Springer Vieweg

ATZ extra

HIGHLY AUTOMATED DRIVING

Game Engines for the Simulation of Sensor Data

ITK ENGINEERING

Simulation of Sensor Data using Game Engines

AUTHOR

Sebastian Wolter, M. Sc.
is Lead Engineer Intelligent
Systems at ITK Engineering GmbH
in Frankfurt

The validation of highly automated vehicles is extremely complex due to the infinite number of sometimes very critical test scenarios. Increasing emphasis is therefore being placed on virtual methods in the early stages of development. ITK Engineering shows how the sensors can be validated in a realistic 3D environment based on established technologies from the gaming industry.

In order to achieve a high level of safety in highly automated driving, real vehicle tests require the extensive support of simulations. Statistical statements derived from a test drive after one million kilometers cannot guarantee comprehensive coverage of all possible driving situations. In addition, real driving cannot be reproduced or controlled and cannot be carried out without risks. Although safety and testability are improved on special test tracks, the number of test scenarios remains limited, and the studies require considerable effort. In addition, data recorded from multisensor systems certainly does not allow closed-loop testing. Data augmentation approaches through a technique known as AI image translation – making changes to real measurement data with the help of artificial intelligence, such as by retrospectively changing the time of day or weather conditions - are currently under development, but are still very difficult to implement for dynamic, interactive scenes [1].

Virtual validation by means of simulations offers a cost-reducing solution that can meet these challenges, is more scalable, and has a truth value that is always available (ground truth). In addition, virtual tests can be integrated into the early stages of development, and multisensor data does not have to be labeled by hand for the fusion of sensor data. The virtual validation requires a highly detailed

3D environment to visualize the test scenarios, as well as a model of the virtual sensors, which can generate sensor data on the basis of the 3D environment. The vehicle's digital twin is thus confronted with critical situations in the simulated 3D environment. In order to generate synthetic data, the vehicle moves (using a driving dynamics model) in the high-resolution virtual environment and interacts with other dynamic and static objects such as pedestrians. To do this, a 3D scene must be created and the behavior of the dynamically moving objects modeled. The simulated lighting is essential for the processing of virtual camera data and must follow physical principles. For other sensor simulations, geometries and material properties have to be reproduced in detail, such as for vehicles, asphalt, or curbs. The sensor models used should be both realistic and rapidly computable (as close to real-time as possible) so that they can be used in software or hardware-in-the-loop test environments. If the data generated in this way flows from the processing logic back to the simulator (closed loop), driving situations and the reactions of a dynamically acting system can be tested completely virtually. If this feedback does not exist (open loop), only individual states and decisions can be tested. The following applies to both variants: the more realistic the input data, the greater the confidence that the system will also function reliably in reality.

REALISTIC VALIDATION USING GAMING TECHNIQUES

Established technologies from the gaming industry are used to validate sensors in a geometrically realistic environment. One such game engine (Unity3D or Unreal Engine) forms the basis of the 3D representation for this. One of its strengths is the visualization of thousands of high-resolution 3D data contents. The integrated 3D editor and compatibility with a wide variety of file formats allow for rapid creation of 3D scenes and integration of 3D models via „drag and drop" from various sources, so-called asset marketplaces, **FIGURE 1** and **FIGURE 2**. Multiple virtual cameras can also be positioned and parameterized. The technique of physically based rendering [2] is used for the most photorealistic image representation possible.

FIGURE 1 Simulation of highway driving at night in heavy traffic (© ITK Engineering | Unity Asset Store)

