 209–212, 1927.

- [31] S. L. Cessie and J. V. Houwelingen, “Ridge estimators in logistic regression,” *Journal of the Royal Statistical Society Series C: Applied Statistics*, vol. 41, no. 1, pp. 191–201, 1992.
- [32] P. Peduzzi, J. Concato, E. Kemper, T. R. Holford, and A. R. Feinstein, “A simulation study of the number of events per variable in logistic regression analysis,” *Journal of clinical epidemiology*, vol. 49, no. 12, pp. 1373–1379, 1996.
- [33] F. Pedregosa et al., “Scikit-learn: Machine learning in python,” *the Journal of machine Learning research*, vol. 12, pp. 2825–2830, 2011.

## Bibliography

---

# A

## Comprehensive Results and AEB Metrics

The following tables present raw AEB metric data per variant for each scenario.

## A. Comprehensive Results and AEB Metrics

---

Table A.1: AEB metrics for scenario: Car to Car front turn across path  
(ccftap\_speed\_20\_45)

Source	aeb_activated	Shadow	PCG	Sun Direction	Elevation	Cloud	$t_{AEB}$ [ms]	$d_{AEB}$ [m]	TTC <sub>AEB</sub> [s]
Unity (Baseline)	True	False	False		90	False	12475	21.992603	0.920000
UE5 (Baseline)	True	False	False		90	False	12475	22.014822	0.920000
UE5	True	False	False		90	True	12350	24.225832	1.040000
UE5	True	False	True		90	False	12550	20.780243	0.860000
UE5	True	False	True		90	True	12550	20.778925	0.840000
UE5	True	True	False		90	False	12475	21.994530	0.920000
UE5	True	True	False		90	True	12475	22.034081	0.920000
UE5	True	True	False	North	10	False	12475	21.979010	0.920000
UE5	True	True	False	North	10	True	12475	21.988857	0.920000
UE5	True	True	False	North	45	False	12475	22.005295	0.920000
UE5	True	True	False	North	45	True	12475	22.046341	0.920000
UE5	True	True	False	East	10	False	12550	20.656324	0.840000
UE5	True	True	False	East	10	True	12550	20.695677	0.840000
UE5	True	True	False	East	45	False	12500	21.573200	0.900000
UE5	True	True	False	East	45	True	12475	22.026516	0.920000
UE5	True	True	False	South	10	False	12550	20.652475	0.840000
UE5	True	True	False	South	10	True	12550	21.058401	0.860000
UE5	True	True	False	South	45	False	12300	24.965027	1.080000
UE5	True	True	False	South	45	True	12350	24.215178	1.040000
UE5	True	True	False	West	10	False	12550	20.641279	0.840000
UE5	True	True	False	West	10	True	12675	18.510876	0.800000
UE5	True	True	False	West	45	False	12525	21.121502	0.860000
UE5	True	True	False	West	45	True	12500	21.600464	0.900000
UE5	True	True	True		90	False	12550	20.698038	0.840000
UE5	True	True	True		90	True	12550	20.685238	0.840000
UE5	True	True	True	North	10	False	12475	22.002485	0.900000
UE5	True	True	True	North	10	True	12325	24.928787	1.060000
UE5	True	True	True	North	45	False	12525	21.086981	0.860000
UE5	True	True	True	North	45	True	12325	24.563889	1.060000
UE5	True	True	True	East	10	False	12325	24.539227	1.060000
UE5	True	True	True	East	10	True	12475	22.005579	0.920000
UE5	True	True	True	East	45	False	12550	20.696495	0.840000
UE5	True	True	True	East	45	True	12475	22.036991	0.920000
UE5	True	True	True	South	10	False	12525	21.048801	0.860000
UE5	True	True	True	South	10	True	12525	21.081856	0.860000
UE5	True	True	True	South	45	False	12475	22.035252	0.920000
UE5	True	True	True	South	45	True	12525	21.104225	0.860000
UE5	True	True	True	West	10	False	12475	21.976866	0.920000
UE5	True	True	True	West	10	True	12525	21.081703	0.860000
UE5	True	True	True	West	45	False	12550	20.663897	0.840000
UE5	True	True	True	West	45	True	12550	20.681263	0.840000

Table A.2: AEB metrics for scenario: Car to Car front turn across path (ccf-tap\_speed\_20\_60)

Source	aeb_activated	Shadow	PCG	Sun Direction	Elevation	Cloud	$t_{AEB}$ [ms]	$d_{AEB}$ [m]	TTC <sub>AEB</sub> [s]
Unity (Baseline)	True	False	False		90	False	12700	20.056802	0.700000
UE5 (Baseline)	True	False	False		90	False	12675	21.161491	0.720000
UE5	True	False	False		90	True	12525	24.831383	0.880000
UE5	True	False	True		90	False	12575	23.737257	0.820000
UE5	True	False	True		90	True	12575	23.772440	0.820000
UE5	True	True	False		90	False	12525	24.283733	0.840000
UE5	True	True	False		90	True	12525	24.307631	0.840000
UE5	True	True	False	North	10	False	12550	24.296915	0.860000
UE5	True	True	False	North	10	True	12675	21.126619	0.720000
UE5	True	True	False	North	45	False	12500	24.939432	0.880000
UE5	True	True	False	North	45	True	12525	24.431171	0.860000
UE5	True	True	False	East	10	False	12675	21.117466	0.740000
UE5	True	True	False	East	10	True	12675	21.521057	0.720000
UE5	True	True	False	East	45	False	12675	21.118774	0.700000
UE5	True	True	False	East	45	True	12675	21.262764	0.720000
UE5	True	True	False	South	10	False	12675	21.528154	0.760000
UE5	True	True	False	South	10	True	12675	21.332275	0.720000
UE5	True	True	False	South	45	False	12525	24.261126	0.860000
UE5	True	True	False	South	45	True	12525	24.801699	0.880000
UE5	True	True	False	West	10	False	12725	20.056803	0.700000
UE5	True	True	False	West	10	True	12700	20.608791	0.700000
UE5	True	True	False	West	45	False	12675	21.145290	0.720000
UE5	True	True	False	West	45	True	12675	21.125105	0.700000
UE5	True	True	True		90	False	12475	25.391768	0.900000
UE5	True	True	True		90	True	12525	24.289616	0.840000
UE5	True	True	True	North	10	False	12475	25.380968	0.900000
UE5	True	True	True	North	10	True	12475	25.968382	0.920000
UE5	True	True	True	North	45	False	12675	21.123457	0.740000
UE5	True	True	True	North	45	True	12475	25.384218	0.900000
UE5	True	True	True	East	10	False	12275	29.527004	1.100000
UE5	True	True	True	East	10	True	12525	24.877047	0.880000
UE5	True	True	True	East	45	False	12675	21.204811	0.720000
UE5	True	True	True	East	45	True	12675	21.148750	0.720000
UE5	True	True	True	South	10	False	12675	21.145466	0.720000
UE5	True	True	True	South	10	True	12675	21.277018	0.720000
UE5	True	True	True	South	45	False	12525	24.289600	0.840000
UE5	True	True	True	South	45	True	12525	24.325066	0.840000
UE5	True	True	True	West	10	False	12550	24.314730	0.860000
UE5	True	True	True	West	10	True	12575	23.727911	0.820000
UE5	True	True	True	West	45	False	12675	21.147051	0.720000
UE5	True	True	True	West	45	True	12550	24.320488	0.860000

