SET Level

Simulation-based Development and Testing of Automated Driving

Supported by:

Federal Ministry for Economic Affairs and Energy

on the basis of a decision by the German Bundestag

PROJECT of the PEGASUS FAMILY

Simulation Use Case 3 - An open-loop simulation for component test and validation

Philip Ockert, Marzena Franek and Philipp Rosenberger

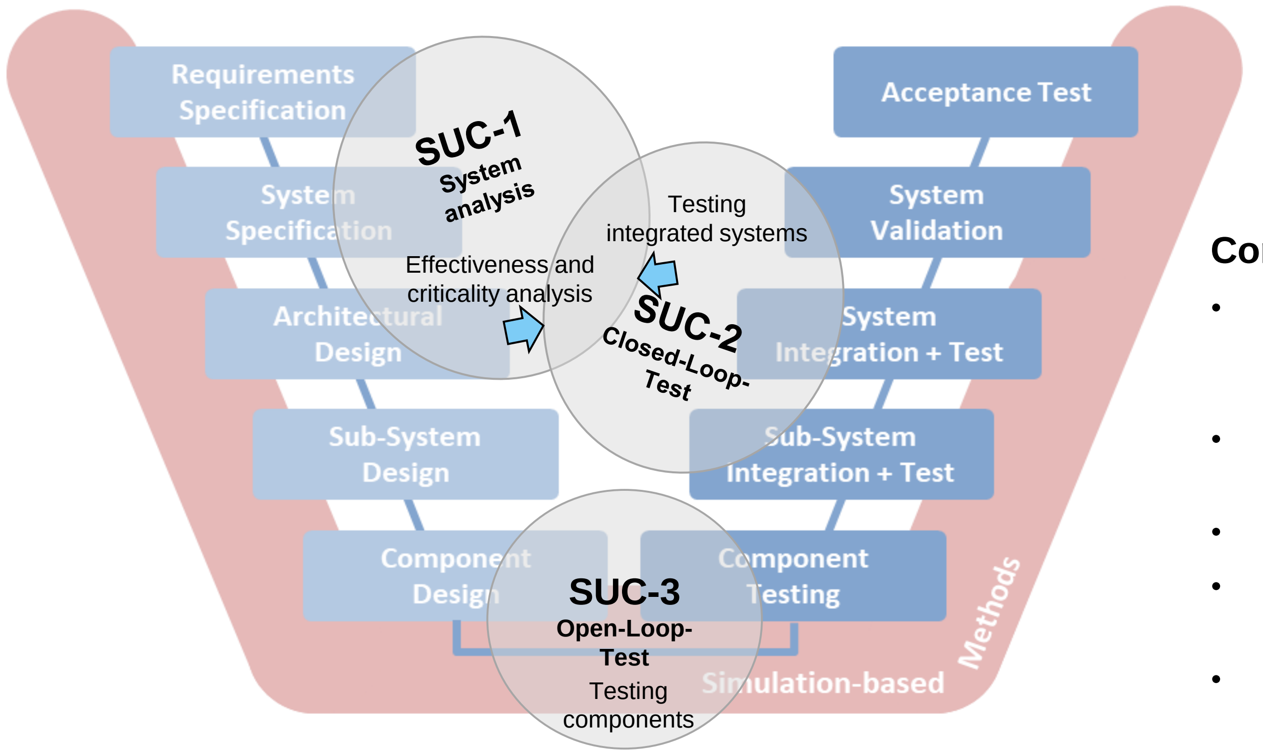
29.04.2021 – 9:00 AM

Agenda

- Introduction of Simulation Use Case 3 (SUC3)
- Specification
- Implementation
    - User stories specific simulation setup for camera and lidar component test
    - IPG and dSPACE tool specific implementation
- Demonstration of results
    - Camera component: Model design, metrics and results
    - Lidar component: Model design, metrics and results
- Summary and outlook
- Q&A

Classification

Introduction of Simulation Use Case 3 (SUC3)



Simulation Use Cases (SUCs):

- Analysis example
- Test examples

Common Demonstration goals:

- Demonstration of the applicability and usability of standards (OSI, FMI, SSP, …)
- Usage of appropriate architectures and interfaces
- Elaboration of KPIs
- Use of the credible simulation process and ensurance of traceability
- Provide project internal feedback and identify need for further work

Simulation goals

Introduction of Simulation Use Case 3 (SUC3)

- Goal is to work on camera and lidar in parallel
- Goal is to execute the simulations in dSPACE and IPG tool

	dSPACE	IPG
Camera	Simulation 1	Simulation 2
Lidar	Simulation 3	Simulation 4

Demonstration of results

- Specifically for each sensor technology
  - Application and evaluation of OSI for component testing
  - Application and evaluation of OpenDrive and OpenScenario
  - Test and verification of SetLevel models for sensor component

User story – camera component test

Introduction of Simulation Use Case 3 (SUC3)

- As a component developer, I would like to
  - test **my camera component** scenario-based, isolated and without interaction in an overall-system simulation agreed with the OEM AND
  - be able to document the editors, processing steps, tools, artifacts and results necessary for traceability of the test process in order
  - to be able to make statements about the **effects of updating optics and imager** on the **edge detection** of the signal processing function and to compare this with a **reference** AND
  - to comply with the obligation to provide evidence (towards the OEM) and to ensure the reproducibility of test results.

- For this purpose, the following **specific test objective** was set:
  - the effect of updating the imager and optics is rated with the help of **standard comparison metrics** such as MSE (mean-square error), RMS (root-mean-square), PSNR (peak signal-to-noise ratio) and SSIM (structural similarity index measure) on **the quality** of the **output images of the camera component**
  - Therefore, the image output of **the camera sensor model** with included edges must be **compared and assessed** with the image output of **a reference camera sensor** model over the course of the **entire simulation**.
  - A **real camera** could also be used **as a reference** camera sensor

User story – lidar component test

Introduction of Simulation Use Case 3 (SUC3)

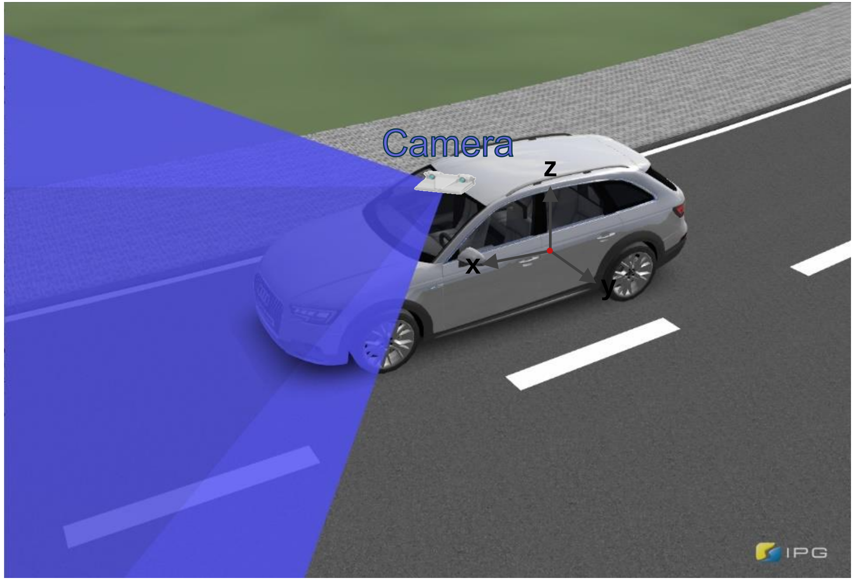
- As a component developer, I would like to
  - test **my lidar component** scenario-based, isolated and without interaction in an overall-system simulation agreed with the OEM AND
  - be able to document the editors, processing steps, tools, artifacts and results necessary for traceability of the test process in order
  - to be able to make statements about the **accuracy of object recognition and tracking** based on a specified metric AND
  - to comply with the obligation to provide evidence (towards the OEM) and to ensure the reproducibility of test results.