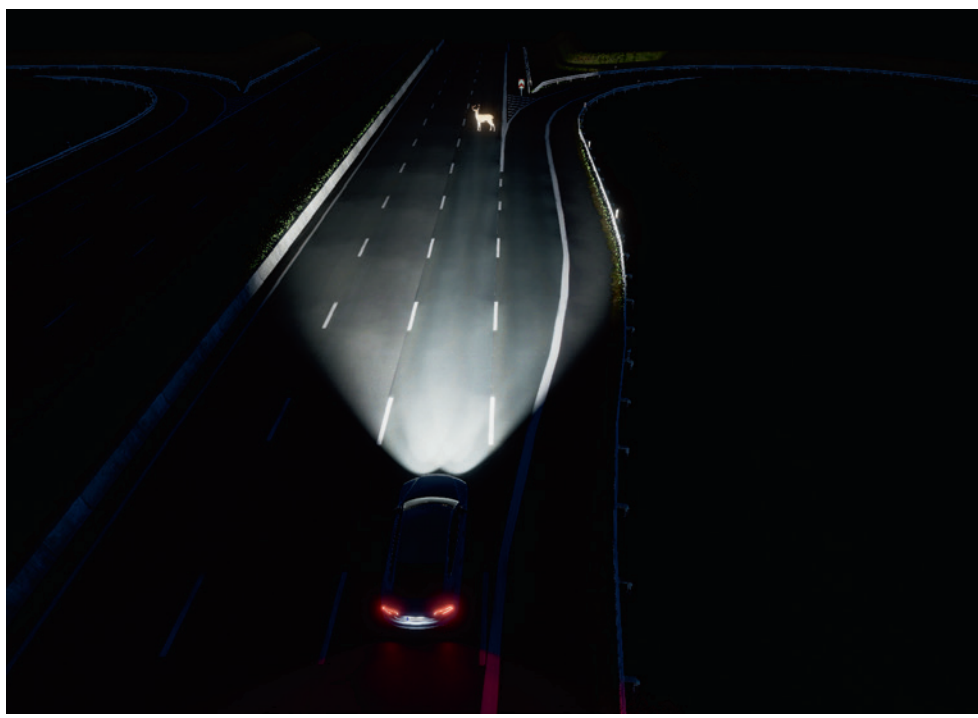


FIGURE 2 Simulation of highway driving at night with deer crossing (© ITK Engineering | Unity Asset Store)

More recently, ray tracing methods are also increasingly being used to enhance the graphic quality. Lens deformations and depth of field are among the effects that can be integrated during post-processing, but custom effects can also be developed. This requires expertise in the field known as shader programming. This establishes a good basis for the realistic simulation of camera data [3].

In contrast to the camera, sensors that operate with lidar, ultrasound, and radar require separate sensor modeling. The physics simulation built into the game engine can support simple sensor modeling to simulate simple distance sensors. It can also be used to model phenomenological models that augment ground truth data with known phenomena. Although these can be computed quickly, objects are only represented approximately by simple bounding volumes and static objects such as a guardrail in a curve are only simplified or not taken into account at all. At the other extreme, calculations of wave propagation using the finite element method (FEM) are physically very precise, but not real-time capable. These are not used in dynamic and complex environments due to the high memory requirements for spatial discretization, but rather to investigate local phenomena or different construction methods and installation effects.

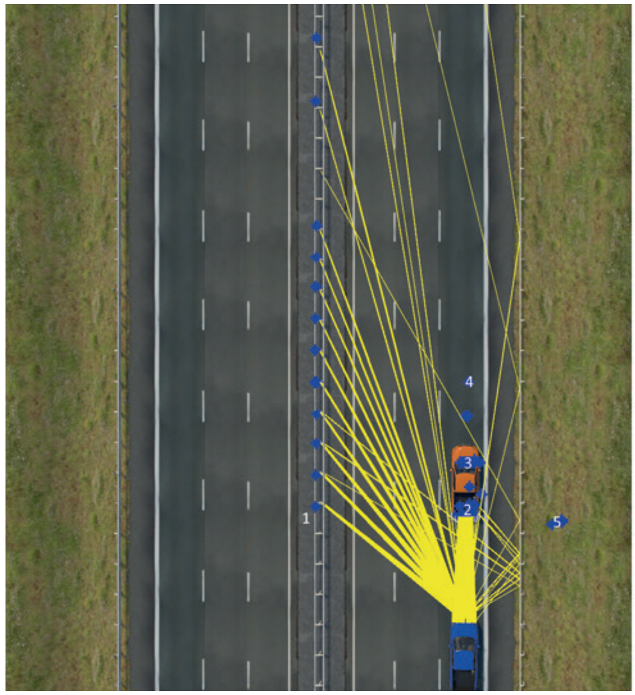
REAL-TIME, CONSISTENT MULTISENSOR DATA GENERATION

Physical models based on the process of ray tracing offer a compromise between computing time and accuracy. These allow the propagation of a wave to be tracked in the scene in real time by using multiple rays, since in such models the wave propagation is mapped physically and directly. The rays are emitted by the virtual sensors into the simulated 3D world, where they then interact with surfaces through reflection, scattering, or transmission. Using a physical model, physical quantities such as relative Doppler velocity, intensity, or sound pressure, are associated with each ray path and updated during the simulation. This process continues until the ray hits a receiving sensor. Here the data that is received is merged, processed further via post-processing modules, and transferred to components of the signal processing chain via external interfaces. The focus here is on the generation of raw data so that any desired perception algorithm can subsequently be run on this data.