## A. Comprehensive Results and AEB Metrics

---

Table A.3: AEB metrics for scenario: Car to Car Rear braking (ccrb\_decel\_6)

Source	aeb_activated	Shadow	PCG	Sun	Direction	Elevation	Cloud	$t_{AEB}$ [ms]	$d_{AEB}$ [m]	TTC <sub>AEB</sub> [s]
Unity (Baseline)	True	False	False			90	False	8425	24.719154	1.360000
UE5 (Baseline)	True	False	False			90	False	8425	24.692036	1.360000
UE5	True	False	False			90	True	8425	24.673859	1.360000
UE5	True	False	True			90	False	8425	24.699368	1.360000
UE5	True	False	True			90	True	8425	24.626497	1.360000
UE5	True	True	False			90	False	8425	24.732346	1.360000
UE5	True	True	False			90	True	8425	24.642298	1.360000
UE5	True	True	False	North		10	False	8425	24.625401	1.360000
UE5	True	True	False	North		10	True	8425	24.749805	1.360000
UE5	True	True	False	North		45	False	8450	24.531530	1.360000
UE5	True	True	False	North		45	True	8425	24.703564	1.360000
UE5	True	True	False	East		10	False	8425	24.646503	1.360000
UE5	True	True	False	East		10	True	8425	24.721703	1.360000
UE5	True	True	False	East		45	False	8425	24.803776	1.380000
UE5	True	True	False	East		45	True	8425	24.669611	1.360000
UE5	True	True	False	South		10	False	8425	24.735056	1.360000
UE5	True	True	False	South		10	True	8425	24.924778	1.360000
UE5	True	True	False	South		45	False	8425	24.791824	1.380000
UE5	True	True	False	South		45	True	8425	24.732981	1.360000
UE5	True	True	False	West		10	False	8425	24.640709	1.360000
UE5	True	True	False	West		10	True	8425	24.812706	1.360000
UE5	True	True	False	West		45	False	8425	24.660370	1.360000
UE5	True	True	False	West		45	True	8425	24.718948	1.360000
UE5	True	True	True			90	False	8425	24.683270	1.360000
UE5	True	True	True			90	True	8425	24.825329	1.380000
UE5	True	True	True	North		10	False	8425	24.725784	1.360000
UE5	True	True	True	North		10	True	8425	24.659800	1.360000
UE5	True	True	True	North		45	False	8425	24.757139	1.360000
UE5	True	True	True	North		45	True	8425	24.642017	1.360000
UE5	True	True	True	East		10	False	8425	24.777044	1.380000
UE5	True	True	True	East		10	True	8425	24.715883	1.360000
UE5	True	True	True	East		45	False	8425	24.733128	1.360000
UE5	True	True	True	East		45	True	8425	24.705868	1.360000
UE5	True	True	True	South		10	False	8425	24.810894	1.380000
UE5	True	True	True	South		10	True	8425	24.791506	1.380000
UE5	True	True	True	South		45	False	8425	24.792330	1.380000
UE5	True	True	True	South		45	True	8425	24.787411	1.380000
UE5	True	True	True	West		10	False	8425	24.718813	1.360000
UE5	True	True	True	West		10	True	8425	24.782213	1.380000
UE5	True	True	True	West		45	False	8425	24.737034	1.360000
UE5	True	True	True	West		45	True	8425	24.718138	1.360000

Table A.4: AEB metrics for scenario: Car to Car Rear braking (ccrb\_decel\_2)

Source	aeb_activated	Shadow	PCG	Sun	Direction	Elevation	Cloud	$t_{AEB}$ [ms]	$d_{AEB}$ [m]	TTC <sub>AEB</sub> [s]
Unity (Baseline)	True	False	False			90	False	10950	17.953394	1.160000
UE5 (Baseline)	True	False	False			90	False	10950	18.066757	1.160000
UE5	True	False	False			90	True	10975	17.867014	1.140000
UE5	True	False	True			90	False	10975	17.882547	1.140000
UE5	True	False	True			90	True	10975	17.908859	1.140000
UE5	True	True	False			90	False	10975	17.807655	1.140000
UE5	True	True	False			90	True	10975	17.851891	1.140000
UE5	True	True	False	North		10	False	10975	17.847755	1.140000
UE5	True	True	False	North		10	True	10975	17.813305	1.140000
UE5	True	True	False	North		45	False	10975	17.875277	1.140000
UE5	True	True	False	North		45	True	10975	17.845407	1.140000
UE5	True	True	False	East		10	False	10975	17.842951	1.140000
UE5	True	True	False	East		10	True	10975	17.881237	1.140000
UE5	True	True	False	East		45	False	10975	17.889086	1.140000
UE5	True	True	False	East		45	True	10975	17.799719	1.140000
UE5	True	True	False	South		10	False	10975	17.863354	1.140000
UE5	True	True	False	South		10	True	10950	18.229933	1.160000
UE5	True	True	False	South		45	False	10950	18.096069	1.160000
UE5	True	True	False	South		45	True	10975	17.842146	1.140000
UE5	True	True	False	West		10	False	10975	17.774826	1.140000
UE5	True	True	False	West		10	True	10975	17.916967	1.140000
UE5	True	True	False	West		45	False	10975	17.8777695	1.140000
UE5	True	True	False	West		45	True	10975	17.851303	1.140000
UE5	True	True	True			90	False	10975	17.867617	1.140000
UE5	True	True	True			90	True	10975	17.832115	1.140000
UE5	True	True	True	North		10	False	10950	17.970354	1.160000
UE5	True	True	True	North		10	True	10975	17.865471	1.140000
UE5	True	True	True	North		45	False	10950	18.083931	1.160000
UE5	True	True	True	North		45	True	10975	17.838030	1.140000
UE5	True	True	True	East		10	False	10975	17.856756	1.140000
UE5	True	True	True	East		10	True	10975	17.856266	1.140000
UE5	True	True	True	East		45	False	10975	17.863388	1.140000
UE5	True	True	True	East		45	True	10950	18.079185	1.160000
UE5	True	True	True	South		10	False	10975	17.862679	1.140000
UE5	False	True	True	South		10	True			
UE5	True	True	True	South		45	False	10975	17.869759	1.140000
UE5	True	True	True	South		45	True	10950	18.054417	1.160000
UE5	True	True	True	West		10	False	10975	17.920374	1.140000
UE5	True	True	True	West		10	True	10975	17.902483	1.140000
UE5	True	True	True	West		45	False	10975	17.865770	1.140000
UE5	True	True	True	West		45	True	10975	17.827946	1.140000

## A. Comprehensive Results and AEB Metrics

---

Table A.5: AEB metrics for scenario: Car to Car Rear moving (ccrm\_speed\_50)