- For this purpose, the following **specific test objective** was set:
  - The **intersection over union** is used to compare the **accuracy of the overlap between two bounding boxes** (Ground truth and detected bounding box)
  - The key figure derived in this way **assesses the detection algorithm and the tracking** of the sensor model over the course of the overall simulation

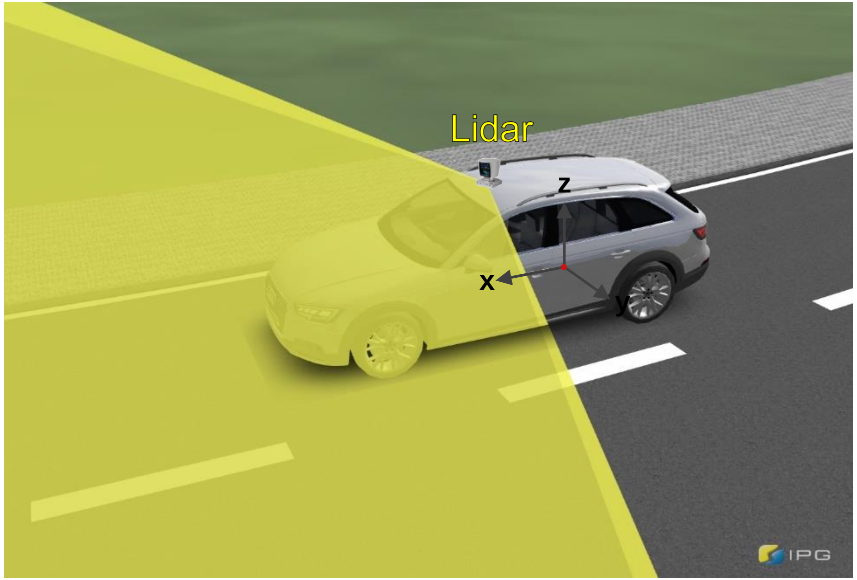
Vehicle setup

Specification

Position camera in simulation

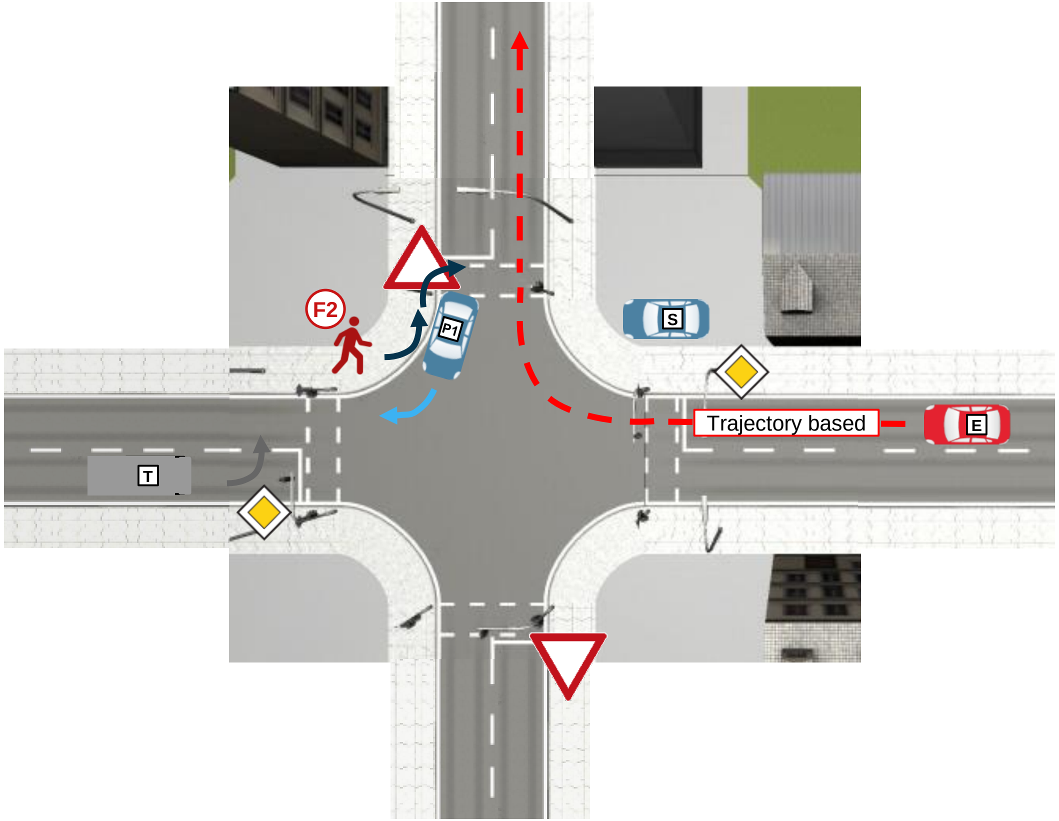


Position lidar in simulation



Scenario

Specification



TS2-4/RS2-6
truck, oncoming carriageway, right-turning driver

Objects:

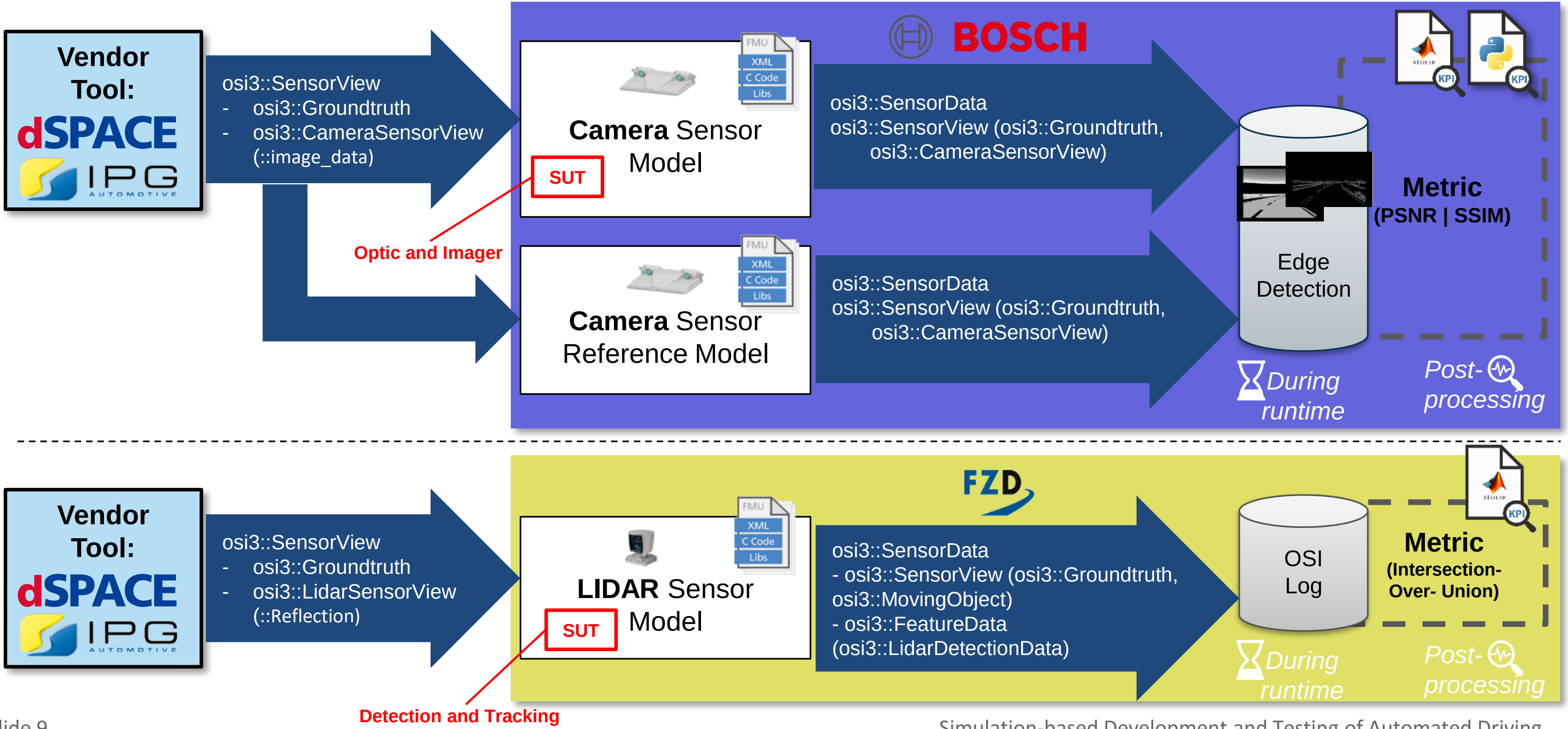
- EGO: car (E)
- traffic participant:
  Car (P1, S), truck (T)
  pedestrian (F2)

Behavior control:

- EGO:
  course is trajectory based (open loop/no HAD)
- Environmental traffic:
  course is trajectory based
- No parameter variation
  ➔ Focus on first integration of sensor models