In order to achieve real-time capability, the game engine, ray tracing, and post-processing are accelerated by parallelization on graphics cards. This approach benefits from constant innovations in the field of graphics hardware as well as from visual effects and the increasing performance of the game engine software.

PHYSICAL RADAR SENSOR MODEL

When using the ray tracing process to generate raw data for a radar signal, the simulation approximates the radar waves as a large number of individual rays.



1 Reflection on guardrail posts
2 Reflection on the rear of the vehicle
3 Reflection on the front of the vehicle
4 Multiple reflections
5 Ghost target/mirror reflection on guardrail

FIGURE 3 Simulation of a radar sensor (© ITK Engineering I Unity Asset Store)

Based on the sensor field of view, thousands of these are emitted into the 3D scene, **FIGURE 3**. The power of the signal received at the radar sensor depends, among other things, on the antenna characteristics of the sensor being modeled. The energy loss due to the spherical expansion is taken into account according to the transit time of the different ray paths. The angle of incidence and the frequency of the carrier wave play a major role in the material interaction. Ambient and atmospheric parameters and polarization effects are also represented. The total available 3D geometry is taken into account for the multipath propagation. Potential ghost targets through reflections from the ground, guardrail, or walls result automatically from the method used, which is a great advantage over phenomenological models. (Hybrid methods from phenomenological and physical ray tracing methods are also possible). The relative Doppler velocity is also resolved at the micro-Doppler level. This means that moving objects have different relative speeds depending on the point of interaction. An example of this is pedestrians who have different speeds due to their arm and leg movements than those resulting purely from body movement. This is important for the classification and motion approximation of the objects and must therefore also be represented in the simulator. Multiple reflections, i.e. multiple traversal of the same ray path on one and the same object, are also resolved directly by the ray tracing process. The modeling of these effects is essential in being able to generate realistic sensor data, **FIGURE 4**.

DIVERSE APPLICATIONS

The simulation method can be applied in the development process throughout the entire V-model. As early as the design phase, during evaluation of different locations for the installation of sensors, ray tracing can be used to identify problems and determine favorable positions, such as for a radar-based turning assistance system. Several sensors can easily be tried out and relocated to ensure constant visibility of objects around the vehicle. Since a mix of different sensor systems and sensor technologies is particularly necessary in highly automated vehicles, this method can help to determine an optimized sensor layout.

In addition, data are required repeatedly during the development of the algorithm before real measurements can be carried out and labeled. For example, when developing object recognition for e-scooters, cyclists, and pedestrians, micro-Doppler sensor data can be made