Source	aeb_activated	Shadow	PCG	Sun	Direction	Elevation	Cloud	$t_{AEB}$ [ms]	$d_{AEB}$ [m]	TTC <sub>AEB</sub> [s]
Unity (Baseline)	True	False	False			90	False	3600	13.957398	0.980000
UE5 (Baseline)	True	False	False			90	False	3600	13.959739	0.960000
UE5	True	False	False			90	True	3575	14.300638	1
UE5	True	False	True			90	False	3575	14.178730	1
UE5	True	False	True			90	True	3575	14.180910	1
UE5	True	True	False			90	False	3575	14.198496	1
UE5	True	True	False			90	True	3575	14.215182	0.980000
UE5	True	True	False	North		10	False	3600	13.987144	0.980000
UE5	False	True	False	North		10	True			
UE5	True	True	False	North		45	False	3600	14.037857	0.980000
UE5	False	True	False	North		45	True			
UE5	True	True	False	East		10	False	3575	14.213135	1
UE5	True	True	False	East		10	True	3600	14.147773	1
UE5	True	True	False	East		45	False	3600	14.149997	1
UE5	True	True	False	East		45	True	3600	14.152203	1
UE5	True	True	False	South		10	False	3600	14.037913	0.980000
UE5	True	True	False	South		10	True	3600	14.078103	0.960000
UE5	True	True	False	South		45	False	3600	13.942674	0.960000
UE5	True	True	False	South		45	True	3600	14.159604	1
UE5	True	True	False	West		10	False	3600	13.973252	0.980000
UE5	True	True	False	West		10	True	3600	14.039132	0.980000
UE5	True	True	False	West		45	False	3600	14.122407	1
UE5	True	True	False	West		45	True	3600	14.169868	1
UE5	True	True	True			90	False	3575	14.184154	1
UE5	True	True	True			90	True	3550	14.346720	1.020000
UE5	True	True	True	North		10	False	3600	14.014119	0.980000
UE5	True	True	True	North		10	True	3600	14.003082	0.980000
UE5	True	True	True	North		45	False	3600	14.040095	0.980000
UE5	True	True	True	North		45	True	3600	14.060188	0.980000
UE5	True	True	True	East		10	False	3600	13.986871	0.980000
UE5	True	True	True	East		10	True	3600	14.035812	0.980000
UE5	True	True	True	East		45	False	3600	14.175366	1
UE5	True	True	True	East		45	True	3600	14.051320	0.980000
UE5	True	True	True	South		10	False	3600	14.058595	0.980000
UE5	True	True	True	South		10	True	3575	14.206508	1
UE5	True	True	True	South		45	False	3575	14.215167	1
UE5	True	True	True	South		45	True	3600	14.121293	1
UE5	True	True	True	West		10	False	3600	14.057456	0.980000
UE5	True	True	True	West		10	True	3600	14.160439	1
UE5	True	True	True	West		45	False	3600	14.136890	1
UE5	True	True	True	West		45	True	3600	14.098336	1

Table A.6: AEB metrics for scenario: Car to Car Rear moving (ccrm\_speed\_80)

Source	aeb_activated	Shadow	PCG	Sun	Direction	Elevation	Cloud	$t_{AEB}$ [ms]	$d_{AEB}$ [m]	TTC <sub>AEB</sub> [s]
Unity (Baseline)	True	False	False			90	False	3425	28.528824	1.380000
UE5 (Baseline)	True	False	False			90	False	3400	29.014679	1.400000
UE5	True	False	False			90	True	3375	29.400280	1.420000
UE5	True	False	True			90	False	3400	29.026175	1.400000
UE5	True	False	True			90	True	3425	28.484865	1.380000
UE5	True	True	False			90	False	3400	28.940065	1.400000
UE5	True	True	False			90	True	3375	29.363411	1.420000
UE5	True	True	False	North		10	False	3400	28.667345	1.380000
UE5	True	True	False	North		10	True	3425	28.551468	1.360000
UE5	True	True	False	North		45	False	3425	28.534414	1.380000
UE5	True	True	False	North		45	True	3400	28.867418	1.400000
UE5	True	True	False	East		10	False	3400	28.794138	1.380000
UE5	True	True	False	East		10	True	3400	28.880877	1.400000
UE5	True	True	False	East		45	False	3400	28.845625	1.400000
UE5	True	True	False	East		45	True	3400	28.890837	1.400000
UE5	True	True	False	South		10	False	3450	28.198927	1.360000
UE5	True	True	False	South		10	True	3350	29.967615	1.440000
UE5	True	True	False	South		45	False	3425	28.514536	1.380000
UE5	True	True	False	South		45	True	3425	28.566452	1.380000
UE5	True	True	False	West		10	False	3425	28.466536	1.380000
UE5	True	True	False	West		10	True	3425	28.713867	1.380000
UE5	True	True	False	West		45	False	3425	28.463831	1.380000
UE5	True	True	False	West		45	True	3400	28.865051	1.400000
UE5	True	True	True			90	False	3400	28.932596	1.400000
UE5	True	True	True			90	True	3400	28.913225	1.400000
UE5	True	True	True	North		10	False	3425	28.556427	1.380000
UE5	True	True	True	North		10	True	3425	28.469187	1.380000
UE5	True	True	True	North		45	False	3400	28.859922	1.400000
UE5	True	True	True	North		45	True	3400	28.807318	1.400000
UE5	True	True	True	East		10	False	3450	28.049444	1.360000
UE5	True	True	True	East		10	True	3425	28.416534	1.360000
UE5	True	True	True	East		45	False	3425	28.632727	1.380000
UE5	True	True	True	East		45	True	3425	28.590738	1.380000
UE5	True	True	True	South		10	False	3425	28.476370	1.380000
UE5	True	True	True	South		10	True	3425	28.493017	1.380000
UE5	True	True	True	South		45	False	3425	28.582336	1.380000
UE5	True	True	True	South		45	True	3425	28.494467	1.380000
UE5	True	True	True	West		10	False	3425	28.419558	1.380000
UE5	True	True	True	West		10	True	3425	28.414314	1.360000
UE5	True	True	True	West		45	False	3425	28.393076	1.360000
UE5	True	True	True	West		45	True	3425	28.493032	1.380000

## A. Comprehensive Results and AEB Metrics

---

Table A.7: AEB metrics for scenario: Car to Car Rear stationary (ccrs\_speed\_50)

