NVIDIA

DRIVE SIM ON OMNIVERSE SYNTHETIC DATA GENERATION

GTC Session A31491 | Matt Cragun | Nov 2021

USING SYNTHETIC DATA FOR AV DEVELOPMENT - BENEFITS

Ground Truth Human Can't Label
Occlusion | Velocity | Depth | Adverse Weather

Programmability
Full control over data diversity and distribution

Speed
Fast Iteration | Development can start from day one

Cost
Saves the cost of collecting and labeling real data

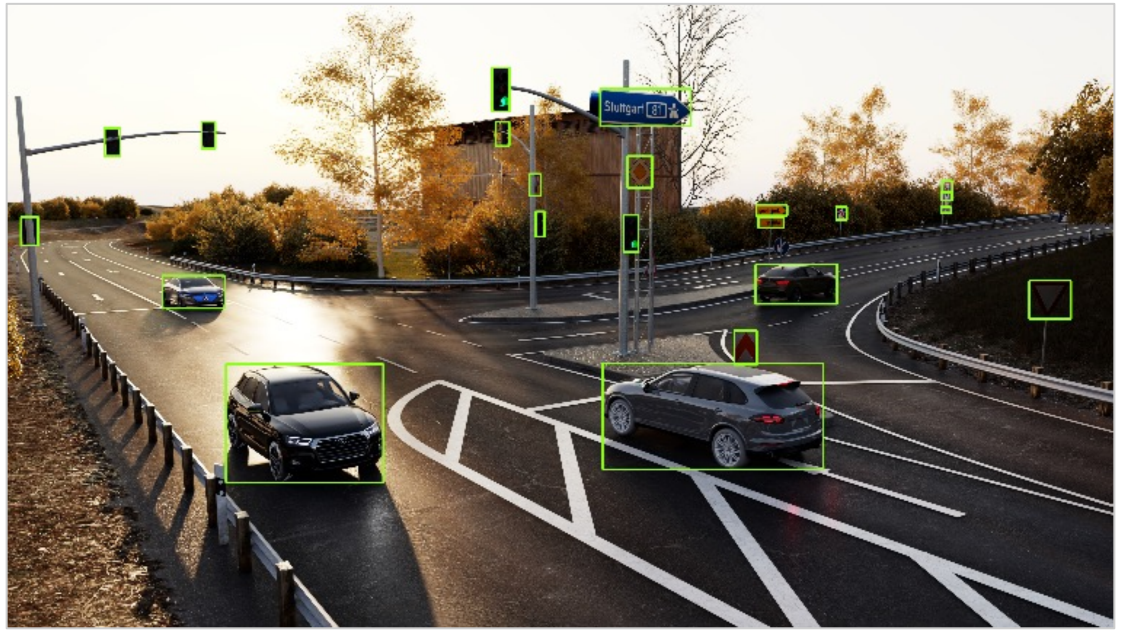
Accuracy
Pixel level accuracy | No human errors