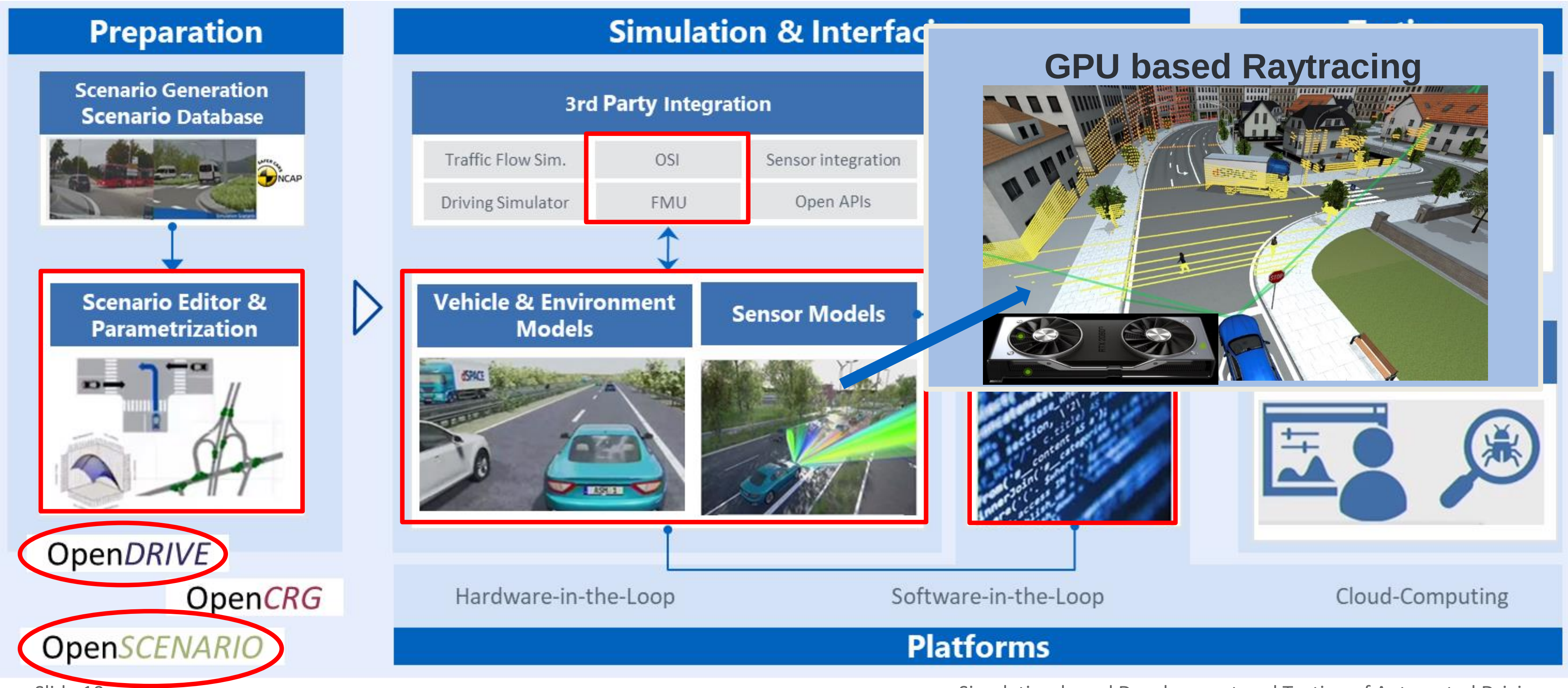
User stories specific simulation setup

Implementation



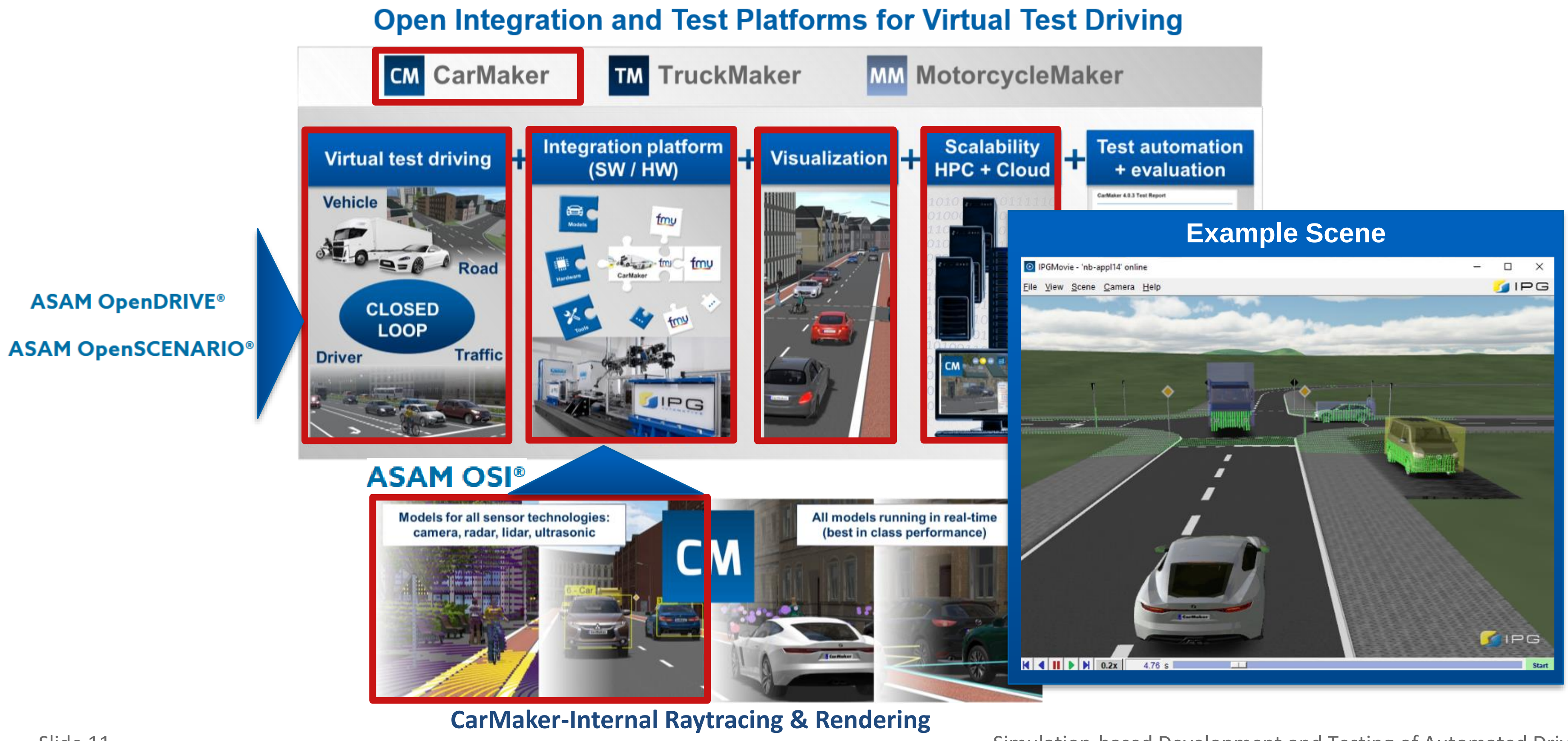
dSPACE - tool specific implementation

Implementation



IPG - Tool specific implementation

Implementation



Model design for camera componet

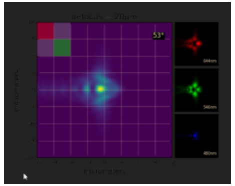
Demonstration of results

- First successful implementation of an image-based video sensor models on base of OSI 3:CameraSensorView Interface

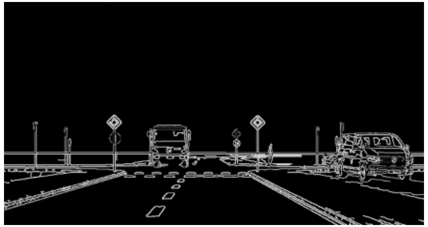
Original Image_Data
32bit RGB

Image_based_video_detected_edges_model

Blur in the image is modeled as a convolution with the **point spread function**



Canny Edge



Output: edge detection data

Simulated:

- SUT 01 „Good Lens“
- SUT 02 „Bad Lens“ (**out of focus lens with lots of optical aberrations**)
- Parameter: pixel_horizontal, pixel_vertical 1920x1080

Metrics for camera component

Demonstration of results

MSE error: the smaller, the bigger the similarity

```
mse = (norm(A(:)-B(:),2).^2)/numel(A);
```

PSNR: smaller, the bigger the similarity

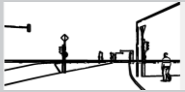
```
err = mse(A,B);
peaksnr = 10*log10(peakval.^2/err);
```

https://en.wikipedia.org/wiki/Peak_signal-to-noise_ratio

SSIM: the bigger, the bigger the similarity

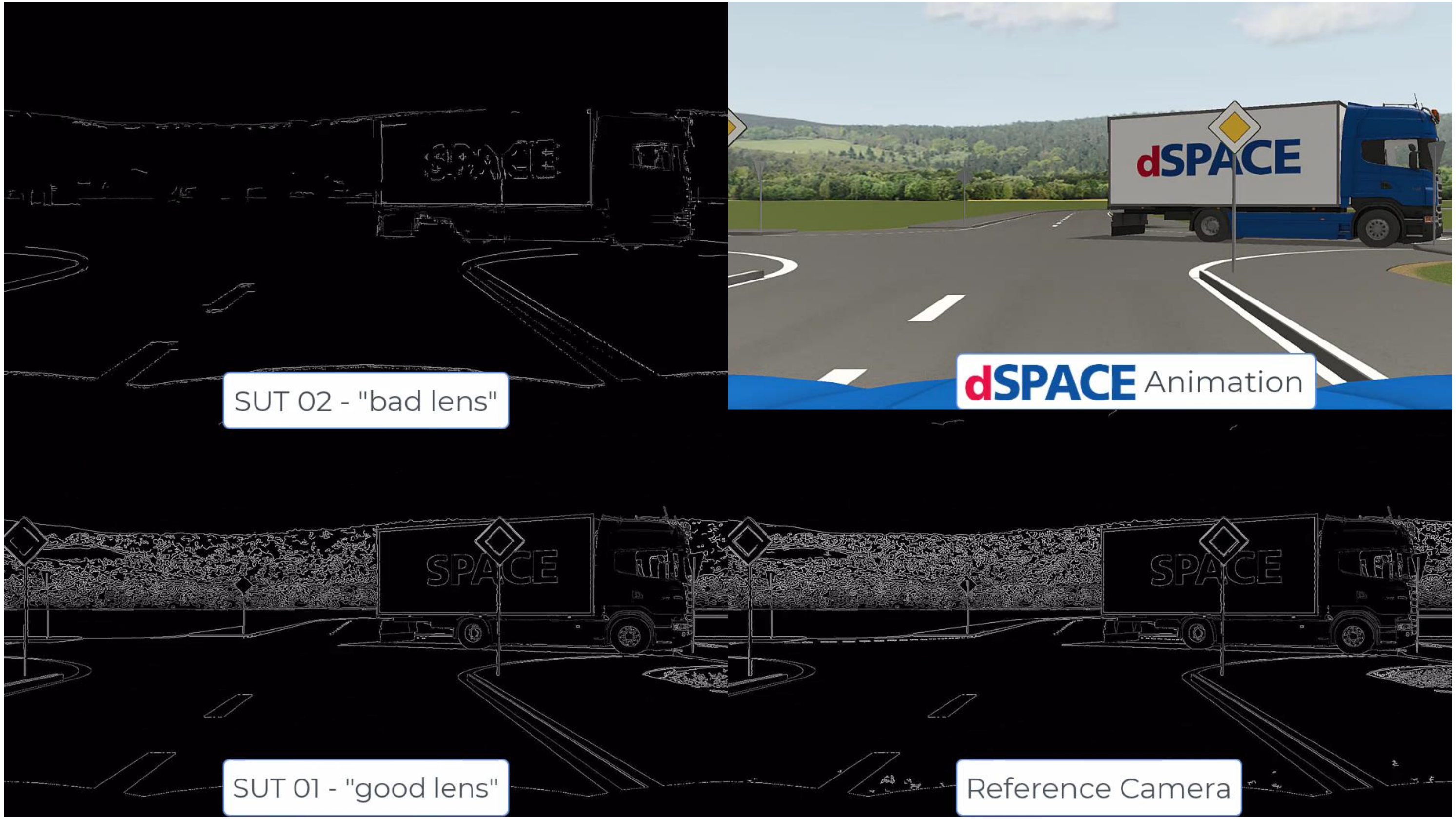
```
[ssim,ssimmap] = ssim(A,B);
```

https://en.wikipedia.org/wiki/Structural_similarity

Reference	Testimage	MSE	PSNR	SSIM	SSIMMap
A_Time0.92	B_Time11.22	368.9	22.7	0.869	
A_Time0.92	B_Time2.8	268	23.9	0.91	

Video for camera component

Demonstration of results

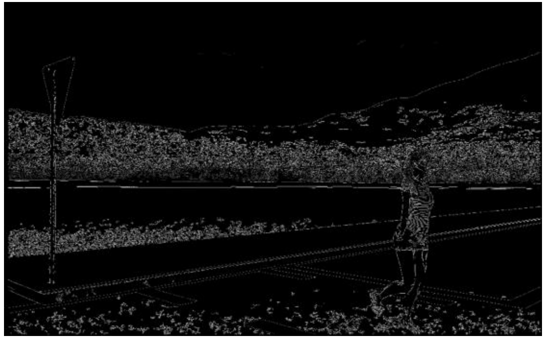


Results for camera component

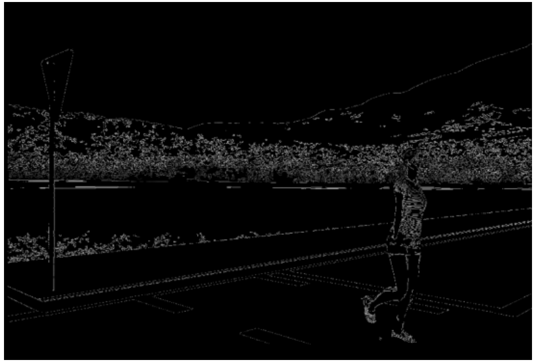
SET Level

Demonstration of results

reference images



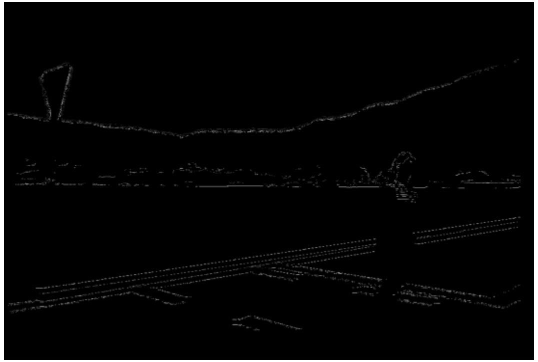
SUT 01 – „good lens“



TimeStamp	MSE	PSNRValue	SSIMerror	Eval
10.77	1107.5	17.688	0.84699	'OK'

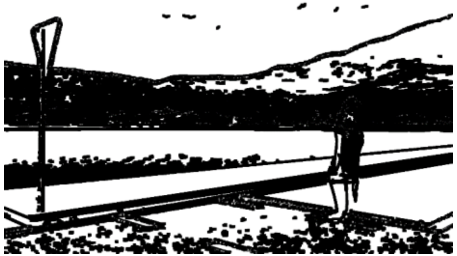
SSIM- Map

SUT 02 – „bad lens“



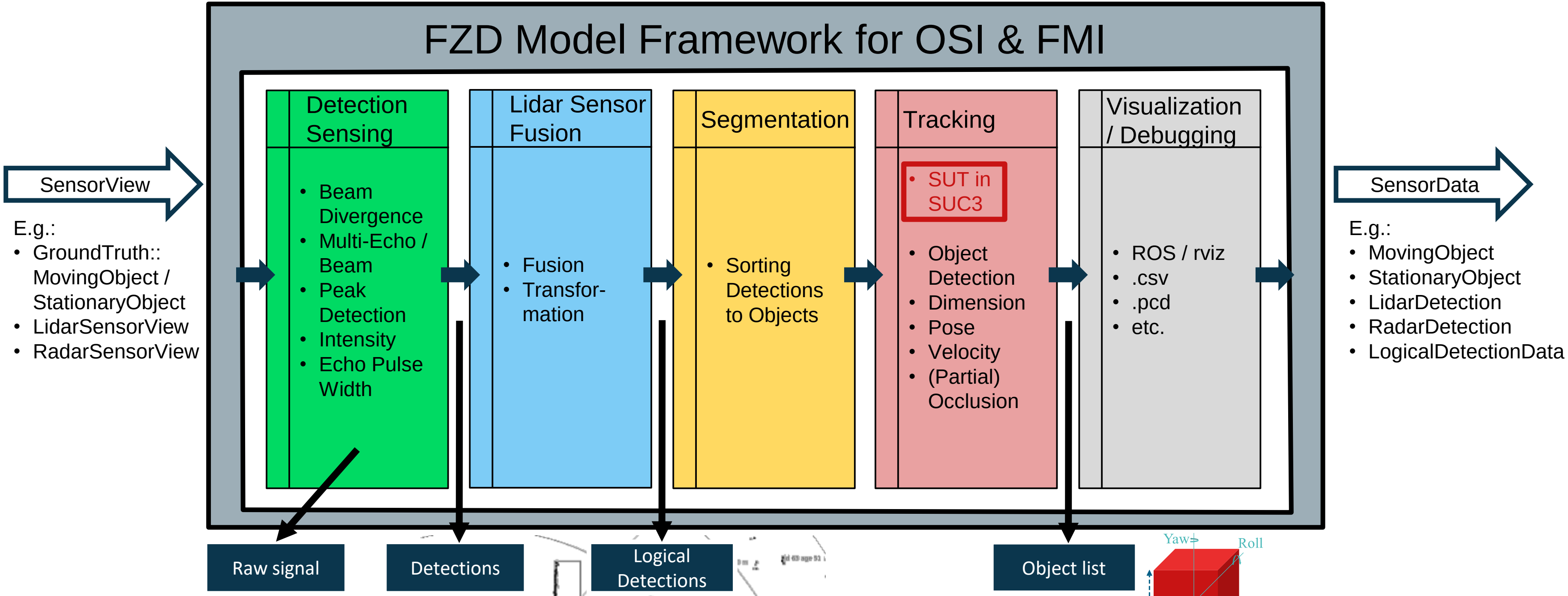
TimeStamp	MSE	PSNRValue	SSIMerror	Eval
10.77	3110.6	13.202	0.55615	'Fail'

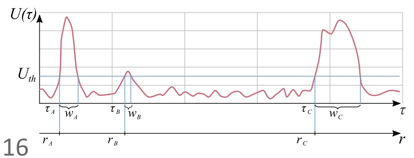
SSIM-Map

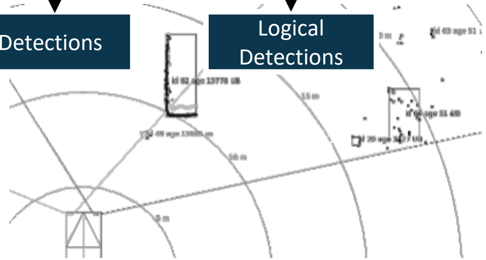


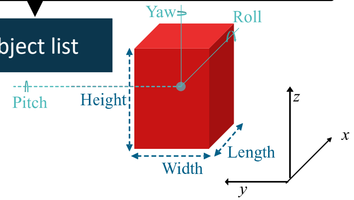
Model design for lidar component

Demonstration of results





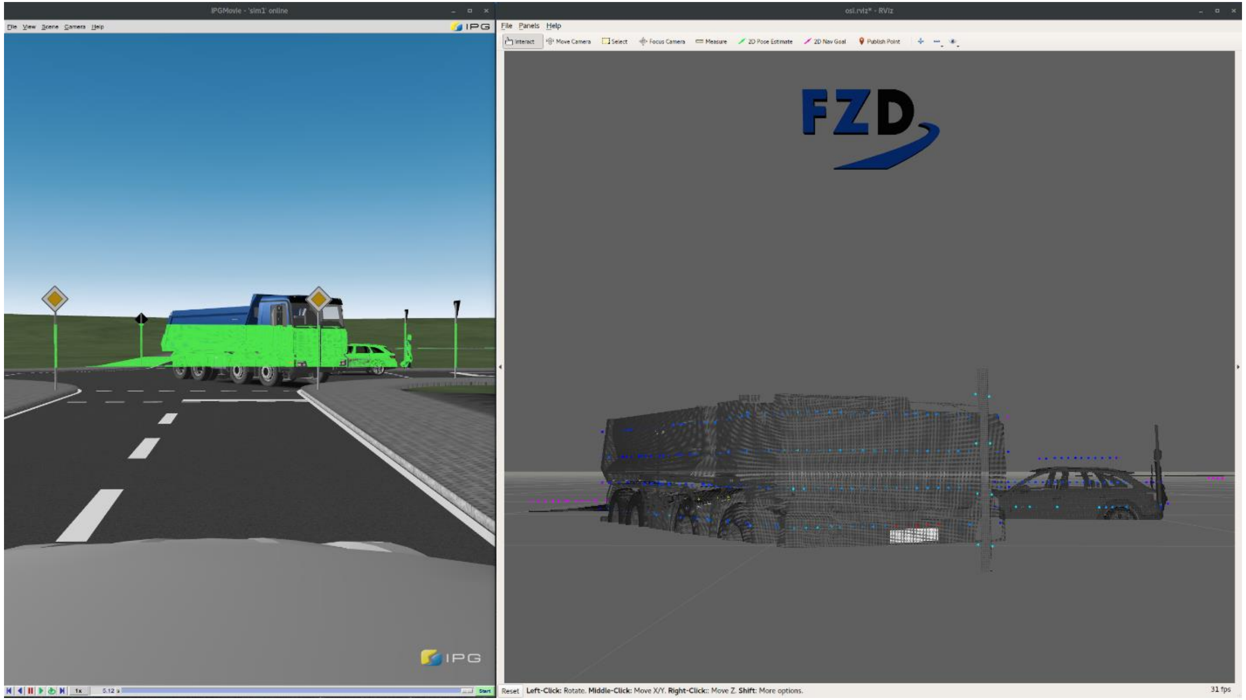




Model design for lidar component

Demonstration of results

- Lidar phenomenon in focus on object list: Occlusion by other objects



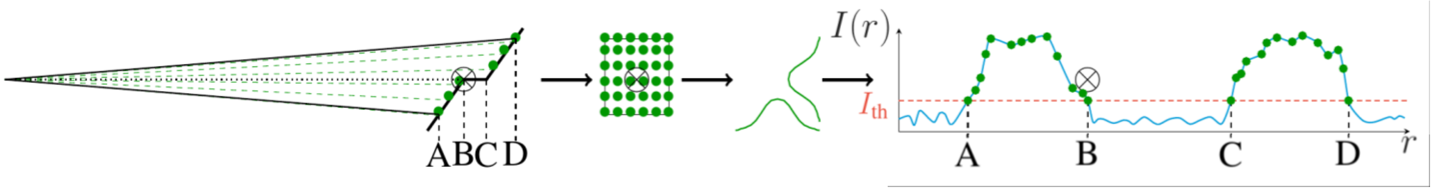
Reflections from tool

Detections from model

Model design for lidar component

Demonstration of results

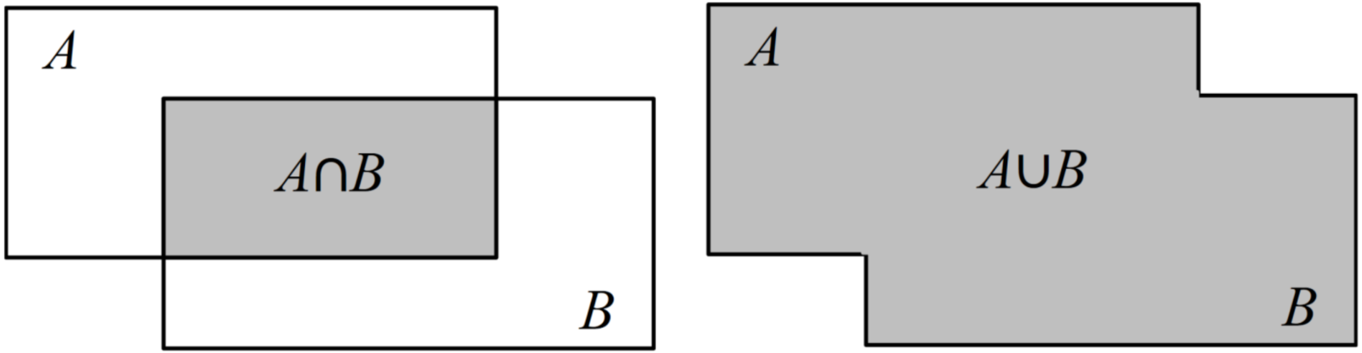
- Lidar effects to be modeled on detection level to generate realistic occlusion behavior on object level:
  - Beam divergence
  - Thresholding on intensity of signal over radial distance (I_{th} on $I(r)$)
  - Multi-echo per beam (A-B and C-D)



Metric for lidar component

Demonstration of results

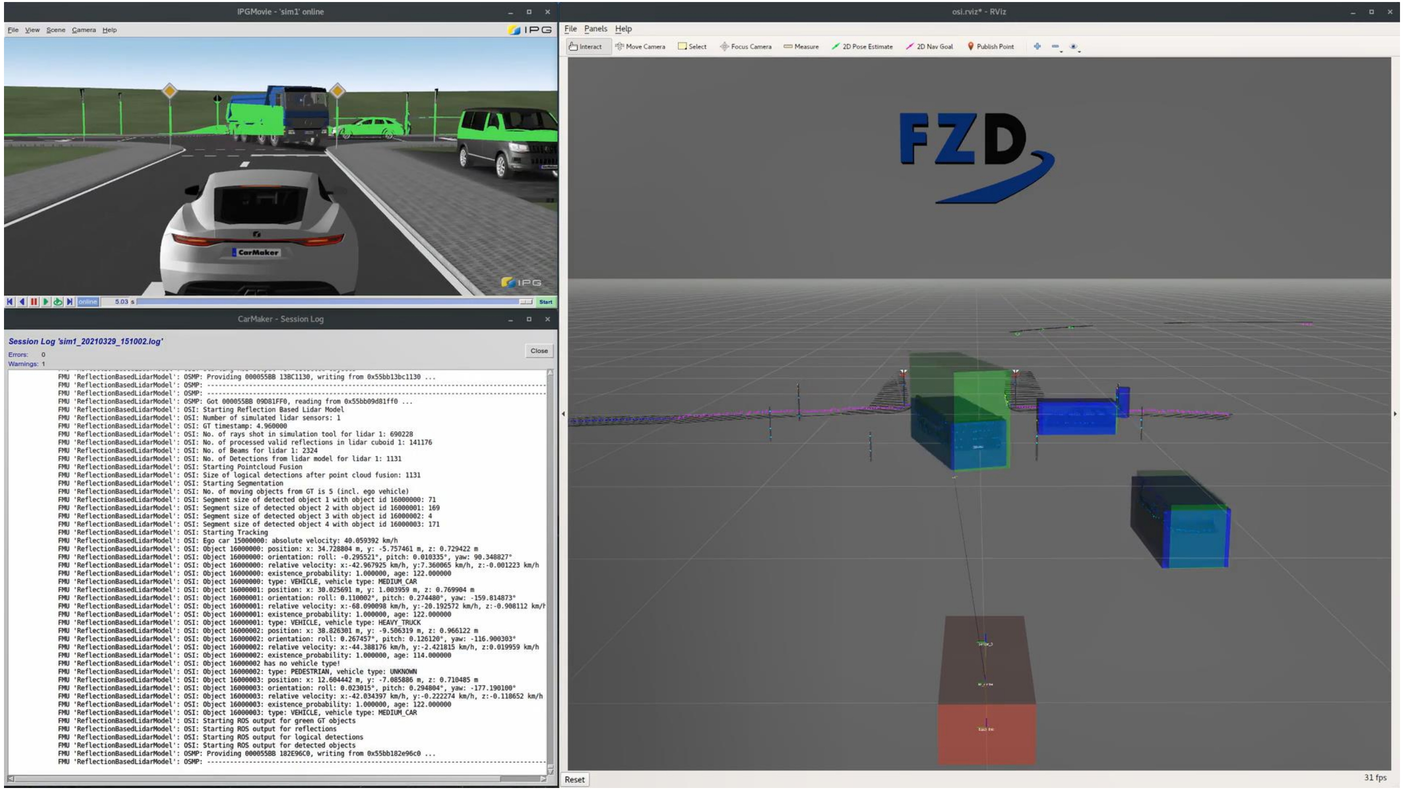
- Intersection over Union (IoU):
  - $IoU(A, B) = \frac{|A \cap B|}{|A \cup B|}$
  - Optimum: 1, minimum: 0



Source: Huch, Sebastian: Entwicklung einer umfassenden Metrik für die Bewertung einer Lidar-Sensor-Simulation durch Betrachtung mehrerer aufeinander folgender Verarbeitungsebenen. Master Thesis, Technische Universität Darmstadt, 2019, p. 47-48

Video for lidar component

Demonstration of results



- Reflections from tool
- Detections from model
- Ego car
- GT objects
- Detected objects

Results for lidar component

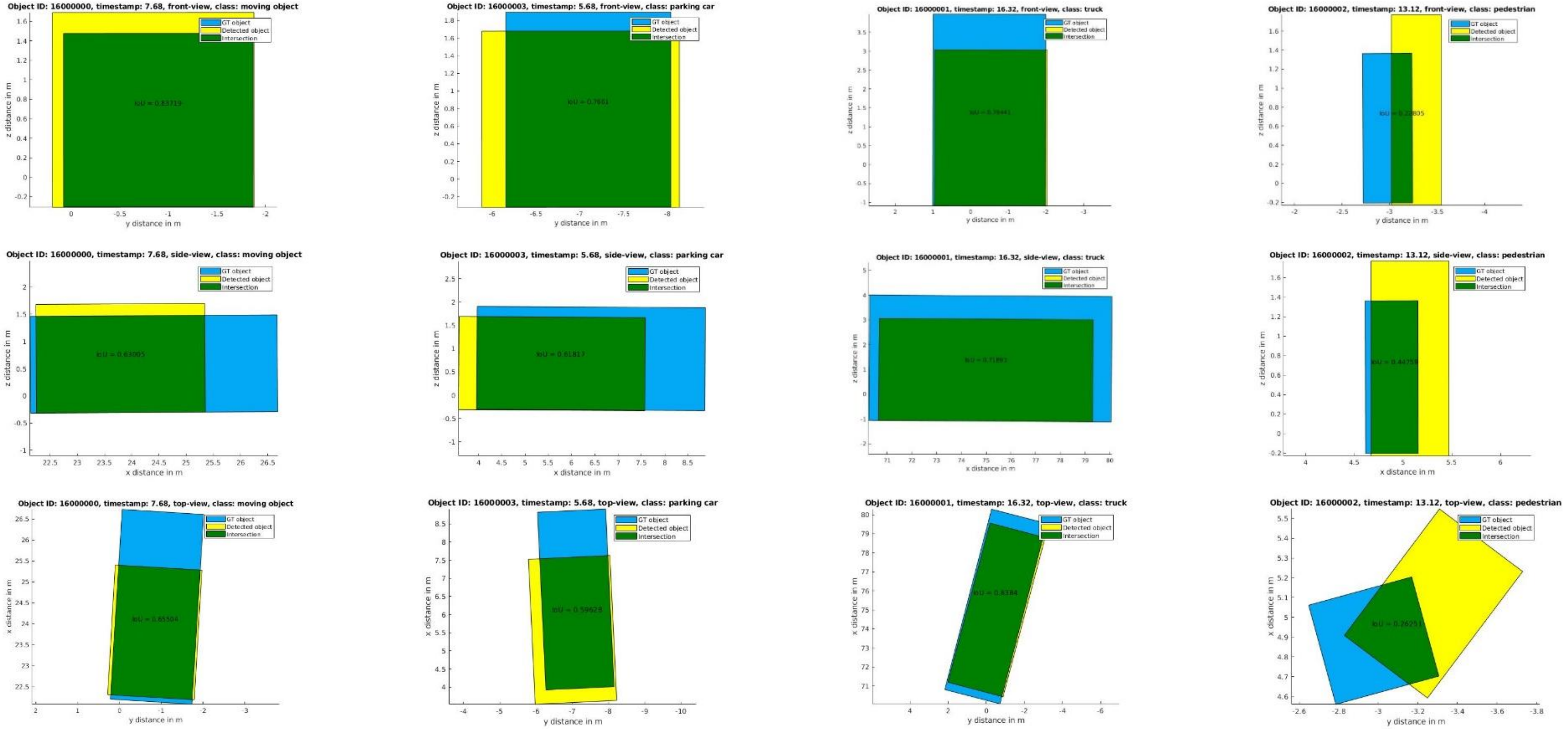
Demonstration of results for single frames

Moving car | Parking car | Truck | Pedestrian

Front-view

Side-view

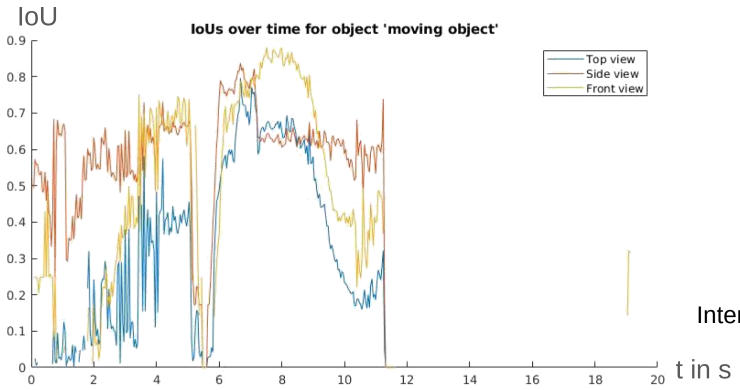
Top-view

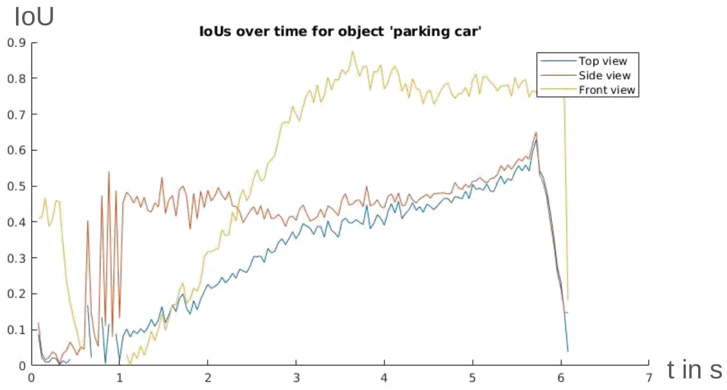


IoU: Intersection over Union, $IoU = \frac{A_\cap}{A_\cup}$, Optimum: 1, Minimum: 0

Results for lidar component

Demonstration of results over time

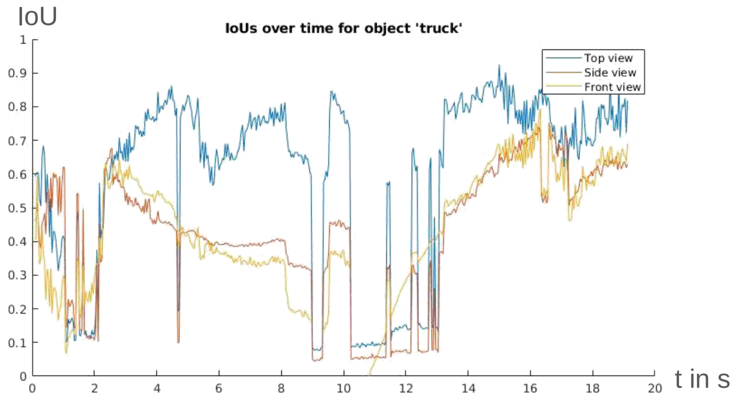


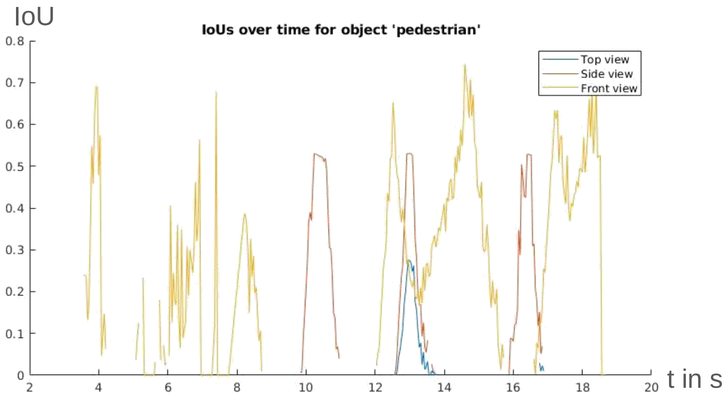


Intersection over Union (IoU)

$$IoU = \frac{A_\cap}{A_\cup}$$

Optimum: 1
Minimum: 0





Summary

Summary and Outlook

- Achievements:
  - Successfully implemented user stories for camera and lidar in two tools
  - Successful usage of standards OSI, OpenDrive and OpenScenario
  - Enabled automatic processing of metrics for sensor technologies
  - Feedback on improvements for OSI implementation and documentation based on findings
- Findings:
  - Lidar specific
    - De- and serialization of simulation takes about 50% of simulation time
    - Shooting of unnecessary rays is slowing down the simulation (Potential of 10 times higher speed)
    - OSI-Logic: rays without reflection are transferred

Outlook

Summary and Outlook

- Extension of sensor technologies by additional radar user story
- Adjusting sensor and car models based on real sensors and cars in VV Methods project (an additional project of the PEGASUS family)
- Checking the possibility to use the open-source approach for material description (OpenMaterial)
- Additional scenarios based on the needs of specific sensor technologies and VV Methods

SET Level

Simulation-based Development and Testing of Automated Driving

Supported by:

Federal Ministry for Economic Affairs and Energy

on the basis of a decision by the German Bundestag

PROJECT of the PEGASUS FAMILY

Thank you

Any question?