Source	aeb_activated	Shadow	PCG	Sun	Direction	Elevation	Cloud	$t_{AEB}$ [ms]	$d_{AEB}$ [m]	TTC <sub>AEB</sub> [s]
Unity (Baseline)	True	False	False			90	False	13500	23.266449	1.260000
UE5 (Baseline)	True	False	False			90	False	13575	23.459267	1.280000
UE5	True	False	False			90	True	13475	23.651377	1.280000
UE5	True	False	True			90	False	13550	23.777697	1.300000
UE5	True	False	True			90	True	13475	23.548458	1.280000
UE5	True	True	False			90	False	13575	23.565609	1.280000
UE5	True	True	False			90	True	13475	23.666147	1.300000
UE5	True	True	False	North		10	False	13400	23.957798	1.320000
UE5	True	True	False	North		10	True	13675	23.614258	1.280000
UE5	True	True	False	North		45	False	13525	23.664560	1.280000
UE5	True	True	False	North		45	True	13450	23.886814	1.300000
UE5	True	True	False	East		10	False	13475	23.743219	1.300000
UE5	True	True	False	East		10	True	13475	23.589680	1.280000
UE5	True	True	False	East		45	False	13475	23.672819	1.300000
UE5	True	True	False	East		45	True	13575	23.522501	1.280000
UE5	True	True	False	South		10	False	13475	23.678093	1.280000
UE5	True	True	False	South		10	True	13700	23.312923	1.240000
UE5	True	True	False	South		45	False	13550	23.760593	1.300000
UE5	True	True	False	South		45	True	13550	23.816288	1.300000
UE5	True	True	False	West		10	False	13475	23.675817	1.280000
UE5	True	True	False	West		10	True	13450	24.161118	1.320000
UE5	True	True	False	West		45	False	13575	23.426079	1.280000
UE5	True	True	False	West		45	True	13575	23.506104	1.280000
UE5	True	True	True			90	False	13450	23.911760	1.300000
UE5	True	True	True			90	True	13550	23.737492	1.300000
UE5	True	True	True	North		10	False	13475	23.626026	1.280000
UE5	True	True	True	North		10	True	13575	23.436213	1.280000
UE5	True	True	True	North		45	False	13450	23.875713	1.300000
UE5	True	True	True	North		45	True	13450	23.851122	1.300000
UE5	True	True	True	East		10	False	13575	23.638645	1.280000
UE5	True	True	True	East		10	True	13450	24.015095	1.320000
UE5	True	True	True	East		45	False	13450	23.997772	1.320000
UE5	True	True	True	East		45	True	13450	23.889475	1.300000
UE5	True	True	True	South		10	False	13475	23.554743	1.280000
UE5	True	True	True	South		10	True	13500	23.338823	1.260000
UE5	True	True	True	South		45	False	13525	23.679308	1.300000
UE5	True	True	True	South		45	True	13575	23.538130	1.280000
UE5	True	True	True	West		10	False	13450	24.043760	1.320000
UE5	True	True	True	West		10	True	13475	23.858736	1.300000
UE5	True	True	True	West		45	False	13475	23.550318	1.280000
UE5	True	True	True	West		45	True	13475	23.639668	1.280000

Table A.8: AEB metrics for scenario: Car to Car Rear stationary (ccrs\_speed\_80)

Source	aeb_activated	Shadow	PCG	Sun	Direction	Elevation	Cloud	$t_{AEB}$ [ms]	$d_{AEB}$ [m]	TTC <sub>AEB</sub> [s]
Unity (Baseline)	True	False	False			90	False	13450	36.470341	1.380000
UE5 (Baseline)	True	False	False			90	False	13425	37.045891	1.420000
UE5	True	False	False			90	True	13425	37.045418	1.420000
UE5	True	False	True			90	False	13400	37.478279	1.440000
UE5	True	False	True			90	True	13425	37.041500	1.420000
UE5	True	True	False			90	False	13425	37.050018	1.420000
UE5	True	True	False			90	True	13425	37.043541	1.420000
UE5	True	True	False	North		10	False	14075	24.878700	0.860000
UE5	True	True	False	North		10	True	14150	23.369761	0.800000
UE5	True	True	False	North		45	False	13425	37.086670	1.420000
UE5	True	True	False	North		45	True	13400	37.680340	1.440000
UE5	True	True	False	East		10	False	13450	36.472614	1.380000
UE5	True	True	False	East		10	True	13450	36.317150	1.380000
UE5	True	True	False	East		45	False	13425	36.993423	1.400000
UE5	True	True	False	East		45	True	13400	37.579510	1.440000
UE5	True	True	False	South		10	False	13425	37.089359	1.420000
UE5	True	True	False	South		10	True	13750	29.658209	1.080000
UE5	True	True	False	South		45	False	13425	37.078281	1.420000
UE5	True	True	False	South		45	True	13425	37.033607	1.420000
UE5	True	True	False	West		10	False	13425	37.007584	1.400000
UE5	True	True	False	West		10	True	13400	37.792343	1.440000
UE5	True	True	False	West		45	False	13400	37.540424	1.440000
UE5	True	True	False	West		45	True	13400	37.564384	1.440000
UE5	True	True	True			90	False	13425	37.020130	1.420000
UE5	True	True	True			90	True	13425	37.049717	1.420000
UE5	True	True	True	North		10	False	14075	23.012762	0.780000
UE5	True	True	True	North		10	True	14150	23.378338	0.800000
UE5	True	True	True	North		45	False	13400	37.579559	1.440000
UE5	True	True	True	North		45	True	13425	37.046478	1.420000
UE5	True	True	True	East		10	False	14150	24.320620	0.840000
UE5	True	True	True	East		10	True	13825	27.737616	1
UE5	True	True	True	East		45	False	13400	37.433487	1.420000
UE5	True	True	True	East		45	True	13425	36.596107	1.400000
UE5	True	True	True	South		10	False	13425	37.071499	1.420000
UE5	True	True	True	South		10	True	13550	32.491314	1.200000
UE5	True	True	True	South		45	False	13425	37.038853	1.420000
UE5	True	True	True	South		45	True	13425	37.030640	1.420000
UE5	True	True	True	West		10	False	13950	25.976574	0.920000
UE5	True	True	True	West		10	True	13475	35.599724	1.340000
UE5	True	True	True	West		45	False	13400	37.351711	1.420000
UE5	True	True	True	West		45	True	13400	37.359467	1.420000

## A. Comprehensive Results and AEB Metrics

---

Table A.9: AEB metrics for scenario: Car to Pedestrian Farside Adult (cpfa\_speed\_50)

Source	aeb_activated	Shadow	PCG	Sun Direction	Elevation	Cloud	$t_{AEB}$ [ms]	$d_{AEB}$ [m]	TTC <sub>AEB</sub> [s]
Unity (Baseline)	True	False	False		90	False	7850	20.132771	1.160000
UE5 (Baseline)	True	False	False		90	False	7775	21.170599	1.240000
UE5	True	False	False		90	True	7775	21.170397	1.240000
UE5	True	False	True		90	False	7775	21.181952	1.240000
UE5	True	False	True		90	True	7775	21.183867	1.240000
UE5	True	True	False		90	False	7775	21.170731	1.240000
UE5	True	True	False		90	True	7775	21.170084	1.240000
UE5	True	True	False	North	10	False	7775	21.174850	1.240000
UE5	True	True	False	North	10	True	7775	21.168756	1.240000
UE5	True	True	False	North	45	False	7775	21.167816	1.240000
UE5	True	True	False	North	45	True	7775	21.161400	1.240000
UE5	True	True	False	East	10	False	7825	20.495630	1.180000
UE5	True	True	False	East	10	True	7775	21.160833	1.240000
UE5	True	True	False	East	45	False	7775	21.161856	1.240000
UE5	True	True	False	East	45	True	7775	21.164366	1.240000
UE5	True	True	False	South	10	False	7775	21.164310	1.240000
UE5	True	True	False	South	10	True	7775	21.165997	1.240000
UE5	True	True	False	South	45	False	7775	21.172081	1.240000
UE5	True	True	False	South	45	True	7775	21.172581	1.240000
UE5	True	True	False	West	10	False	7875	19.811607	1.140000
UE5	True	True	False	West	10	True	7775	21.164022	1.240000
UE5	True	True	False	West	45	False	7775	21.165865	1.240000
UE5	True	True	False	West	45	True	7775	21.168169	1.240000
UE5	True	True	True		90	False	7775	21.186045	1.240000
UE5	True	True	True		90	True	7750	21.531912	1.260000
UE5	True	True	True	North	10	False	7825	20.496010	1.180000
UE5	True	True	True	North	10	True	7825	20.482037	1.180000
UE5	True	True	True	North	45	False	7775	21.184998	1.240000
UE5	True	True	True	North	45	True	7775	21.184132	1.240000
UE5	True	True	True	East	10	False	7875	19.809765	1.140000
UE5	True	True	True	East	10	True	7925	19.088921	1.080000
UE5	True	True	True	East	45	False	7775	21.176905	1.240000
UE5	True	True	True	East	45	True	7775	21.176933	1.240000
UE5	True	True	True	South	10	False	7775	21.192623	1.240000
UE5	True	True	True	South	10	True	7775	21.197199	1.240000
UE5	True	True	True	South	45	False	7775	21.188972	1.240000
UE5	True	True	True	South	45	True	7750	21.530403	1.260000
UE5	True	True	True	West	10	False	7875	19.807159	1.140000
UE5	True	True	True	West	10	True	7775	21.208843	1.240000
UE5	True	True	True	West	45	False	7975	18.395506	1.040000
UE5	True	True	True	West	45	True	7800	20.820536	1.220000

Table A.10: AEB metrics for scenario: Car to Pedestrian Farside Adult (cpfa\_speed\_60)

Source	aeb_activated	Shadow	PCG	Sun Direction	Elevation	Cloud	$t_{AEB}$ [ms]	$d_{AEB}$ [m]	TTC <sub>AEB</sub> [s]
Unity (Baseline)	True	False	False		90	False	7850	23.251616	1.160000
UE5 (Baseline)	True	False	False		90	False	7775	24.513674	1.240000
UE5	True	False	False		90	True	7775	24.514751	1.240000
UE5	True	False	True		90	False	7775	24.524948	1.240000
UE5	True	False	True		90	True	7775	24.525517	1.240000
UE5	True	True	False		90	False	7775	24.512854	1.240000
UE5	True	True	False		90	True	7775	24.514141	1.240000
UE5	True	True	False	North	10	False	7775	24.512440	1.240000
UE5	True	True	False	North	10	True	7775	24.501425	1.240000
UE5	True	True	False	North	45	False	7775	24.505507	1.240000
UE5	True	True	False	North	45	True	7775	24.495602	1.220000
UE5	True	True	False	East	10	False	7950	21.604086	1.060000
UE5	True	True	False	East	10	True	7775	24.502693	1.240000
UE5	True	True	False	East	45	False	7775	24.504587	1.240000
UE5	True	True	False	East	45	True	7775	24.508856	1.240000
UE5	True	True	False	South	10	False	7775	24.504879	1.220000
UE5	True	True	False	South	10	True	7775	24.499088	1.220000
UE5	True	True	False	South	45	False	7775	24.512865	1.240000
UE5	True	True	False	South	45	True	7775	24.513147	1.240000
UE5	True	True	False	West	10	False	7975	21.206884	1.040000
UE5	True	True	False	West	10	True	7775	24.503752	1.240000
UE5	True	True	False	West	45	False	7775	24.501045	1.220000
UE5	True	True	False	West	45	True	7775	24.508175	1.240000
UE5	True	True	True		90	False	7775	24.528086	1.240000
UE5	True	True	True		90	True	7775	24.528774	1.240000
UE5	True	True	True	North	10	False	7775	24.515644	1.240000
UE5	True	True	True	North	10	True	7825	23.674494	1.180000
UE5	True	True	True	North	45	False	7800	24.092155	1.200000
UE5	True	True	True	North	45	True	7800	24.090969	1.200000
UE5	True	True	True	East	10	False	7975	21.198139	1.040000
UE5	True	True	True	East	10	True	8025	20.356506	0.980000
UE5	True	True	True	East	45	False	7775	24.515965	1.240000
UE5	True	True	True	East	45	True	7775	24.521177	1.240000
UE5	True	True	True	South	10	False	7775	24.526501	1.240000
UE5	True	True	True	South	10	True	7775	24.511652	1.240000
UE5	True	True	True	South	45	False	7775	24.525038	1.240000
UE5	True	True	True	South	45	True	7775	24.526005	1.240000
UE5	True	True	True	West	10	False	7925	22.017294	1.080000
UE5	True	True	True	West	10	True	7975	21.214239	1.040000
UE5	True	True	True	West	45	False	7850	23.243866	1.160000
UE5	True	True	True	West	45	True	7825	23.671610	1.180000

## A. Comprehensive Results and AEB Metrics

---

Table A.11: AEB metrics for scenario: Car to Pedestrian Longitudinal Adult (cpla\_speed\_50)

Source	aeb_activated	Shadow	PCG	Sun Direction	Elevation	Cloud	$t_{AEB}$ [ms]	$d_{AEB}$ [m]	TTC <sub>AEB</sub> [s]
Unity (Baseline)	True	False	False		90	False	5225	18.670464	1.220000
UE5 (Baseline)	True	False	False		90	False	5225	18.670464	1.220000
UE5	False	False	False		90	True			
UE5	True	False	True		90	False	5225	18.648870	1.220000
UE5	True	False	True		90	True	5200	18.691282	1.220000
UE5	False	True	False		90	False			
UE5	True	True	False		90	True	5175	18.967342	1.240000
UE5	True	True	False	North	10	False	5225	18.650328	1.220000
UE5	False	True	False	North	10	True			
UE5	True	True	False	North	45	False	5175	18.922821	1.240000
UE5	False	True	False	North	45	True			
UE5	True	True	False	East	10	False	5175	18.918383	1.240000
UE5	True	True	False	East	10	True	5175	18.921810	1.240000
UE5	False	True	False	East	45	False			
UE5	True	True	False	East	45	True	5175	18.965017	1.240000
UE5	True	True	False	South	10	False	5175	18.939972	1.240000
UE5	True	True	False	South	10	True	5175	18.893826	1.240000
UE5	False	True	False	South	45	False			
UE5	True	True	False	South	45	True	5200	18.896479	1.240000
UE5	True	True	False	West	10	False	5200	18.908611	1.240000
UE5	True	True	False	West	10	True	5175	18.977615	1.240000
UE5	True	True	False	West	45	False	5200	18.635159	1.220000
UE5	True	True	False	West	45	True	5200	18.787025	1.240000
UE5	False	True	True		90	False			
UE5	False	True	True		90	True			
UE5	True	True	True	North	10	False	5300	18.346264	1.180000
UE5	True	True	True	North	10	True	5175	18.971188	1.240000
UE5	True	True	True	North	45	False	5200	18.698898	1.220000
UE5	True	True	True	North	45	True	5175	18.935560	1.240000
UE5	True	True	True	East	10	False	5275	18.506298	1.200000
UE5	True	True	True	East	10	True	5225	18.670464	1.220000
UE5	True	True	True	East	45	False	5200	18.931341	1.240000
UE5	True	True	True	East	45	True	5200	18.939804	1.240000
UE5	True	True	True	South	10	False	5175	18.913832	1.240000
UE5	True	True	True	South	10	True	5175	18.925900	1.240000
UE5	True	True	True	South	45	False	5175	18.969923	1.240000
UE5	True	True	True	South	45	True	5175	18.939837	1.240000
UE5	True	True	True	West	10	False	5300	17.838709	1.160000
UE5	True	True	True	West	10	True	5175	19.054850	1.240000
UE5	True	True	True	West	45	False	5200	18.799520	1.240000
UE5	True	True	True	West	45	True	5225	18.665379	1.220000

Table A.12: AEB metrics for scenario: Car to Pedestrian Longitudinal Adult (cpla\_speed\_80)

Source	aeb_activated	Shadow	PCG	Sun Direction	Elevation	Cloud	$t_{AEB}$ [ms]	$d_{AEB}$ [m]	TTC <sub>AEB</sub> [s]
Unity (Baseline)	True	False	False		90	False	4850	32.261547	1.360000
UE5 (Baseline)	True	False	False		90	False	4825	32.791924	1.380000
UE5	True	False	False		90	True	4850	32.236458	1.360000
UE5	True	False	True		90	False	4850	32.436893	1.360000
UE5	True	False	True		90	True	4825	32.792137	1.380000
UE5	True	True	False		90	False	4850	32.266579	1.360000
UE5	True	True	False		90	True	4850	32.236504	1.360000
UE5	True	True	False	North	10	False	4825	32.810741	1.380000
UE5	True	True	False	North	10	True	4875	31.880730	1.340000
UE5	True	True	False	North	45	False	4850	32.253445	1.360000
UE5	True	True	False	North	45	True	4875	31.717182	1.320000
UE5	True	True	False	East	10	False	4825	32.679329	1.380000
UE5	True	True	False	East	10	True	4850	32.448093	1.360000
UE5	True	True	False	East	45	False	4825	32.787041	1.380000
UE5	True	True	False	East	45	True	4825	32.788834	1.380000
UE5	True	True	False	South	10	False	4875	31.707197	1.320000
UE5	True	True	False	South	10	True	4850	32.123440	1.360000
UE5	True	True	False	South	45	False	4875	31.889767	1.340000
UE5	True	True	False	South	45	True	4900	31.369726	1.320000
UE5	True	True	False	West	10	False	4775	33.686771	1.420000
UE5	True	True	False	West	10	True	4825	32.794167	1.380000
UE5	True	True	False	West	45	False	4875	31.747435	1.340000
UE5	True	True	False	West	45	True	4850	32.437420	1.360000
UE5	True	True	True		90	False	4825	32.751667	1.380000
UE5	True	True	True		90	True	4850	32.270611	1.360000
UE5	True	True	True	North	10	False	5025	28.488304	1.180000
UE5	True	True	True	North	10	True	4850	32.133858	1.360000
UE5	True	True	True	North	45	False	4850	32.260624	1.360000
UE5	True	True	True	North	45	True	4850	32.255733	1.360000
UE5	True	True	True	East	10	False	4825	33.052925	1.360000
UE5	True	True	True	East	10	True	4825	32.705723	1.380000
UE5	True	True	True	East	45	False	4900	31.796093	1.360000
UE5	True	True	True	East	45	True	4850	32.230152	1.360000
UE5	True	True	True	South	10	False	4875	31.876381	1.340000
UE5	True	True	True	South	10	True	4850	32.121925	1.360000
UE5	True	True	True	South	45	False	4850	32.170673	1.360000
UE5	True	True	True	South	45	True	4875	31.671534	1.320000
UE5	True	True	True	West	10	False	4875	32.226093	1.340000
UE5	True	True	True	West	10	True	4825	32.657642	1.380000
UE5	True	True	True	West	45	False	4875	31.746912	1.340000
UE5	True	True	True	West	45	True	4950	30.833418	1.320000

## A. Comprehensive Results and AEB Metrics

---

Table A.13: AEB metrics for scenario: Car to Pedestrian Nearside Adult (cpna\_speed\_50)

Source	aeb_activated	Shadow	PCG	Sun Direction	Elevation	Cloud	$t_{AEB}$ [ms]	$d_{AEB}$ [m]	TTC <sub>AEB</sub> [s]
Unity (Baseline)	True	False	False		90	False	7825	20.433614	1.180000
UE5 (Baseline)	True	False	False		90	False	7825	20.430323	1.180000
UE5	True	False	False		90	True	7825	20.434738	1.180000
UE5	True	False	True		90	False	7825	20.437914	1.180000
UE5	True	False	True		90	True	7825	20.441677	1.180000
UE5	True	True	False		90	False	7825	20.428402	1.180000
UE5	True	True	False		90	True	7825	20.435299	1.180000
UE5	True	True	False	North	10	False	7825	20.417156	1.180000
UE5	True	True	False	North	10	True	7825	20.423988	1.180000
UE5	True	True	False	North	45	False	7825	20.427738	1.180000
UE5	True	True	False	North	45	True	7825	20.421307	1.180000
UE5	True	True	False	East	10	False	7825	20.420332	1.180000
UE5	True	True	False	East	10	True	7825	20.416868	1.180000
UE5	True	True	False	East	45	False	7825	20.422045	1.180000
UE5	True	True	False	East	45	True	7825	20.427790	1.180000
UE5	True	True	False	South	10	False	7825	20.425186	1.180000
UE5	True	True	False	South	10	True	7850	20.029562	1.160000
UE5	True	True	False	South	45	False	7825	20.427444	1.180000
UE5	True	True	False	South	45	True	7825	20.434334	1.180000
UE5	True	True	False	West	10	False	7825	20.399633	1.180000
UE5	True	True	False	West	10	True	7825	20.436758	1.180000
UE5	True	True	False	West	45	False	7825	20.421583	1.180000
UE5	True	True	False	West	45	True	7825	20.428967	1.180000
UE5	True	True	True		90	False	7775	21.142591	1.240000
UE5	True	True	True		90	True	7775	21.145655	1.240000
UE5	True	True	True	North	10	False	7775	21.128147	1.240000
UE5	True	True	True	North	10	True	7825	20.423986	1.180000
UE5	True	True	True	North	45	False	7825	20.439974	1.180000
UE5	True	True	True	North	45	True	7775	21.140913	1.240000
UE5	True	True	True	East	10	False	7775	21.130575	1.240000
UE5	True	True	True	East	10	True	7975	18.388239	1.040000
UE5	True	True	True	East	45	False	7825	20.437714	1.180000
UE5	True	True	True	East	45	True	7825	20.455101	1.180000
UE5	True	True	True	South	10	False	7825	20.434692	1.180000
UE5	True	True	True	South	10	True	7775	21.120731	1.240000
UE5	True	True	True	South	45	False	7775	21.138309	1.240000
UE5	True	True	True	South	45	True	7800	20.780373	1.200000
UE5	True	True	True	West	10	False	7825	20.470520	1.180000
UE5	True	True	True	West	10	True	7975	18.422060	1.040000
UE5	True	True	True	West	45	False	7775	21.139494	1.240000
UE5	True	True	True	West	45	True	7775	21.127869	1.240000

Table A.14: AEB metrics for scenario: Car to Pedestrian Nearside Adult (cpna\_speed\_60)

Source	aeb_activated	Shadow	PCG	Sun Direction	Elevation	Cloud	$t_{AEB}$ [ms]	$d_{AEB}$ [m]	$TTC_{AEB}$ [s]
Unity (Baseline)	True	False	False		90	False	7900	22.432718	1.100000
UE5 (Baseline)	True	False	False		90	False	7800	24.143545	1.200000
UE5	True	False	False		90	True	7800	24.149820	1.200000
UE5	True	False	True		90	False	7775	24.598810	1.240000
UE5	True	False	True		90	True	7775	24.606062	1.240000
UE5	True	True	False		90	False	7825	23.731644	1.180000
UE5	True	True	False		90	True	7800	24.149328	1.200000
UE5	True	True	False	North	10	False	7775	24.555496	1.240000
UE5	True	True	False	North	10	True	7775	24.581764	1.240000
UE5	True	True	False	North	45	False	7825	23.734491	1.180000
UE5	True	True	False	North	45	True	7825	23.730953	1.180000
UE5	True	True	False	East	10	False	7725	25.428207	1.280000
UE5	True	True	False	East	10	True	7775	24.587732	1.240000
UE5	True	True	False	East	45	False	7775	24.565908	1.240000
UE5	True	True	False	East	45	True	7800	24.144627	1.200000
UE5	True	True	False	South	10	False	7825	23.719654	1.180000
UE5	True	True	False	South	10	True	7775	24.582682	1.240000
UE5	True	True	False	South	45	False	7800	24.142698	1.200000
UE5	True	True	False	South	45	True	7775	24.579433	1.240000
UE5	True	True	False	West	10	False	8025	20.318312	0.980000
UE5	True	True	False	West	10	True	7775	24.577637	1.240000
UE5	True	True	False	West	45	False	7800	24.142691	1.200000
UE5	True	True	False	West	45	True	7825	23.728725	1.180000
UE5	True	True	True		90	False	7800	24.175282	1.220000
UE5	True	True	True		90	True	7775	24.613287	1.240000
UE5	True	True	True	North	10	False	7775	24.594763	1.240000
UE5	True	True	True	North	10	True	7875	22.918505	1.140000
UE5	True	True	True	North	45	False	7775	24.584764	1.240000
UE5	True	True	True	North	45	True	7875	22.908951	1.140000
UE5	True	True	True	East	10	False	7875	22.952164	1.140000
UE5	True	True	True	East	10	True	8250	16.625271	0.760000
UE5	True	True	True	East	45	False	7775	24.597067	1.240000
UE5	True	True	True	East	45	True	7775	24.600988	1.240000
UE5	True	True	True	South	10	False	7775	24.600492	1.240000
UE5	True	True	True	South	10	True	7775	24.581869	1.240000
UE5	True	True	True	South	45	False	7775	24.595150	1.240000
UE5	True	True	True	South	45	True	7775	24.598782	1.240000
UE5	True	True	True	West	10	False	7775	24.604622	1.240000
UE5	True	True	True	West	10	True	8100	19.071049	0.900000
UE5	True	True	True	West	45	False	7775	24.590113	1.240000
UE5	True	True	True	West	45	True	7775	24.579037	1.240000

## A. Comprehensive Results and AEB Metrics

---

Table A.15: AEB metrics for scenario: Car to Pedestrian Nearside Child Obstructed (cpnco\_speed\_50)

Source	aeb_activated	Shadow	PCG	Sun Direction	Elevation	Cloud	$t_{AEB}$ [ms]	$d_{AEB}$ [m]	TTC <sub>AEB</sub> [s]
Unity (Baseline)	True	False	False		90	False	7975	18.414640	1.040000
UE5 (Baseline)	True	False	False		90	False	7900	19.495506	1.100000
UE5	False	False	False		90	True			
UE5	True	False	True		90	False	7875	19.768755	1.140000
UE5	True	False	True		90	True	7975	18.422293	1.040000
UE5	True	True	False		90	False	7975	18.452734	1.040000
UE5	True	True	False		90	True	7875	19.764744	1.120000
UE5	True	True	False	North	10	False	8350	14.008658	0.700000
UE5	True	True	False	North	10	True	8250	14.513721	0.760000
UE5	True	True	False	North	45	False	7975	18.433289	1.040000
UE5	True	True	False	North	45	True	7975	18.402502	1.040000
UE5	True	True	False	East	10	False	8250	14.509875	0.740000
UE5	True	True	False	East	10	True	8000	18.053556	1.020000
UE5	True	True	False	East	45	False	7825	20.431211	1.180000
UE5	True	True	False	East	45	True	7875	19.758642	1.140000
UE5	True	True	False	South	10	False	7900	19.492857	1.100000
UE5	True	True	False	South	10	True	7850	20.045416	1.160000
UE5	True	True	False	South	45	False	7975	18.474640	1.040000
UE5	True	True	False	South	45	True	8225	14.883404	0.780000
UE5	True	True	False	West	10	False	8675	9.551875	0.400000
UE5	True	True	False	West	10	True	8225	14.894769	0.780000
UE5	True	True	False	West	45	False	8050	17.344381	0.960000
UE5	True	True	False	West	45	True	8025	17.696676	0.980000
UE5	True	True	True		90	False	8150	15.877398	0.860000
UE5	True	True	True		90	True	8175	15.539079	0.820000
UE5	True	True	True	North	10	False	8175	15.560305	0.820000
UE5	True	True	True	North	10	True	8375	12.767088	0.620000
UE5	True	True	True	North	45	False	8175	15.546997	0.820000
UE5	True	True	True	North	45	True	8175	15.556678	0.820000
UE5	True	True	True	East	10	False	8225	14.879871	0.780000
UE5	True	True	True	East	10	True	8250	14.546817	0.760000
UE5	True	True	True	East	45	False	8175	15.537661	0.820000
UE5	True	True	True	East	45	True	8100	16.632729	0.900000
UE5	True	True	True	South	10	False	8225	14.885742	0.780000
UE5	True	True	True	South	10	True	8275	14.553331	0.760000
UE5	True	True	True	South	45	False	8100	16.651987	0.900000
UE5	True	True	True	South	45	True	8100	16.635059	0.900000
UE5	True	True	True	West	10	False	8275	15.437057	0.800000
UE5	True	True	True	West	10	True	8425	12.138608	0.580000
UE5	True	True	True	West	45	False	8325	13.485094	0.680000
UE5	True	True	True	West	45	True	8000	18.004488	1

Table A.16: AEB metrics for scenario: Car to Pedestrian Nearside Child Obstructed (cpnco\_speed\_60)

Source	aeb_activated	Shadow	PCG	Sun Direction	Elevation	Cloud	$t_{AEB}$ [ms]	$d_{AEB}$ [m]	TTC <sub>AEB</sub> [s]
Unity (Baseline)	True	False	False		90	False	7975	21.234196	1.020000
UE5 (Baseline)	True	False	False		90	False	8050	19.980902	0.960000
UE5	False	False	False		90	True			
UE5	True	False	True		90	False	7975	21.211634	1.040000
UE5	True	False	True		90	True	7975	21.401428	1.040000
UE5	True	True	False		90	False	8200	17.670412	0.820000
UE5	True	True	False		90	True	8050	20.081051	0.960000
UE5	True	True	False	North	10	False	8050	20.004333	0.960000
UE5	True	True	False	North	10	True	8150	18.375706	0.860000
UE5	True	True	False	North	45	False	8650	11.122294	0.420000
UE5	True	True	False	North	45	True	7900	22.530161	1.100000
UE5	True	True	False	East	10	False	8250	16.705299	0.760000
UE5	True	True	False	East	10	True	8150	18.376335	0.860000
UE5	True	True	False	East	45	False	7975	21.264961	1.040000
UE5	True	True	False	East	45	True	8200	17.635790	0.820000
UE5	True	True	False	South	10	False	8075	19.810110	0.940000
UE5	True	True	False	South	10	True	8150	18.767038	0.880000
UE5	True	True	False	South	45	False	7950	21.744860	1.060000
UE5	True	True	False	South	45	True	7975	21.242659	1.040000
UE5	False	True	False	West	10	False			
UE5	True	True	False	West	10	True	7975	21.195585	1.040000
UE5	False	True	False	West	45	False			
UE5	True	True	False	West	45	True	7975	21.251320	1.040000
UE5	True	True	True		90	False	8050	20.081270	0.960000
UE5	True	True	True		90	True	8075	19.655746	0.940000
UE5	True	True	True	North	10	False	8150	18.395746	0.860000
UE5	True	True	True	North	10	True	8250	16.710253	0.760000
UE5	True	True	True	North	45	False	8275	16.261009	0.740000
UE5	True	True	True	North	45	True	8050	20.016195	0.960000
UE5	True	True	True	East	10	False	8325	15.472323	0.680000
UE5	True	True	True	East	10	True	8425	13.835559	0.600000
UE5	True	True	True	East	45	False	7950	21.661390	1.060000
UE5	True	True	True	East	45	True	8050	20.047468	0.960000
UE5	True	True	True	South	10	False	8275	17.274683	0.780000
UE5	True	True	True	South	10	True	8000	20.875786	1.020000
UE5	True	True	True	South	45	False	8150	18.435328	0.860000
UE5	True	True	True	South	45	True	7875	22.920420	1.140000
UE5	True	True	True	West	10	False	8050	20.157181	0.960000
UE5	True	True	True	West	10	True	8375	14.902132	0.660000
UE5	True	True	True	West	45	False	8075	19.586861	0.940000
UE5	True	True	True	West	45	True	8125	18.726536	0.880000

## A. Comprehensive Results and AEB Metrics

---

# B

## Source Code Listings

This appendix contains the source code listings referenced in Chapter 3.

```

/* Tick Message Format */
#define TRANSFORM_NAME_MAX_SIZE 64
#define TRANSFORM_BATCH_COUNT    32

struct Vec3 { float x, y, z; };
struct Transform {
    char name[TRANSFORM_NAME_MAX_SIZE];
    Vec3 Position;           // metres (OpenSCENARIO world)
    Vec3 Orientation;        // roll, pitch, yaw in rad
    int ID;                  // OpenSCENARIO object ID
};

/* esmini -> UE5 */
struct TickMessage {
    uint64_t step;           // simulation step index (monotonic)
    int64_t time_ns;         // simulation time since start [ns]
    float deltaTime;         // t since previous frame [s]
    int count;               // number of valid transforms
    bool isLastMessage;      // multi-packet support
    Transform transformsBuffer[TRANSFORM_BATCH_COUNT];
};

/* UE5 -> esmini */
struct TockMessage {
    uint64_t step;           // echoed step index
    int64_t time_ns;         // echoed simulation time [ns]
};

/* UE5 <-> Zenseact */
struct Trigger { uint8_t value; }; // handshake signal
struct Image_Data{
    struct Header {
        int32_t image_size_bytes; // payload size
        int64_t time_ns;          // simulation time [ns]
        uint64_t step;            // step index
    } header;
    std::vector<uint8_t> image_data; // RGBA bytes
};

```

Listing 1: Message layout for inter-process communication.

```

bool EsminiRunner::Step()
{
    osi3::GroundTruth gt;
    CollectDynamicGroundTruth(gt);

    int start = 0;
    while(start < gt.moving_object_size()){
        TickMessage t{};
        start = CreateMessage(t, gt, start);
        WriteBlocked(tickPipe, &t, sizeof(t));
    }

    TockMessage tk{};
    ReadBlocked(tockPipe, &tk, sizeof(tk));
    logical_time_ = tk.time;

    return ++current_step_ < nr_of_steps_;
}

```

Listing 2: esmini endpoint

```

void UMyGameInstance::Init()
{
    Super::Init();

    OnPreTickHandle =
        FWorldDelegates::OnWorldTickStart.AddUObject(
            this, &UMyGameInstance::OnPreTick);
    OnPostTickHandle =
        FWorldDelegates::OnWorldPostActorTick.AddUObject(
            this, &UMyGameInstance::OnPostTick);

    tickPipe = open("/tmp/myfifo", O_RDONLY); // esmini -> UE5
    tockPipe = open("/tmp/myfifo2", O_WRONLY); // UE5 -> esmini
    check(tickPipe != -1 && tockPipe != -1);
}

```

Listing 3: Init

## B. Source Code Listings

---

```
void UMyGameInstance::OnPreTick(UWorld*, ELevelTick, float)
{
    TickMessage msg{};
    ReadBlocked(tickPipe, &msg, sizeof(msg));

    // Cache for echo in PostTick
    LastTickStep = msg.step;
    LastTickTimeNs = msg.time_ns;

    for (int i=0; i<msg.count; ++i) {
        const auto& T = msg.transformsBuffer[i];
        const FTransform UE5 =
            CoordTranslate::OdrToUnreal::ToTransform(
                T.Position, T.Orientation);
        UpdateOrSpawnActor(T.ID, FString(T.name), UE5);
    }
}
```

Listing 4: Pre tick

```
void UMyGameInstance::OnPostTick(UWorld*, ELevelTick, float)
{
    CaptureAndLog();

    TockMessage tm{ LastTickStep, LastTickTimeNs };
    WriteBlocked(tockPipe, &tm, sizeof(tm));
}
```

Listing 5: Post tick

```

void FRoadMeshBuilder::BuildMesh(UWorld* W,
                                  const odr::Mesh3D& M,
                                  FString Name,
                                  FString Mat,
                                  float Zlift)
{
    TArray< FVector> UEverts; UEverts.Reserve(M.vertices.size());
    for(const auto& v : M.vertices){
        FVector P = CoordTranslate::OdrToUnreal::ToLocation(v);
        P.Z += Zlift;
        UEverts.Add(P);
    }

    UProceduralMeshComponent* PM =
        NewObject<UProceduralMeshComponent>(W->GetWorldSettings());
    PM->RegisterComponent();
    PM->CreateMeshSection(0, UEverts,
                          ToUE(M.indices),
                          UpNormals(UEverts),
                          ToUV(M.st_coordinates),
                          {}, {}, true);
    PM->SetMaterial(0, LoadObject<UMaterialInterface>(nullptr, *Mat));
}

```

Listing 6: Road-mesh builder

```

PCGComponent->GetGraphInstance()
    ->SetGraphParameter<float>("NewProperty",
                                0.01f); // density
PCGComponent->GetGraphInstance()
    ->SetGraphParameter<float>("NewProperty_1",
                                Seed); // seed

```

Listing 7: Procedural generation

```

EgoCam->InitializeCamera(ConvertedPos,
                           ConvertedRot,
                           DestSize, Crop, Center,
                           EEgoCarImageLog::PNGLogger);

```

Listing 8: Camera setup

```
/* AEgoCarCameraActor::CaptureAndLog() */
SceneCaptureComp->CaptureScene();
FlushRenderingCommands();                                // wait for GPU
TArray<FColor> Raw;
UKismetRenderingLibrary::ReadRenderTarget(this,
                                         RenderTarget,
                                         Raw,
                                         false);
ProcessCapturedImage(Raw, Crop);
ZmqSocket->send(Crop.GetData(), Crop.Num()*4);
ZmqSocket->recv();                                     // wait
```

Listing 9: Virtual image capture