Corner Cases
Addresses the long tail | Covers known unknowns

No Data Restrictions
Avoids privacy and regulation issues

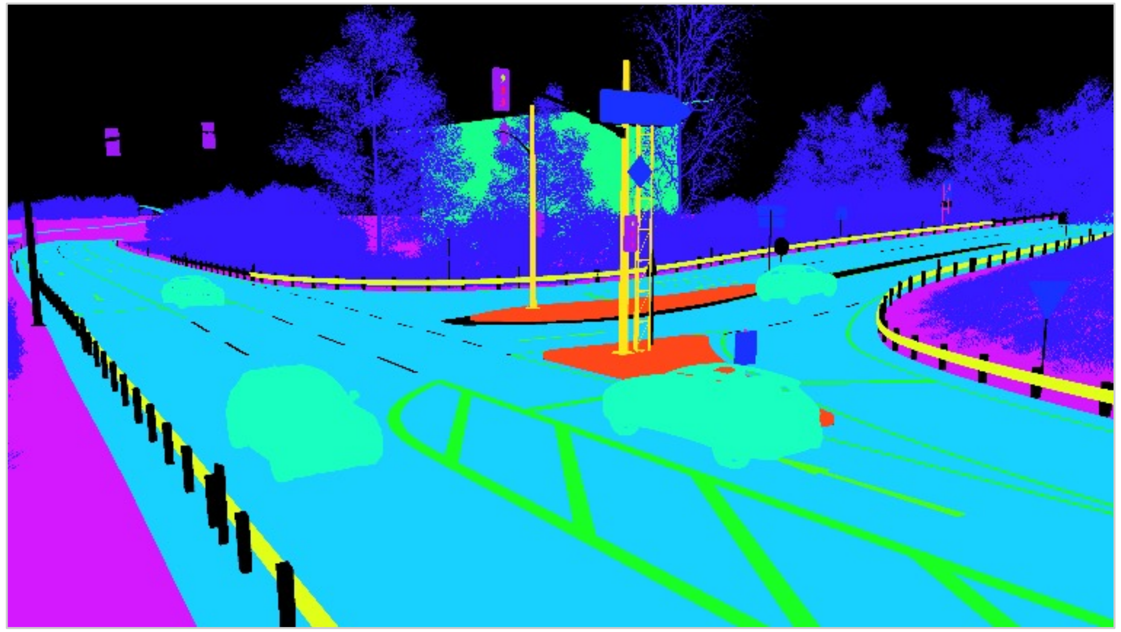
DRIVE SIM - GROUND TRUTH EXAMPLES

Photoreal, Physically-Based Image



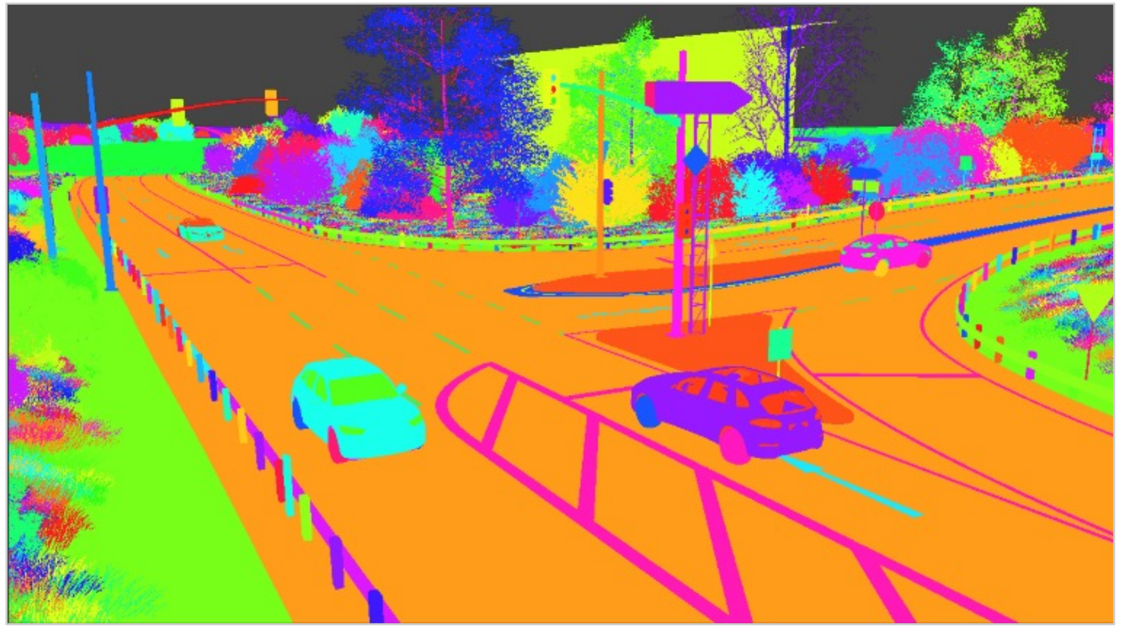
Automatic and Precise Labeling

Depth



Class Segmentation

3D Bounding Box



Instance Segmentation

SOME REAL-LIFE SCENES ARE VERY HARD TO LABEL

Example data from the NVIDIA AV Database

Dark | Blurry | Hazy

Occluded pedestrians and vehicles

Irregular scenes

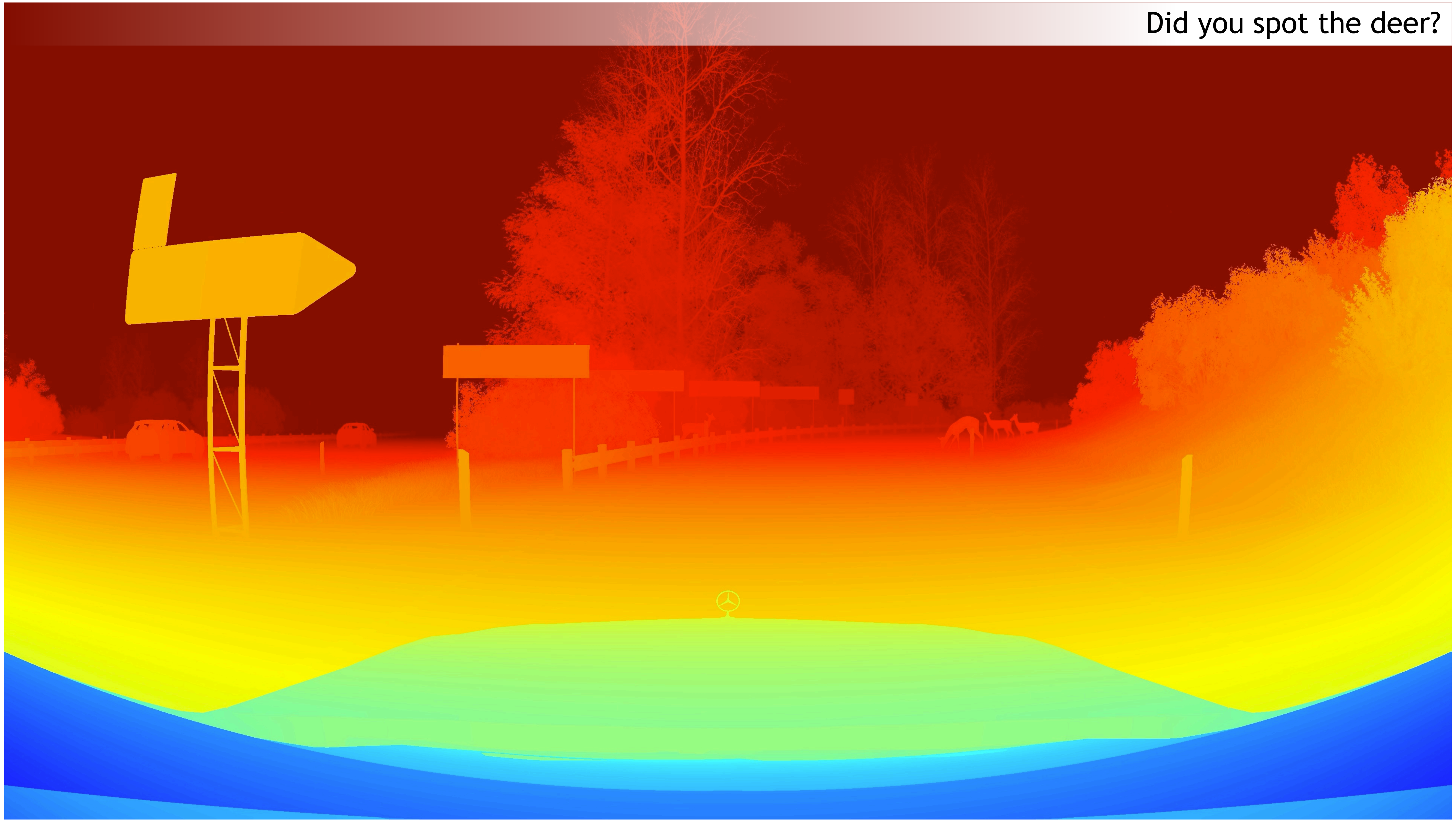
High quantity labels

SYNTHETIC DATA FOR RARE / DIFFICULT SCENES

Adverse Lighting / Animals

SYNTHETIC DATA FOR RARE / DIFFICULT SCENES

Adverse Lighting / Animals



SYNTHETIC DATA FOR RARE / DIFFICULT SCENES

Night / Emergency vehicles

SYNTHETIC DATA FOR RARE / DIFFICULT SCENES

Occluded Objects

DATA DRIVEN AI DEVELOPMENT

DATA DEVELOPMENT

Focus on good data

Design the distribution

MODEL DEVELOPMENT

Modify architecture

Test different models

ACCELERATING AV DEVELOPMENT WITH SDG AT NVIDIA

WE ARE USING IT – IT WORKS!

ACCELERATING AV DEVELOPMENT WITH SYNTHETIC DATA AT NVIDIA

PHASE I: BOOTSTRAP

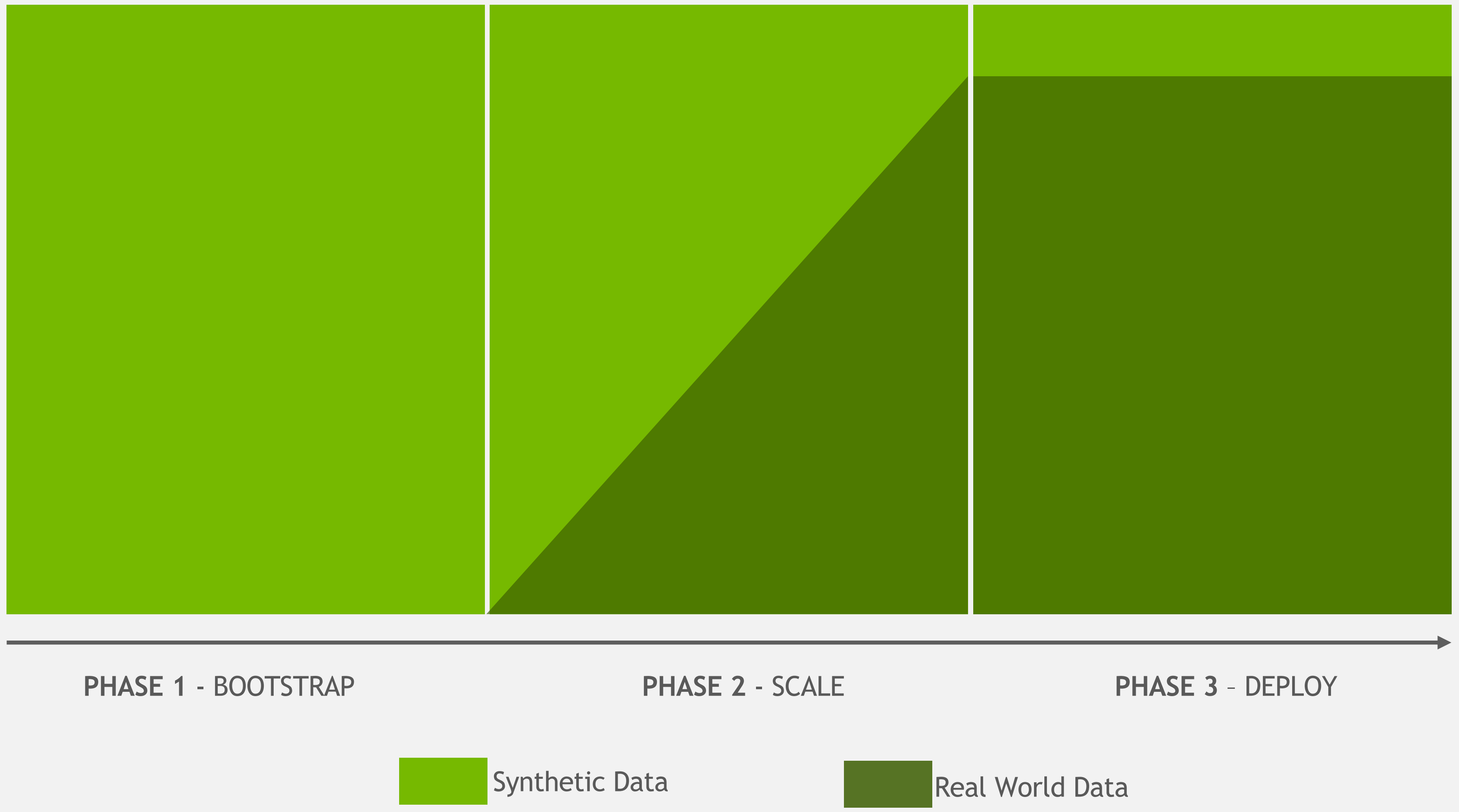
- Optimize for speed. Fail fast and early. Propagate successful ideas to real data.
- Evaluate DNNs and perception algorithms
- Evaluate sensor configs and placement.

PHASE II: SCALE

- Bring up labeling pipeline with synthetically trained DNNs.
- Augment real data with synthetic data. Optimize data distribution.
- Cover rare and dangerous cases with synthetic data – known unknowns.

PHASE III: DEPLOY

- Continuously identify unknown/unseen data gaps.
- Fill in gaps with synthetic data.



SDG Use: Next-Gen Sensor/DNN Bring-up

Goal: Migrate from Hyp7 to Hyp8 sensor set | Migrate to late fusion DNN architecture

Challenge: Get started before the physical sensors are installed on cars

Solution: Generated 2M images & GT before sensors were available. DNNs trained and bootstrapped, ready day 1, delivered good results in real.

Impact: Accelerated development by months

NVIDIA

SDG Use: NVIDIA PathNet

Goal:	PathNet determines the drivable lane. Improve lane detection when the vehicle was not centered on the lane.
Challenge:	Low count of real GT data because driving half out of the lane is unsafe for data collection
Solution:	Generated 2M images & GT Focus on offset, night, adverse sun, pavement/line type
Impact:	Improved Mean Average Distance to failure (MAD) 10+%

NVIDIA

SDG Use: NVIDIA LightNet

Goal:	LightNet detects stoplights and classifies the type of light displayed. Improve light detection.
Challenge:	Difficulty detecting lights at extreme angles. Right-Arrow-Red often misclassified.
Solution:	Generated dataset of lights in various poses and orientations Focus on variation in color, brightness, blur
Impact:	Improved F1-score 2.65%

NVIDIA

SDG Use: NVIDIA SignNet

Goal:	SignNet detects and classifies road signs. Improve detection of signs in the EU.
Challenge:	Subset of signs being misclassified due to lack of examples.
Solution:	Generated 60K images across 6 categories; Augmented real dataset with >400 categories
Impact:	Improved F1-score 2.03%

NVIDIA

WHY IS THIS A HARD PROBLEM ?

THE SIM-REAL DOMAIN GAP

THE SIM-REAL DOMAIN GAP PROBLEM

Appearance Gap

Pixel-level differences to real sensor output

Content Gap

Differences in diversity, context and behavior to real data

KITTI - REAL

VIRTUAL KITTI - SIM

WHAT DOES IT TAKE TO NARROW THE DOMAIN GAP?

Appearance Gap

Pixel-Level differences to real sensor output

Narrowing the Appearance Gap

High fidelity 3D assets

Physically based rendering and materials

Validated sensor models

Multi sensor support

Content Gap

Differences in diversity, context and behavior to real data

Narrowing the Content Gap

Access to a large pool of content

Highly automated import pipeline, built on open standards

Tech and tools to generate new and diverse scenes

Realistic and repeatable behavior models

Real to Sim pipeline | Replicate real scenes in Sim

Scalable platform | Cloud native

NARROWING THE APPEARANCE GAP
OMNIVERSE RTX RENDERER

OMNIVERSE RTX RENDERER

Two Rendering Modes: Path Traced | Realtime Path Traced

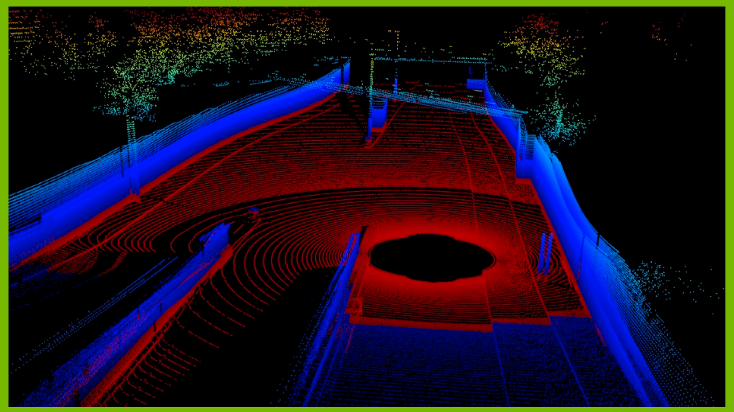
Scalable Across Multiple GPUs and Multiple GPU Nodes

Physically-Based Lighting and Materials

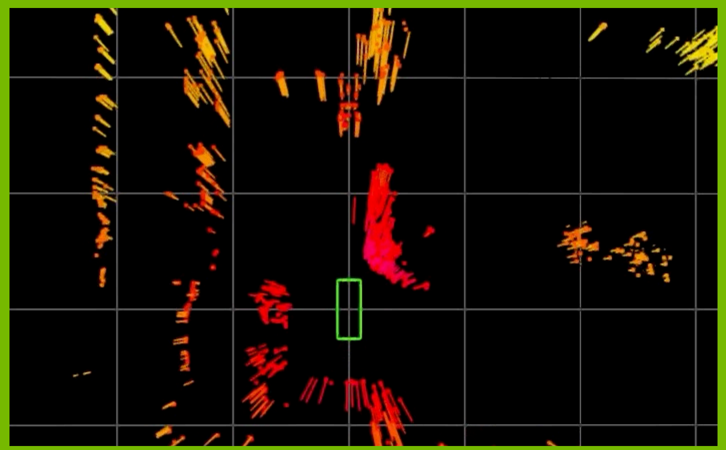
Time-Stamped World Model & Per-Ray Timestamps

Camera

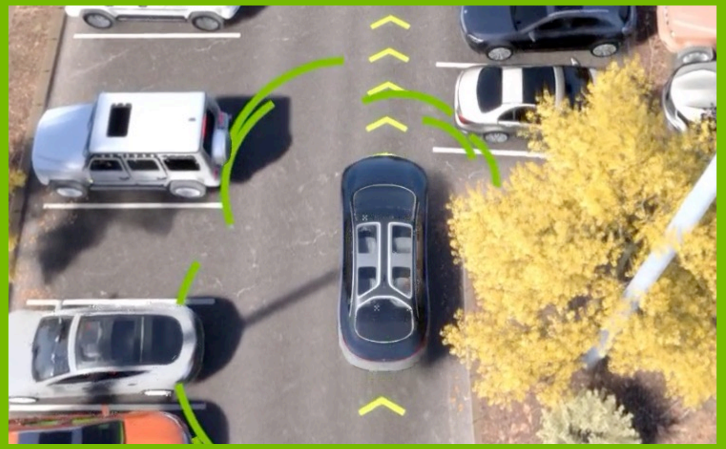
Lidar



Radar



Ultrasonics



RTX CAMERA MODEL

DRIVE SIM - RTX CAMERA MODEL

Lens Model: Generate rays from optical lens model; not post-warp on pinhole camera (f-theta, fisheye)

Dynamic Lighting: Variable times of day and weather conditions; High quality without "pre-baking"

Indirect Lighting: Crucial for night & tunnels; reflection of headlights

Motion Blur: Time-stamped world model & per-ray timestamps for inter-frame motion

Rolling Shutter: Time-stamped world model & per-ray timestamps for per-scanline motion

LED Flicker Mitigation: Time-stamped LED state & per-ray timestamps to model flicker

Physically-Based Lighting: Described in NITs, IES profiles. Validated & used for training & testing.

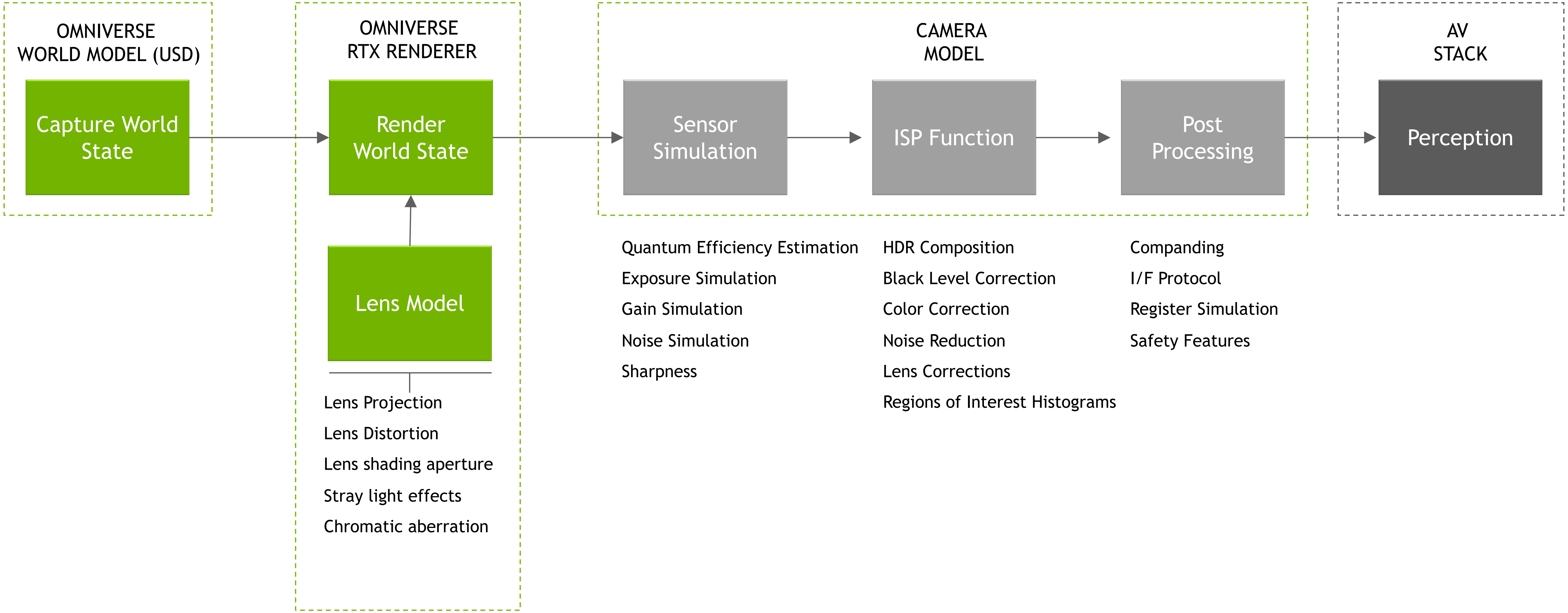
Physically-Based Materials: Material Definition Language (MDL) for measurable BRDF, refraction, scattering

DRIVE SIM CAMERA MODEL - EXAMPLES

MULTI-CAMERA VIEW

7 Camera vehicle with 3D cuboid ground truth

DRIVE SIM - CAMERA MODEL PIPELINE



USING CUSTOM SENSOR MODELS IN DRIVE SIM

PARTNER SENSOR MODELS IN DRIVE SIM

Preset Models

DRIVE Sim directly support many sensors from well-known manufacturers

Configure Models

Parameters of NVIDIA sensor models can be adjusted to meet specific sensor requirements

Custom Models

Custom models can be created by users for new sensor types

Partner Models

NVIDIA Ecosystem Partners build DRIVE Sim compatible models

PARTNER SENSOR MODELS SUPPORTED IN DRIVE SIM

DRIVE SIM CAMERA MODEL PIPELINE

3D Environment / World State	RTX Render + Lens Model	Sensor Model	ISP Model
			
DRIVE Sim updates world state: Positions of all actors and objects in the 3D environment.	RTX Renderer renders the scene and warps rays according to lens behavior	Sensor model applied to add sensor specific behavior such as noise, etc.	ISP implemented to produce output.

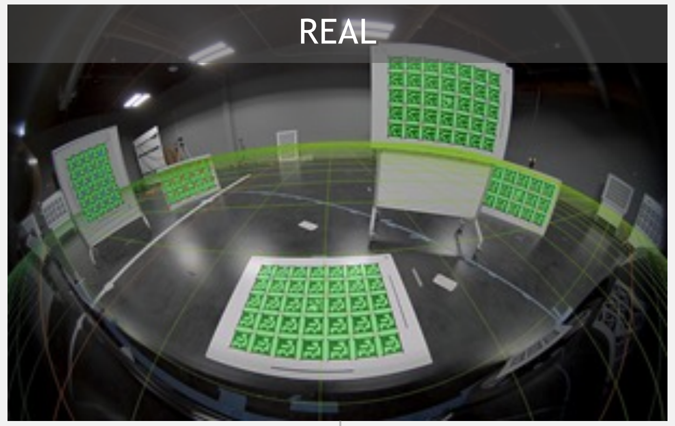
NARROWING THE APPEARANCE GAP: VALIDATED SENSOR MODELS

CAMERA MODEL VALIDATION

Camera Calibration

TEST SETUP

(1)

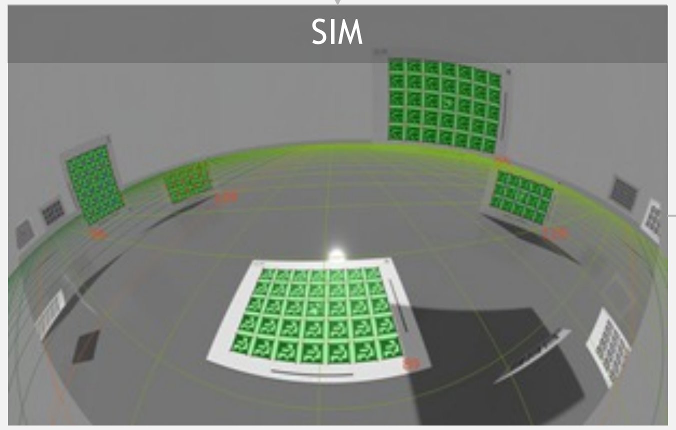


(2)

DRIVE WORKS
SENSOR CALIBRATION

RECREATE
SCENE IN SIM

(3)



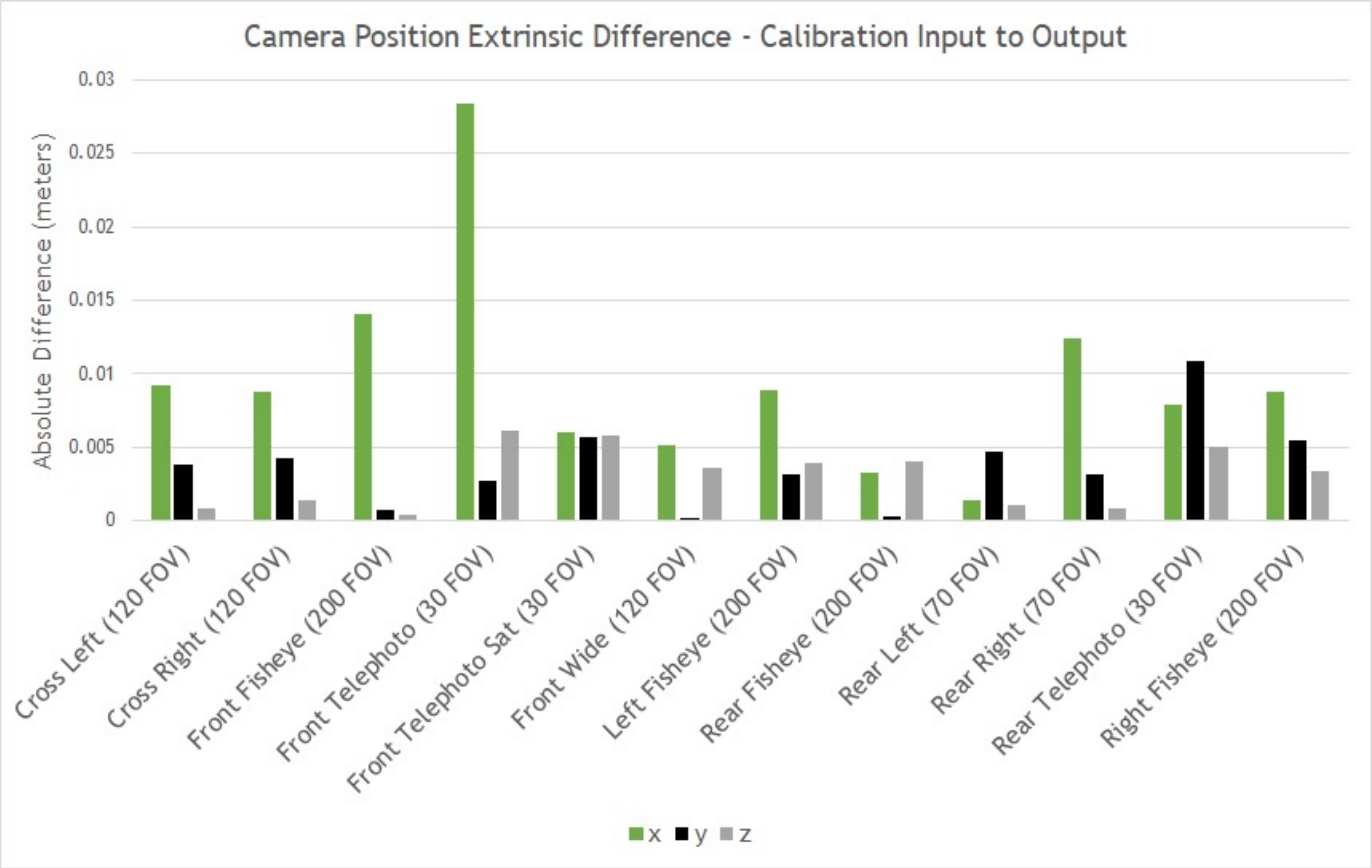
(4)

DRIVE WORKS
SENSOR CALIBRATION

(5)

Compare Calculated Camera Positions, Orientations, Lens Models

RESULTS



Extrinsics: Camera positions calculated from real and synthetic calibration images matched within **3 cm** and **0.4 degrees** along all axes.

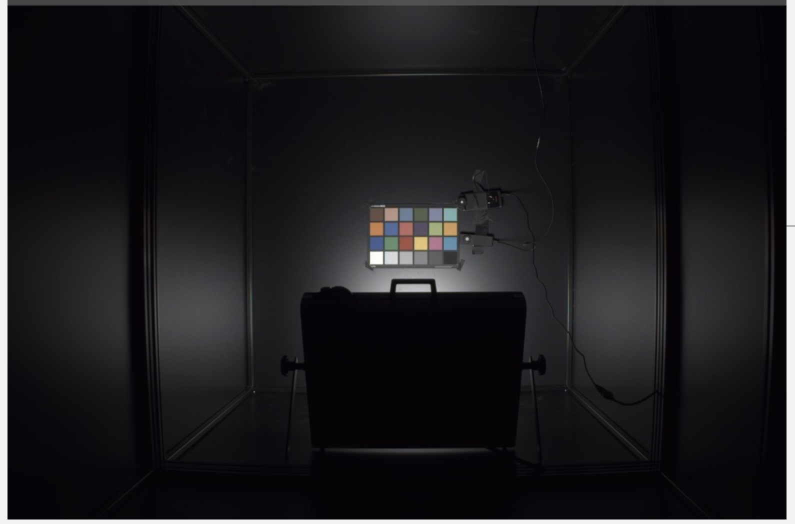
Intrinsics: Less than 0.5% distortion between real and simulated lenses (in terms of max angular difference, as a percentage of each lens's theoretical field of view).

CAMERA MODEL VALIDATION

Color Reproduction

TEST SETUP

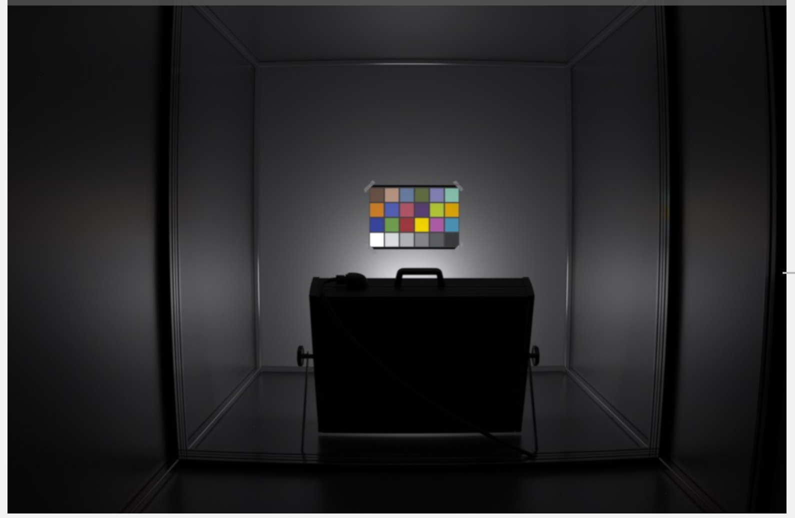
REAL



(1) Capture image of Macbeth ColorChecker chart in test chamber

(2) Reproduce the scene in DRIVE Sim.

SIM



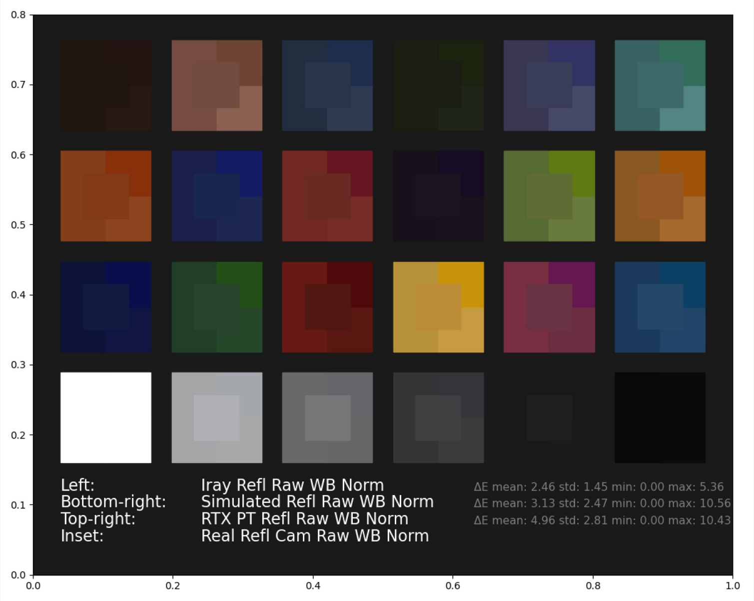
(3) Generate synthetic image with the DRIVE Sim camera model.

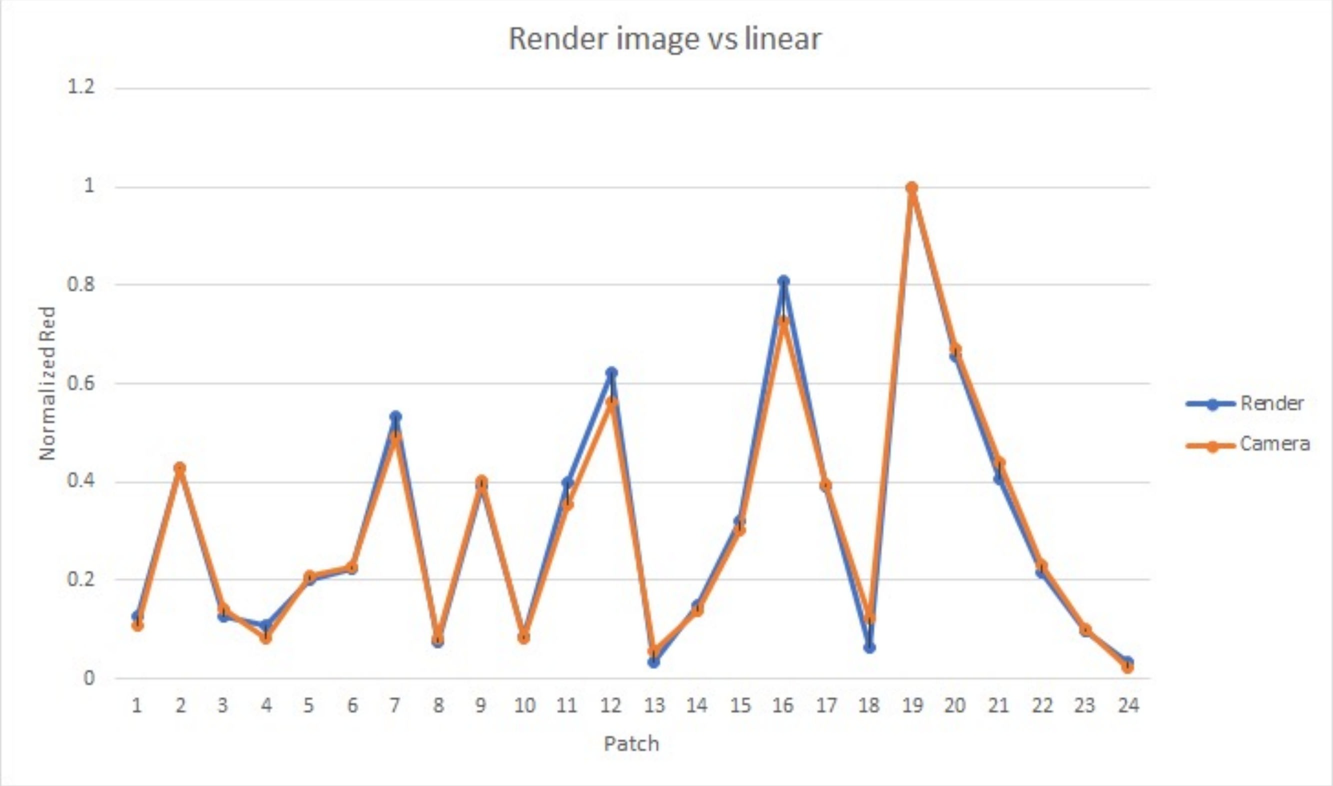
(4) Compare

Extract mean RGB and brightness values for each color patch.

Compare white-balanced values for synthetic and real images.

RESULTS





ACTIVE RTX SENSOR
LIDAR | RADAR | ULTRASONICS

DRIVE SIM - ACTIVE RTX SENSORS

Lidar | Radar | Ultrasonics

Physically Accurate: Ideal and divergent beams | Reflections and transmissions | Multi-bounce and multi-path effects

Time Based: Per-ray timestamps | Environment and ego interpolated per ray.

Materials: Computes full BSDF per hit - diffuse, specular, reflective returns | Built in Library | Customizable BSDF

Configurable: Full control over the origin, pattern and direction of each ray

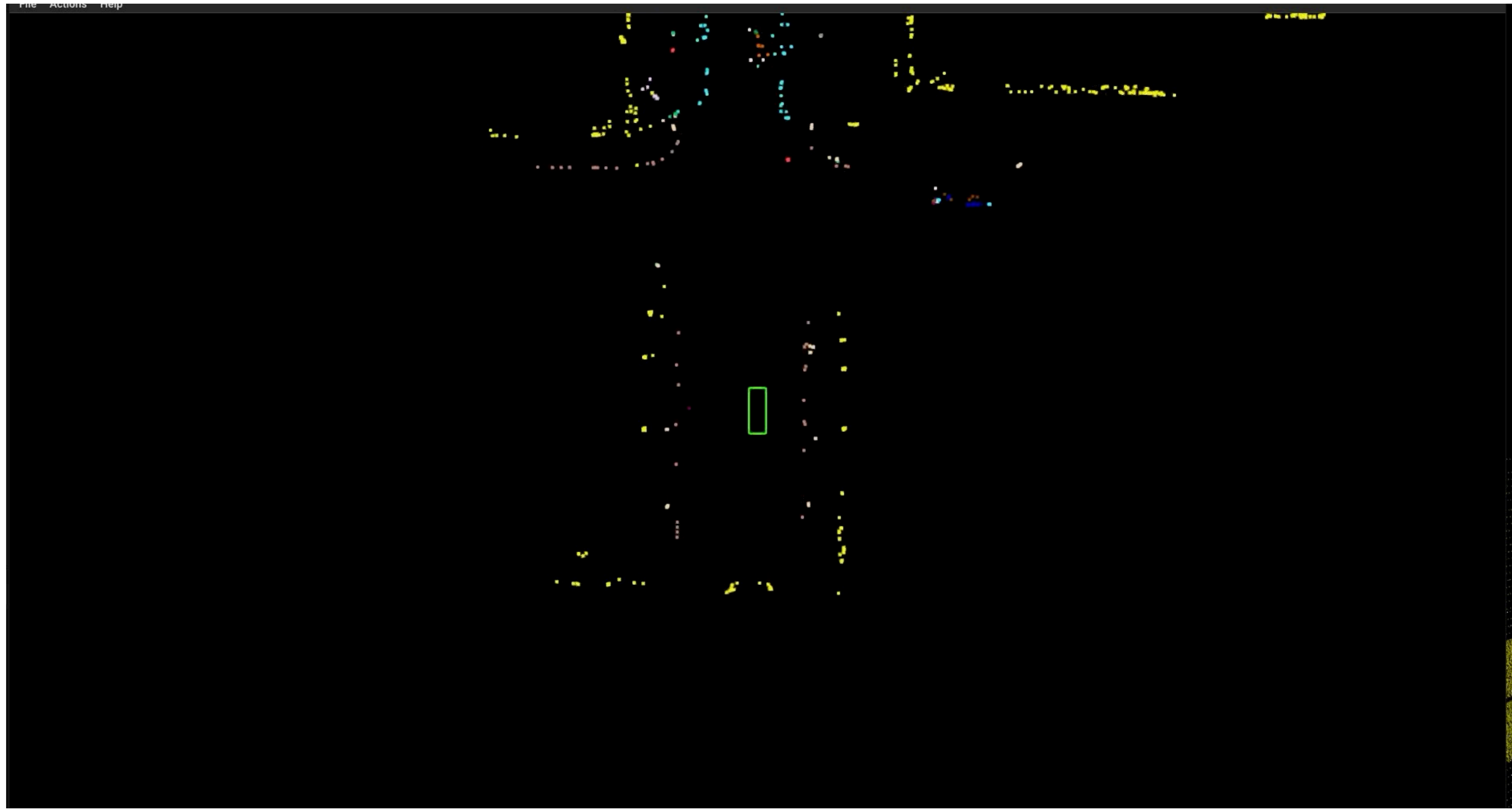
Out of the box support: DRIVE Sim supports multiple Lidar/Radar/USS models out of the box

Extensible: Custom sensor models can be built without exposing any of their internals or IP

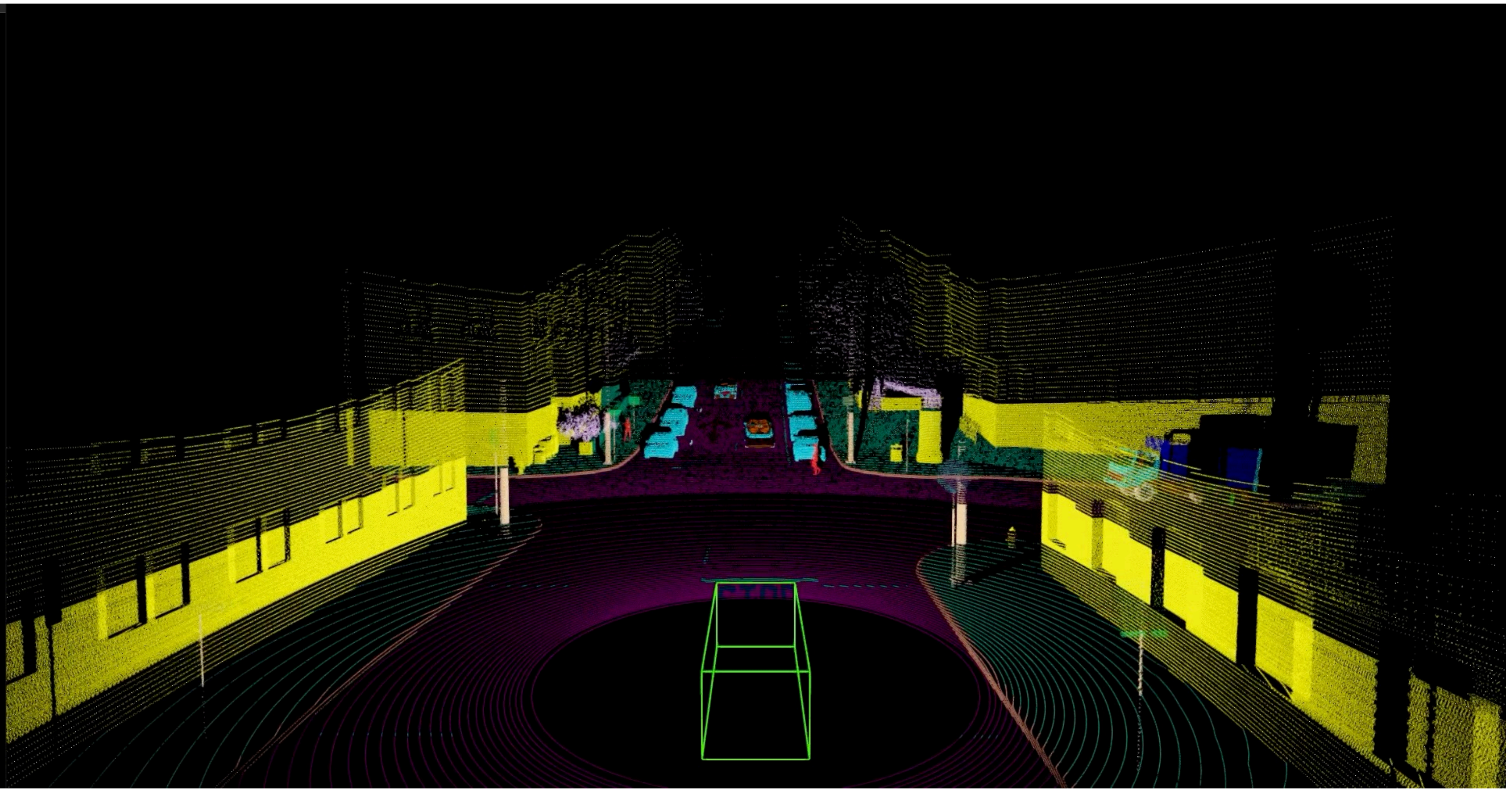
Scalable: Workload distributed across multiple GPUs and GPU nodes

RADAR / LIDAR EXAMPLE

8 Radar Aggregate View



360 deg Lidar View



LIDAR/RADAR MATERIALS

Generic material model

Covers material properties for non-visual spectrum

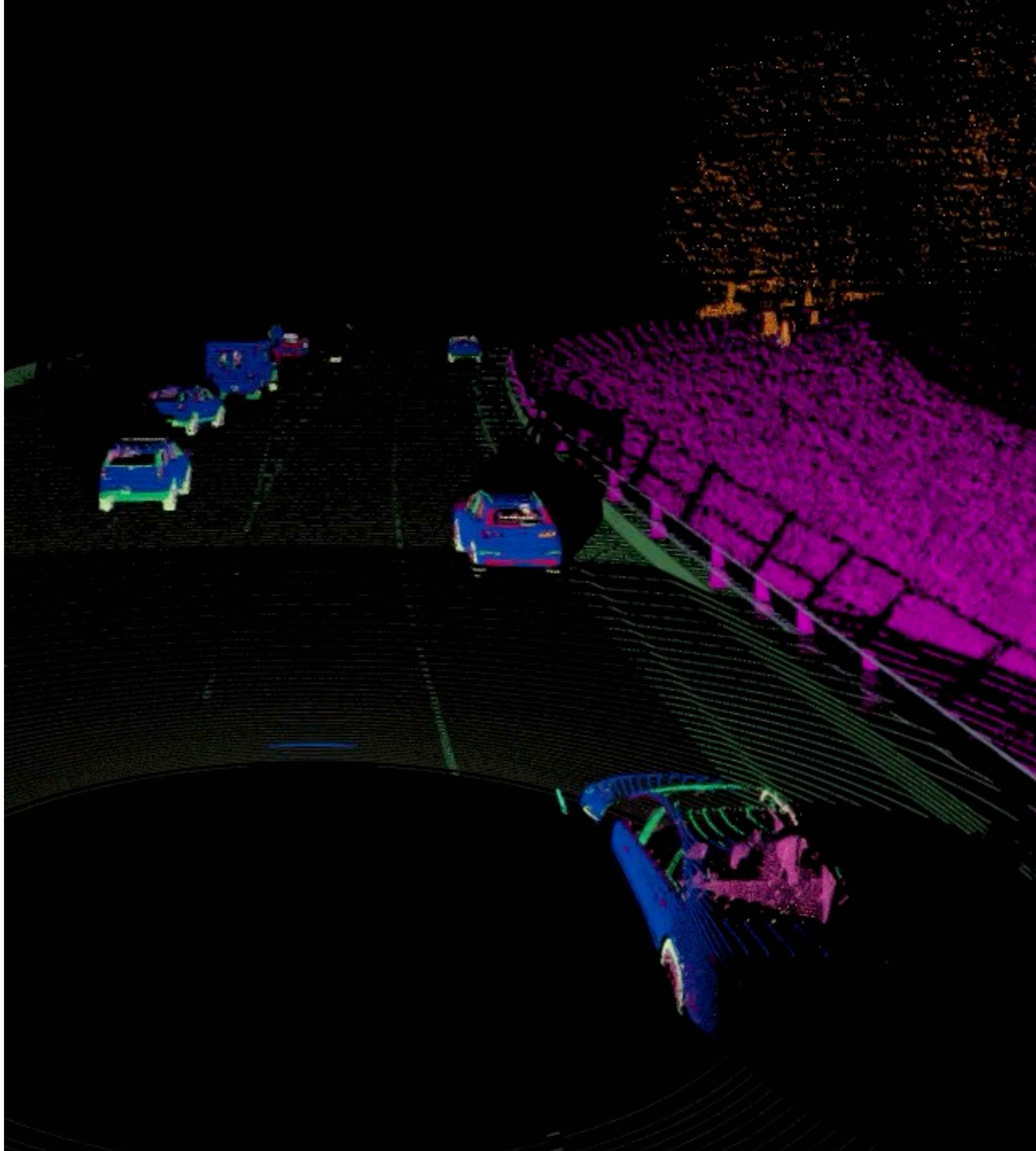
Ready-to-Use materials database

Implementation supports advanced modeling effects

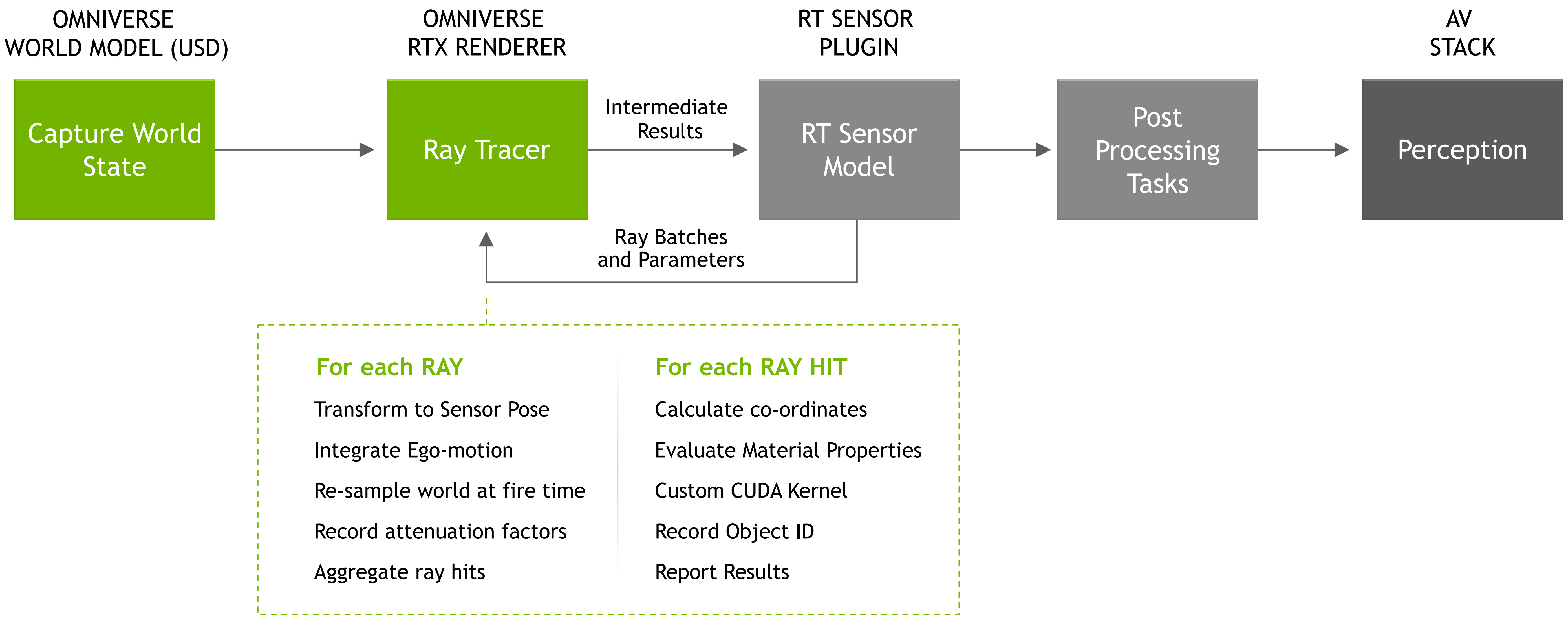
(Polarized light, LIDAR speckles…)

User-definable material properties

IP-protected material model exchange

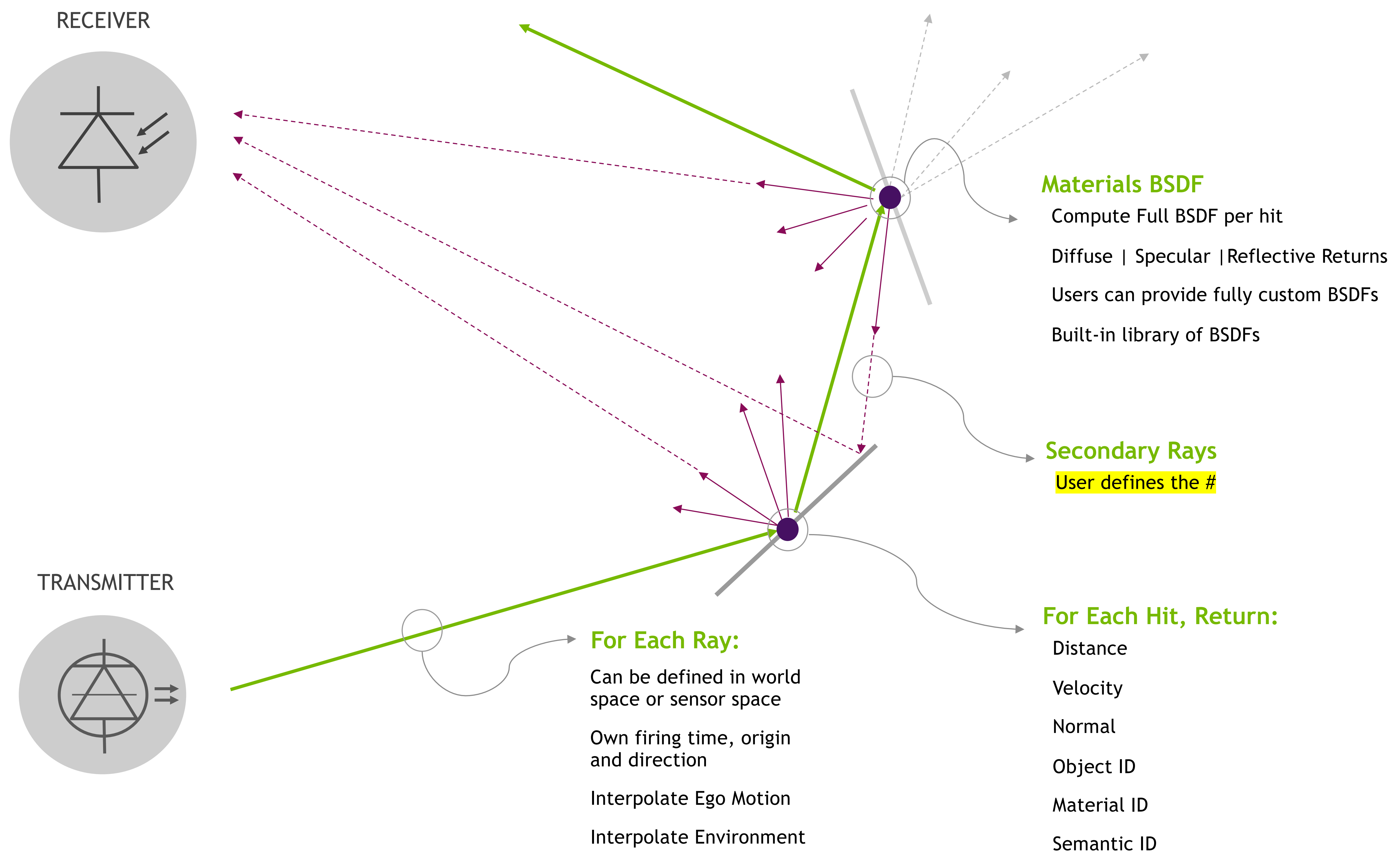


RTX SENSOR MODEL PIPELINE



DRIVE SIM - ACTIVE SENSORS WITH RTX

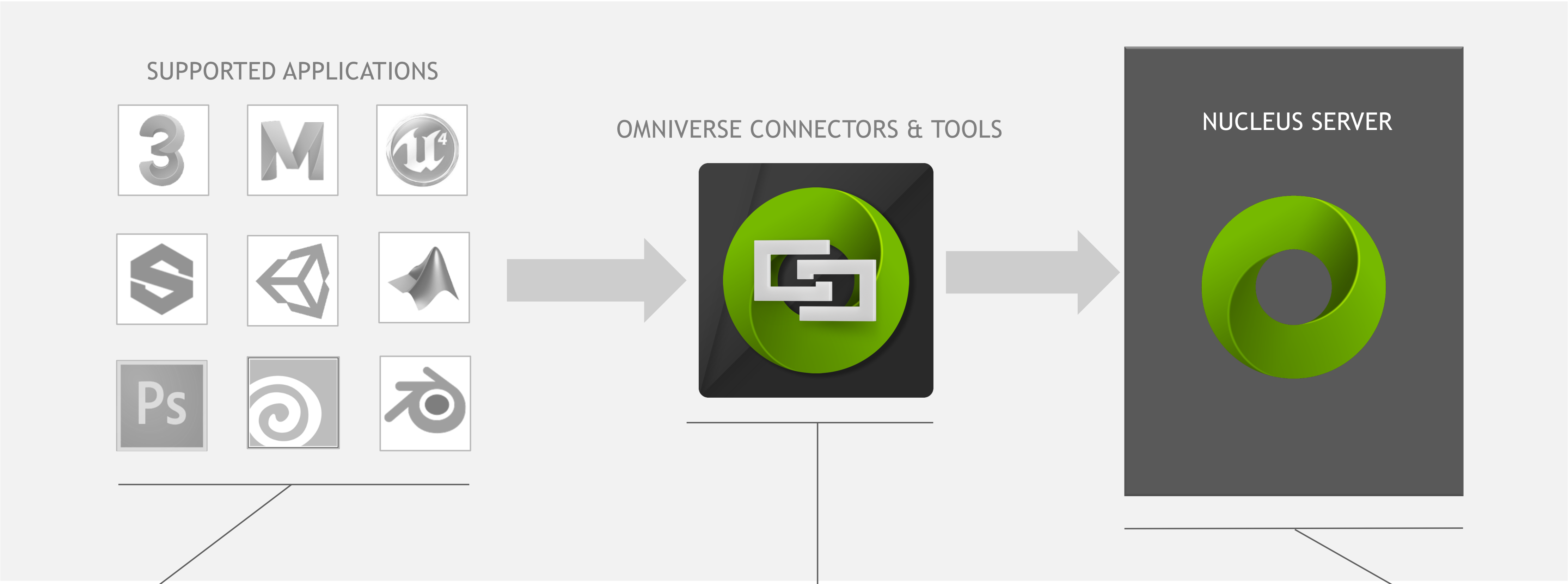
--



NARROWING THE CONTENT GAP WITH DRIVE SIM SDG AND OMNIVERSE

ASSET AND ENVIRONMENT IMPORT

ENABLING ACCESS TO A LARGE ASSET POOL



AV Scene 3D models created in DCC tools

Omniverse Application Plugins Convert DCC Assets/Scenes to **SimReady USD** Content

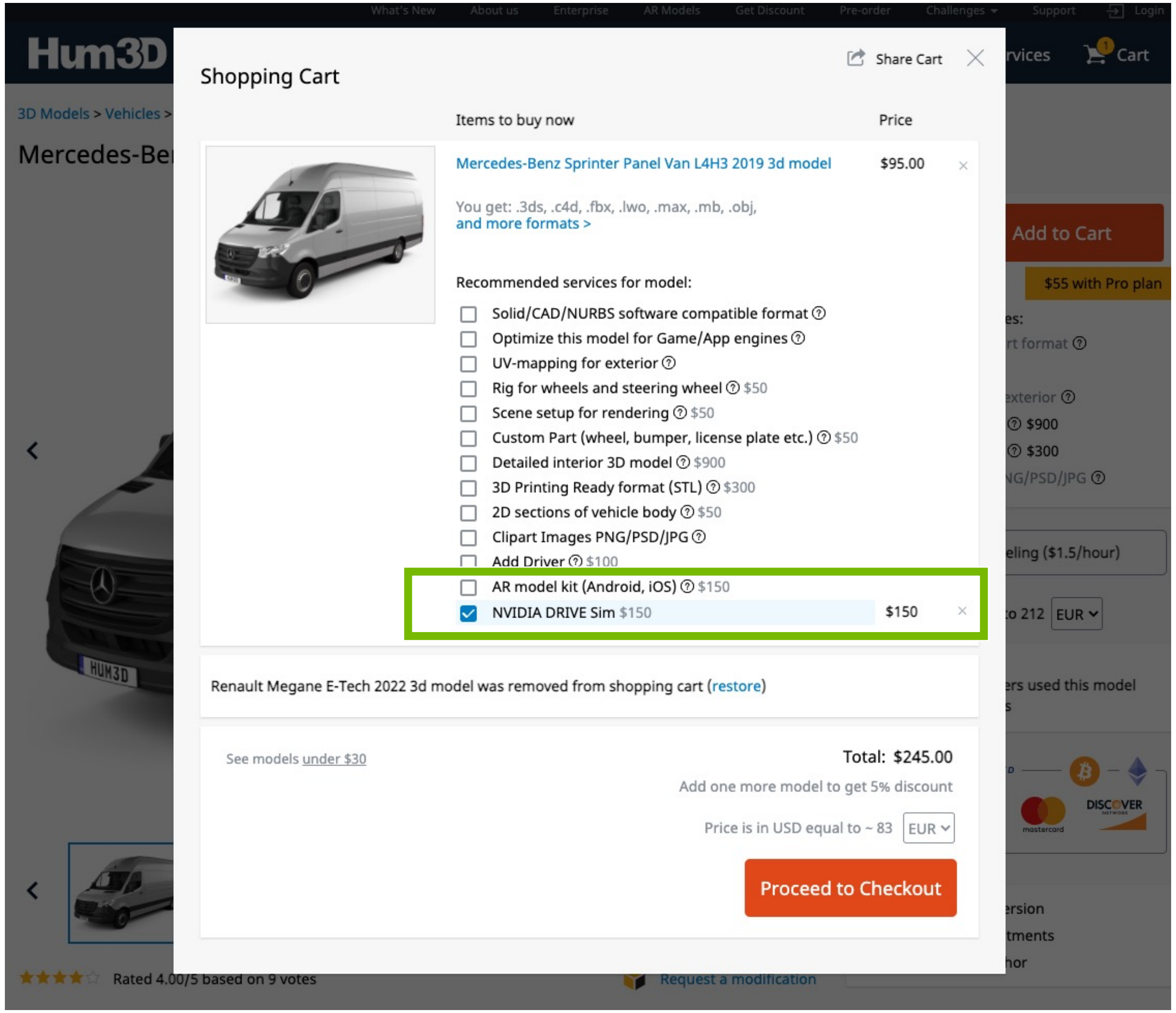
Content fidelity maintained/enhanced in conversion including geometry & materials

SimReady USD Assets stored on Nucleus server

- Materials compatible with Sim sensors
- Training ready
- Physics ready
- Functionally ready

DRIVE SIM READY ASSETS

- Models built to a set of requirements and best practices, including:
  - Model size, dimension, orientation
  - Parts grouped in proper hierarchy
  - Named appropriately for class designation
- Optimized for rendering
- Prepared for ingest by DRIVE Sim content tools
- Ready for DRIVE Sim material application
- Supported by various ecosystem partners



NVIDIA

ENABLING DRIVE SIM 3D CONTENT ECOSYSTEM

DRIVE Sim Ready USD Assets

Vehicles & Props

HKV

Vehicles & Props

Hum3D

Humans

REALLUSION®

Props

TURBOSQUID by shutterstock

Props

SIMACTION SIMULATION

Vegetation

Globe Plants

Vegetation

Graswald

Props, Signs & Buildings

bdesign3D

Chinese Props & Signs

51 WORLD

Buildings & Vegetation
Procedurally Generated

blackshark.ai

CREATING ROAD NETWORKS FOR DRIVE SIM

Reference Map (**Optional**)

Survey Map

Aerial Map

MathWorks®

OpenDRIVE Map

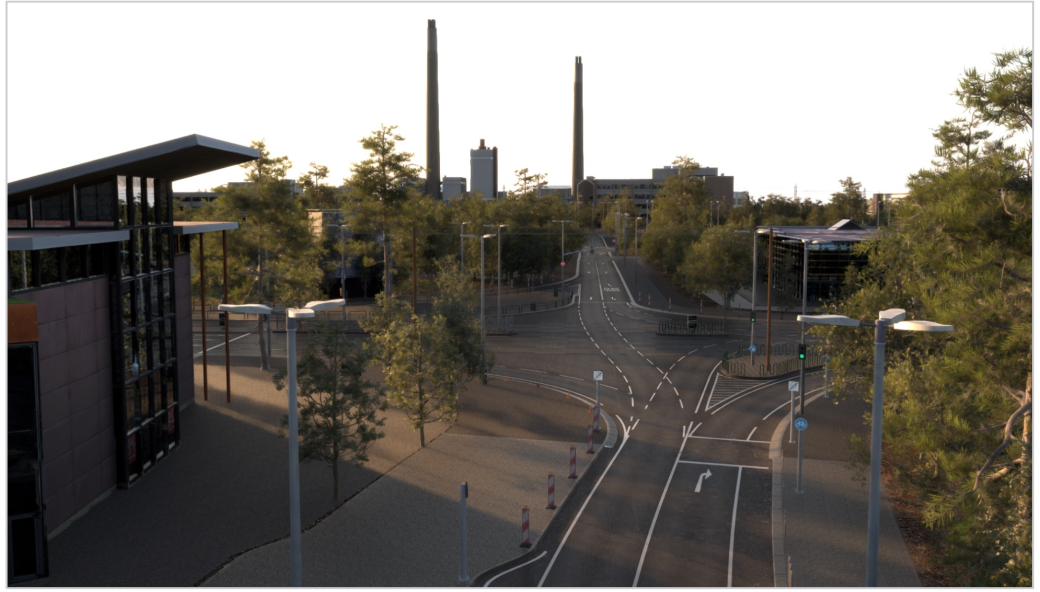
Signal Phasing

Scene Geometry

NUCLEUS SERVER

SCENES CREATED FOR DRIVE SIM

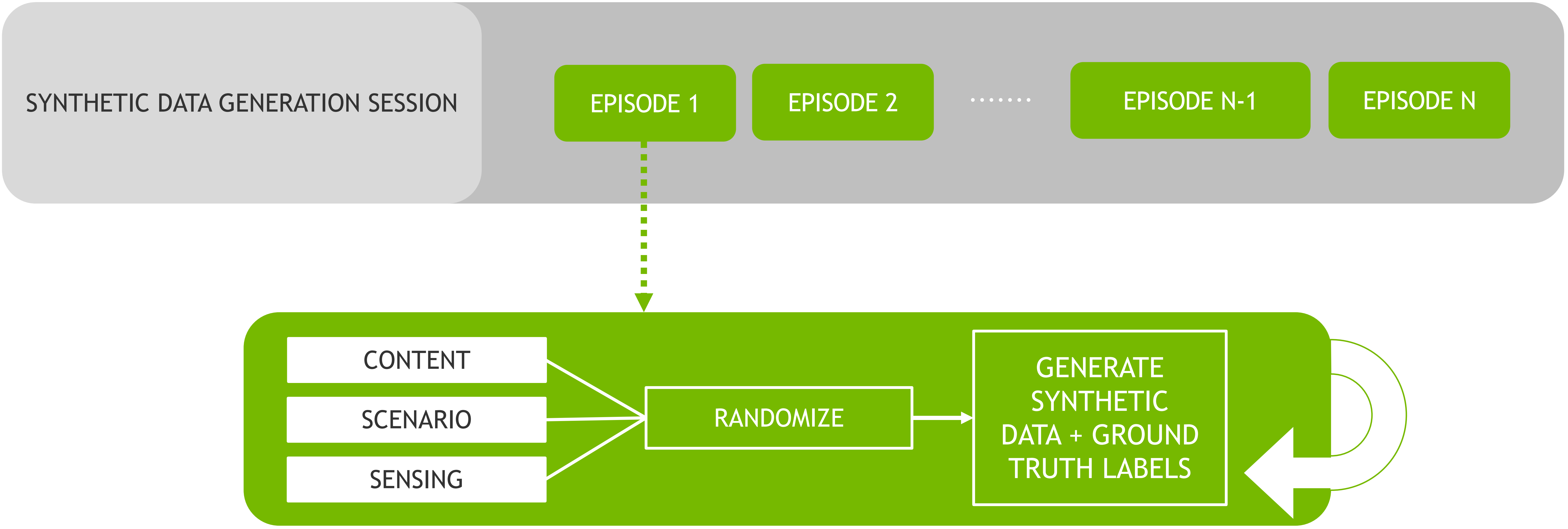


DOMAIN RANDOMIZATION

DOMAIN RANDOMIZATION

Scaling synthetic data



DOMAIN RANDOMIZATION ENGINE

Apply randomizations in a structured way

Ensure scene coherence (e.g. cars are not overlapping when placed, etc)

Use simulation engine to apply motion / vehicle physics / sequential actions for sequential datasets

Repeatable – random seeds to ensure the same random result can be produced again

Tools are a combination GUI + Python-based scripting

Randomization Vectors

Environment:

- 3D scene
- Lighting
- Weather
- Material
- Road markings
- Sign / Signal type
- Dirt / weathering / wear

Ego:

- Vehicle Type
- Sensor Position
- Sensor Type

Assets:

- Type
- Color
- Position / orientation
- Movement / behavior
- Lights / light intensity

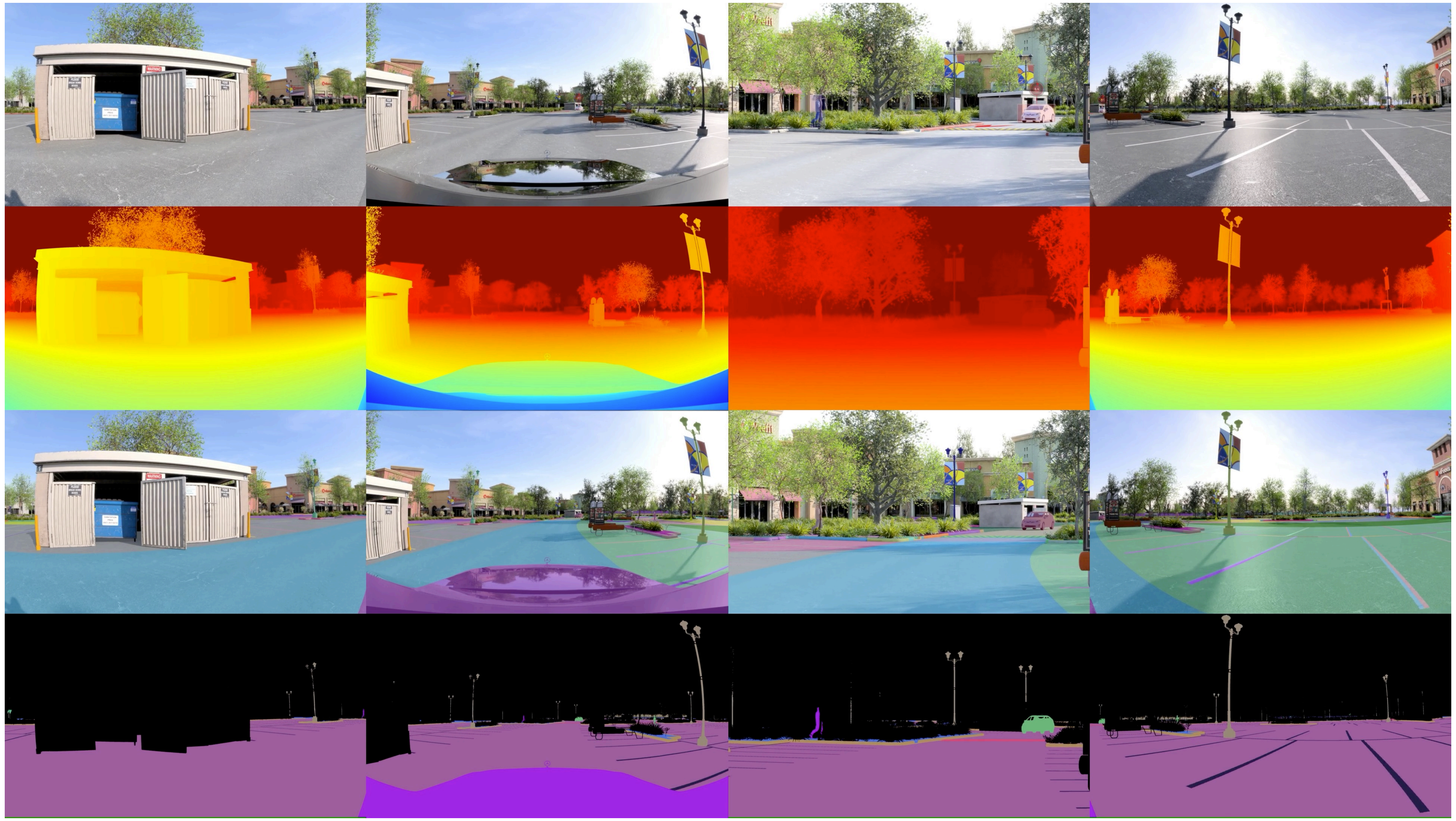
DOMAIN RANDOMIZATION

Parking lot example

DOMAIN RANDOMIZATION

Parking lot example with ground truth

	Left 120FOV	Center 120FOV	Center 30FOV	Right 120FOV
3D Cuboids				
Depth				
Instance Segmentation				
Semantic Segmentation				

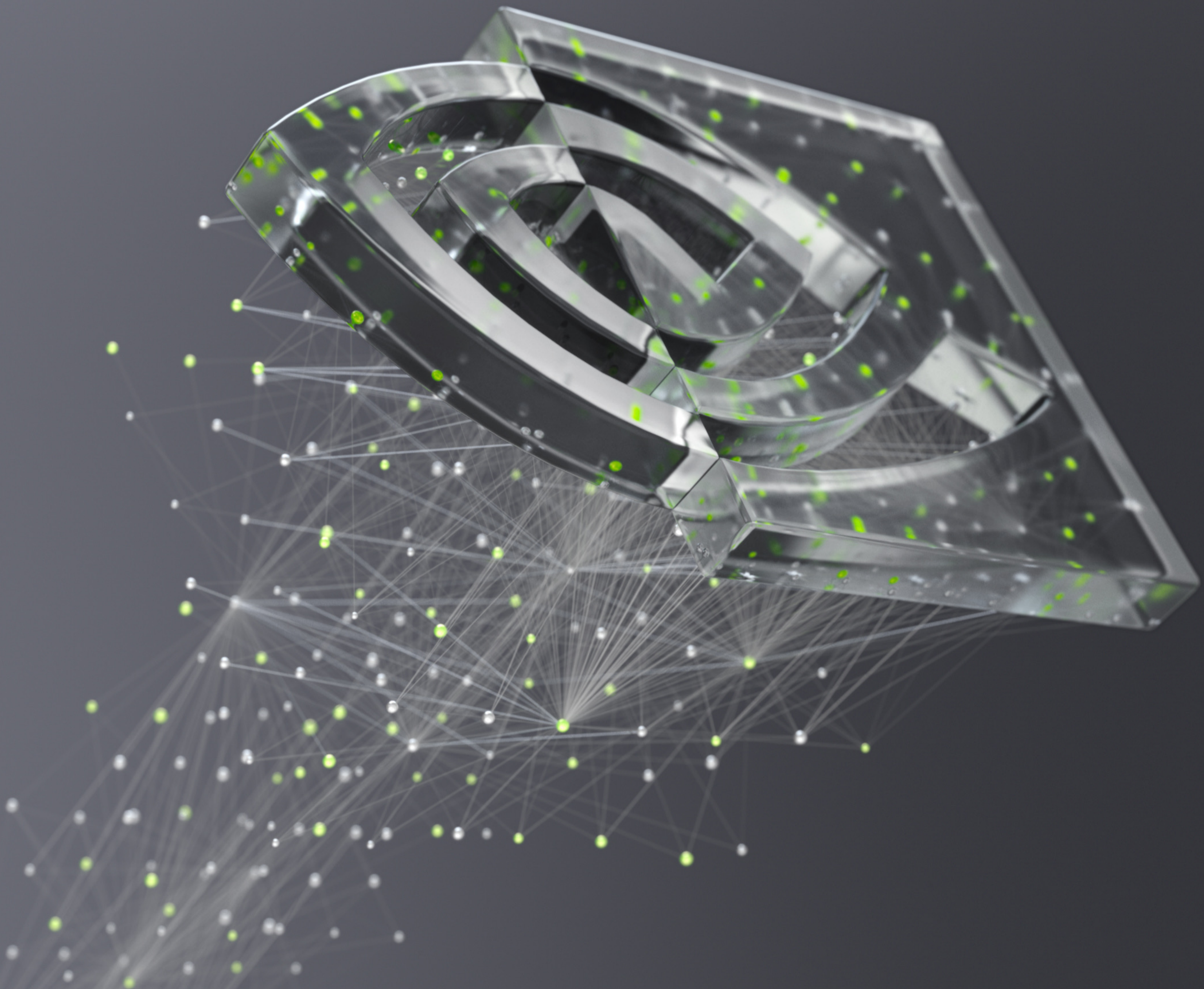


SUMMARY

Synthetic Data Generation offers many new opportunities including:

- Shift to focus on engineering GOOD data
- Reduce costs / iterate faster
- Improve label quality
- New data source for impossible scenes (e.g. occluded objects)
- Tackle the long tail

Matt Cragun | mcragun@nvidia.com



NVIDIA