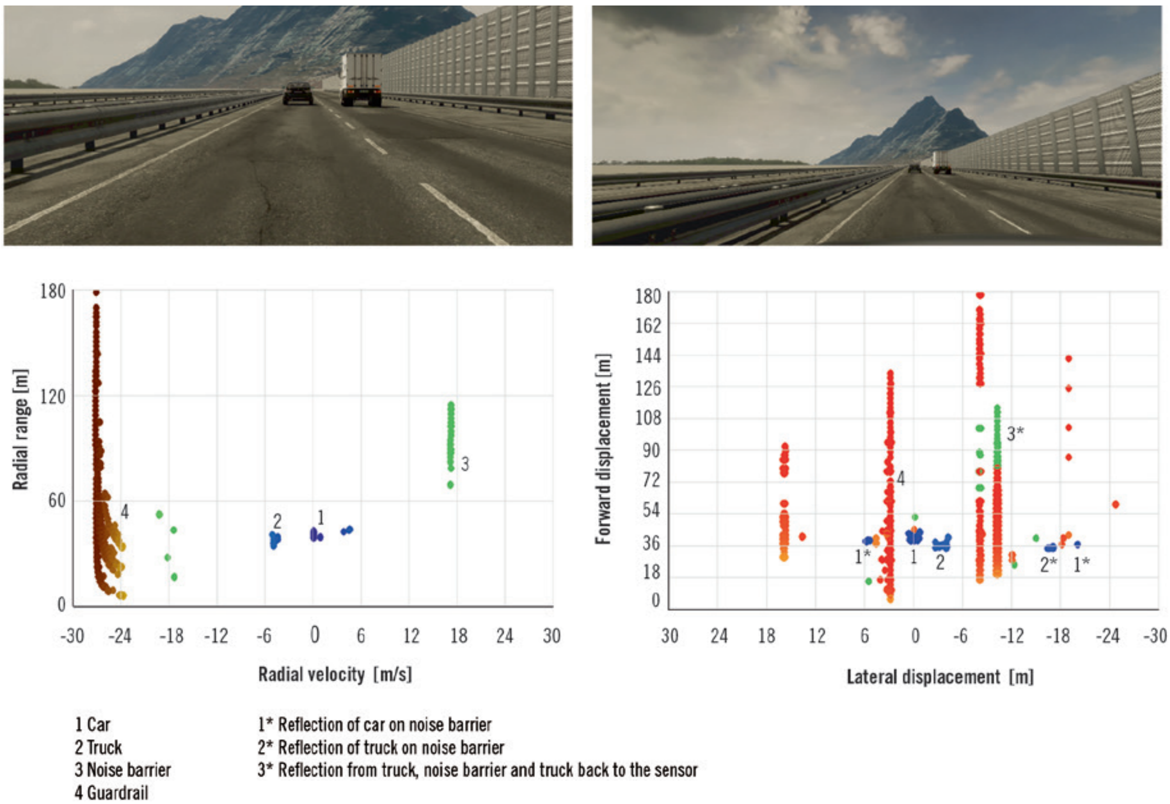


FIGURE 4 Example simulation of highway driving: Radar range Doppler (left) and position plan (right) (© ITK Engineering | Unity Asset Store)

available for perception purposes. Consistent data can also be generated for the development of sensor fusion algorithms, as is normally provided by multisensor systems.

In terms of validation, this methodology supplies data at various levels in the area of X-in-the-loop test procedures. Ground truth and object data can be used for functional tests, object recognition requires pixel data or point clouds, other perception algorithms require environmental and functional data, up to and including the signals of certain sensor types. In addition, this method allows the test scenarios to be changed automatically in order to efficiently identify borderline cases from a large number of procedural scenario variants.

CONCLUSION AND PROSPECTS

Gaming methods can be used to create custom, realistic 3D environments for simulating and creating patterns in a rapid prototype process with qualitative, synthetically generated sensor data. Physical sensor models fill the gap between classical numerical methods and phenomenological models. By using game engines, there is the potential to benefit directly from future hardware and software developments. Scalable tests can be carried out virtually at an early stage of development to complement real vehicle tests, thus enabling the development of safe, highly automated driving functions.

REFERENCES

[1] Liu, M.-Y.; Breuel, T.; Kautz, J.: Unsupervised Image-to-Image Translation Networks. Neural Information Processing Systems annual meeting (NIPS), 2017

[2] Pharr, M.; Jakob, W.; Humphreys, G.: Physically Based Rendering: From Theory to Implementation. Morgan Kaufmann: Burlington, 2016

[3] Mayer, N.; Ilg, E.; Fischer, P.; Hazirbas, C.; Cremers, D.; Dosovitskiy, A.; Brox, T.: What Makes Good Synthetic Training Data for Learning Disparity and Optical Flow Estimation? In: International Journal of Computer Vision 2018, No. 126, pp. 942-960

IMPRINT:
Special Edition 2021 in cooperation with ITK Engineering GmbH,
Lochhamer Straße 15, 82152 Martinsried;
Springer Fachmedien Wiesbaden GmbH,
Postfach 1546, 65173 Wiesbaden,
Amtsgericht Wiesbaden, HRB 9754, USt-IdNr. DE811484199

MANAGING DIRECTORS:
Stefanie Burgmaier | Andreas Funk | Joachim Krieger

PROJECT MANAGEMENT: Anja Trabusch

COVER PHOTO: © hutterstock/metamorworks

The original German version of this article appears in „ATZextra Automotive Engineering Partners“ in June 2021

ITK Engineering

Stability, reliability and methodological expertise – this is what we have stood for since our founding in 1994. At all times, our customers have benefitted from our dedicated multi-industry know-how, especially in the fields of control systems design and model-based design. Customers can count on us – from conception through to deployment, we cover the entire development process.

Our areas of expertise include:

- Software development
- Hardware development
- Electrical & electronic systems
- System integration
- Software as a product
- Turnkey systems
- Customer specific development
- Technical consulting
- Seminars
- Quality assurance

The satisfaction of each of our partners and mutually respectful cooperation shape our corporate philosophy, in which four values are firmly anchored: Read more about this on the web.

V.1.0.0_e_2021

ITK Engineering GmbH
Headquarters: Ruelzheim
Im Speyerer Tal 6
76761 Ruelzheim, Germany
T: + 49 (0)7272 7703-0
F: + 49 (0)7272 7703-100
info@itk-engineering.com

www.itk-engineering.com
www.itk-career.com

Founded in 1994
Branch offices throughout Germany – ITK companies worldwide.

Follow us on: