

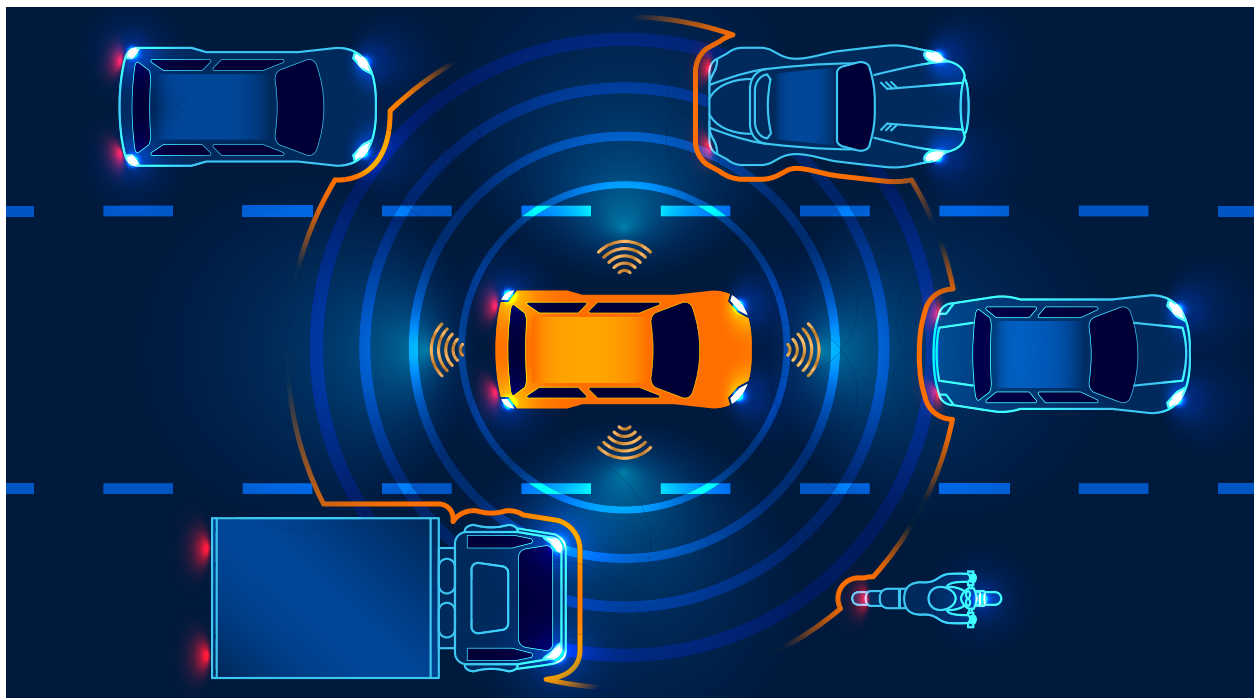
Evolving Your ADAS and AV Tests with Emulation Capability

The Evolution of Radar Sensors: A Critical Part of ADAS / AV Systems

Creating safe and robust autonomous driving (AD) systems is a complex task. Automakers must overcome immediate challenges to realize the future of autonomous mobility.

Autonomous vehicles (AVs) have hundreds of sensors, all of which need to work with one another inside the car and with other smart vehicles. The software algorithms enabling autonomous driving features will ultimately need to synthesize all the information collected from these sensors to ensure that the vehicle responds appropriately. These algorithms require testing against millions of complex scenes covering various driving scenarios. Automakers need to be able to sign off on new advanced driver-assistance systems (ADAS) and AV functionality confidently.

Achieving the next level of vehicle autonomy will require many innovations and technological advancements. Continuous investments in sensor technologies such as radar, lidar, and cameras will improve environmental scanning. Each sensor type has its advantages and disadvantages, and they need to complement each other to ensure that the object detection process has built-in redundancy.



Recognizing the Complexity of Radar Sensor Test

Powerful software algorithms are necessary to combine and carry a large amount of high-resolution sensor data, including vehicle-to-everything communication inputs. Machine learning is the established method for training self-improving algorithms. Automotive original equipment manufacturers (OEMs) apply the algorithms to enhance decision-making in complex traffic situations.

Validating these algorithms with the most realistic stimuli available, in a repeatable and controlled fashion in the lab, is crucial for their accuracy and safe deployment. Automotive design and test engineers identify the scenarios for testing, rendering them from 3D simulation environments to real radar signal input to the radar modules before road-testing them.

Building and updating an accurate 3D map of the world that the vehicle senses, and interpreting it to the electronic control unit, are vital steps for autonomous driving. Although radar technology has been around for a long time, new radar sensor front ends are evolving and gaining complexity with the following features:

- wider bandwidth, from 76 to 77 GHz and now 77 to 81 GHz
- improved accuracy that comes from exploiting multipath propagation through multiple-input, multiple-output (MIMO), using multiple transmission and receiving antennas
- higher resolution with the fourth dimension of height, 4D imaging radar

Older 2D and 3D radars do not perceive the height dimension because of their limited capabilities and lack of large MIMO. Newer generations of radar sensors that include wide antenna apertures using large MIMO structures unlock the fourth sensing dimension — height. These radar sensors offer better perception capabilities with increased resolution and bandwidth to 4 GHz. With that, automotive test solutions must leapfrog the existing sensing technology and offer better capabilities so that the measurement equipment won't limit testing.

Validating these algorithms with the most realistic stimuli available, in a repeatable and controlled fashion in the lab, is crucial to ensure accuracy and safety once deployed.

Behind the sensor, the detection algorithms also deal with increased complexity, resulting in more stringent requirements for testing and validation of the detection algorithms.

Training these sensors requires more than just point targets. On a real road, radar sensors must be able to differentiate between another car, a truck, a bicycle, and a pedestrian. The process of identifying and classifying objects is crucial as it affects the reaction of the vehicle and the safety of the passengers. This is where current in-lab solutions fall short. Robotic automation of up to eight moving targets with just one point per target does not provide enough detail to help the vehicle learn how to classify these different objects.

Reimagine Test Tactics

R&D engineers use the DevOps (development and operations) model across software application development cycles, from development and test to deployment and back to development. The cycle continues in a loop, where R&D engineers collect the feedback and improve the product with every iteration. The DevOps model is common in software industries, and automotive companies are beginning to use this process as vehicles become more software-based. The next section breaks down the DevOps model into different iterations: simulation, emulation, and deployment.

Simulation

A simulator creates an environment that mimics the behavior and configurations of an actual device. Automotive companies spend significant time on sensors and control modules, simulating environments with software-in-the-loop testing. Auto developers will integrate, tweak, and then loop the tests again, as shown in Figure 1.

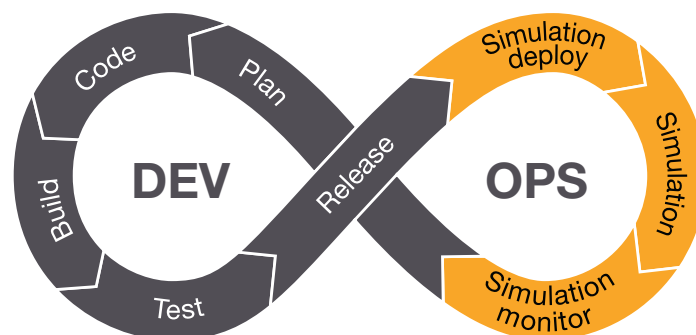


Figure 1. Simulation: Example of software-in-the-loop test, integrating, tweaking, and looping the results

Emulation

Once you can simulate something entirely in software, the next step is to add some hardware reality to it. As such, an emulator can duplicate the hardware and software features of a real device. When building complex machines, one vital step to ensuring that the machine is safe and reliable is to re-create a part of the system in the lab, with actual components, placed in a hardware-in-the-loop setup. This step bridges the gap between simulation and road testing to save time and cost. This is becoming more important to AD and ADAS system development as algorithm complexity rises.

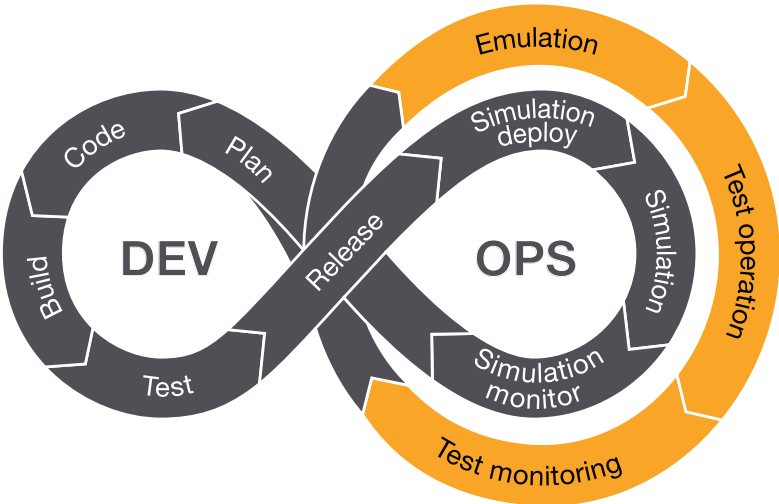


Figure 2. Emulation: Hardware-in-the-loop test process with real devices

Testing on open roads

Road testing or test track drives with a complete vehicle require an integrated system in a prototype or road-legal vehicle. This testing allows OEMs to validate the final product before bringing it to the market. Road tests or test track drives are risky and expensive. Although design engineers have the chance to update the software, it is challenging to update the system-level design, and going back to the drawing board would prolong the development time.

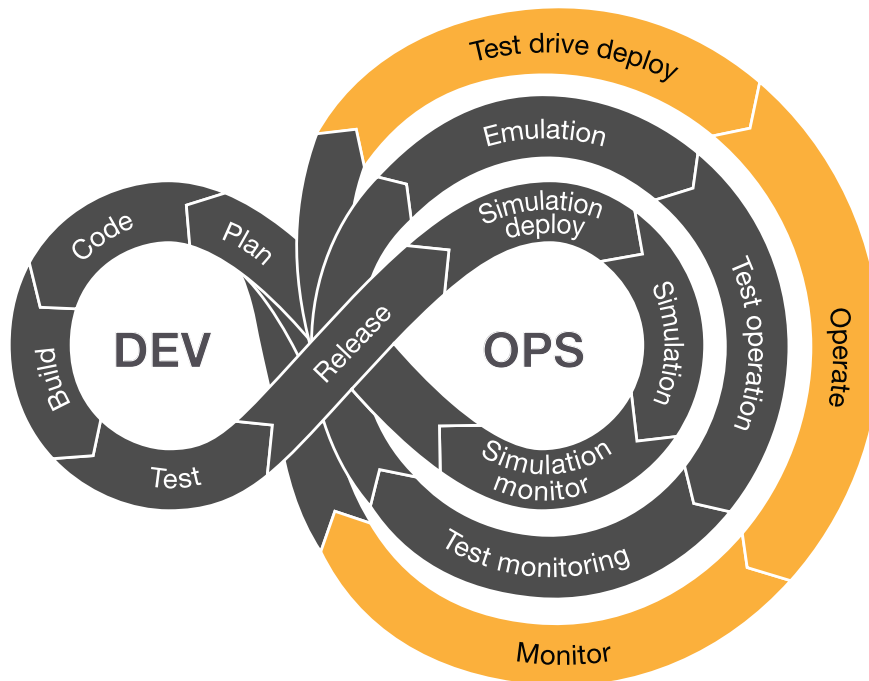


Figure 3. Road test: Full vehicle test, complete with integrated systems

Shift in thinking

In Figure 3, the result from simulation feeds on to emulation and then to road testing. Each loop builds on the previous one. Design and test engineers employ this test process with fundamental components replacing parts of the simulation, such as a brake or a steering wheel subsystem. Implementing this test process on radar systems in the past was not easy. There was no way to emulate a near-real-life scenario with sufficient details to help the radar algorithms learn and prepare for the open road. This is not problematic for simple scenarios, but it is critical for complex corner-case situations that are impossible to test on an open road.

Unlocking Level 3+¹ autonomous driving means that the AD functions are responsible for more than driver, passenger, and pedestrian safety. They oversee every maneuver the vehicle makes on the road, with potential legal implications.

Automotive OEMs need to be careful when developing driving functions, especially fully automated functions. Emulating real-world scenes in the lab, with real radar sensors and signals, takes ADAS and AV tests to a new level.

Setting Up Your AV for Success

Software is driving vehicle development trends and topics, such as autonomous driving and electrification. The focus of vehicle development is therefore shifting from hardware to software. Vehicles with ADAS Level 3 and beyond require testing and validation against the growing number of scenarios and the surrounding environment. Not only will the number of tests increase, but the complexity of the tests will also increase.

For example, with adaptive cruise control, it was sufficient to pay attention to the vehicle up front. Today's system-level tests should also consider various road users. One example is highway driving. In addition to following the lead vehicle and keeping a safe distance, the test should also consider automated maneuvers such as exiting a lane, passing, and re-entering the lane. Complexity increases further with city driving. Think about intersection and turning scenarios involving pedestrians, cyclists, and e-scooters. A real test drive on its own cannot represent this level of complexity and variability. Therefore, simulation is essential to developing and validating AV systems.

¹ Level 3+ defined as conditional automation, high automation, and full automation by the [Society of Automotive Engineers](#).

NCAP provides standardized scenarios

New Car Assessment Program (NCAP) tests started as a goal for OEMs to ensure a common safety standard for drivers, passengers, and third parties. It is a voluntary safety rating system, but it gained popularity fast and became a recognized benchmarking mechanism on the consumer side and a handy yet sometimes challenging sales and marketing strategy for OEMs. Automotive OEMs often perform crash tests with dummies in a controlled environment. They strive to hit the five-star safety rating for branding and commercial messaging purposes.

The push toward more vehicle autonomy has added complexity to crash tests. A seat belt pretensioner, a side head airbag, or a child restraint system would suffice in the past. But now, the test must cover the vehicle’s ability to autonomously break when detecting an object on the road at a certain distance — whether it is a vulnerable road user (VRU) such as a pedestrian or a cyclist or another vehicle cutting in at a lower speed.

Table 1 captures possible scenarios that testing needs to cover, ranging from NCAP and US National Highway Traffic Safety Administration (NHTSA) ² test cases to common radar-based ADAS / AD features and functions. This list is neither extensive nor exhaustive; the library of situations that require testing could be the topic of a whole second paper.

Table 1. Testing scenarios

NCAP / NHTSA test protocols	Common radar-based autonomous vehicle and advanced driver assistance systems
<p>Safety assist</p> <ul style="list-style-type: none"> • autonomous emergency braking (AEB) car to car • lane support systems (LSS) 	<p>Autonomous vehicle systems</p> <ul style="list-style-type: none"> • active cruise control (ACC) • lane-keeping assistance (LKA) • evasive steering assist (ESA)
<p>Vulnerable Road User (VRU) Protection</p> <ul style="list-style-type: none"> • AEB pedestrian • AEB cyclist 	<p>Advanced driver assistance systems</p> <ul style="list-style-type: none"> • blind-spot detection (BSD) • cross-traffic alert (CTA) • forward collision warning (FCW) • lane departure warning (LDW) • rear automatic braking (RAB) • rear collision warning (RCW)

² NCAP and NHTSA are regional organizations. Other local and regional organizations govern these standards.

A thought experiment among autonomous driving artificial intelligence (AI) developers is the choice between saving the lives of a vehicle's passengers and saving pedestrians in case of a collision. MIT researchers conducted extensive **ethical studies** to determine possible human biases that developers might build into an AI algorithm. The goal is for the AVs to navigate such scenarios in a faster, more rational, and more informed way than their human counterparts.

Because of the requirement to reproduce the scene precisely, NCAP scenarios lend themselves to lab testing. With increased pressure on automotive OEMs to get to market faster, products must work earlier in the design cycle.

Automotive OEMs recognized that supplementing lab tests early with NCAP scenarios, from component level to system level and throughout the design cycle, would save time and money. With scene emulation capability, automotive OEMs can verify their radar integration earlier in the lab and be better prepared for open road testing or testing on accredited tracks in the integration stage. The process helps reduce the risk of test failures significantly.



NCAP Scenario 1: Autonomous emergency braking

For example, we will look at one NCAP scenario, autonomous emergency braking (AEB). Rear-end collisions are the most frequent collisions on open roads, according to the Euro NCAP [website](#). They occur when distracted drivers in the rear vehicle fail to notice that automobiles ahead of them have slowed down or stopped and collide into them.

Testing of AEB car-to-car systems spans a wide range of speeds, vehicles, and traffic situations.

FCW and AEB systems aim to prevent or mitigate the potential damage. These systems notify the driver of a hazardous situation ahead or automatically apply the brakes if the driver fails to see the hazard. Because of the urgent nature of the potential hazard, along with the proximity of the road obstacle in question, line-of-sight (LOS) sensors are best suited to detect these situations. To that end, automotive manufacturers use cameras, lidar, or radar sensors — individually or in concert (sensor fusion) — with different degrees of performance fitting different situations.

Emulating this in a lab for the sensors requires a minimum of one target — just the car in front. That is pretty easy to do with a radar target simulator today. However, that is testing an ideal but unrealistic scenario. In a real roadway, there are guard rails, reflections from signs, and other cars. What if the radar does not interpret those correctly?

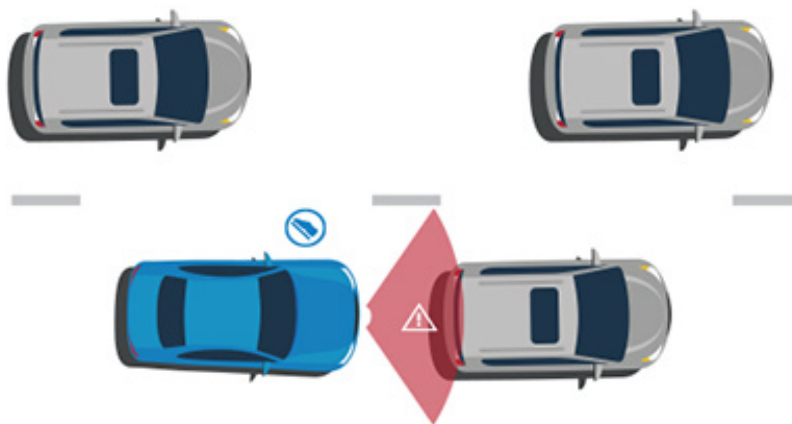


Figure 4. Autonomous emergency braking diagram (Image courtesy of NHTSA website)

AEB scenario experiment

AVSimulation's SCANer, which has a library of NCAP scenarios is an AEB car-to-car rear moving (CCRM) NCAP scenario. The viewpoint is from the ego vehicle or the vehicle that has the radar under test.

Figure 6 and 7 show a top view of the XY horizontal plane of the road, as depicted in the SCANer simulation. This tool includes both the expected and the detected radar targets and provides the user with a real-time validation of the radar-based ADAS / AD algorithm's reaction to the scenario. The red cone indicates long-range radar, and the blue cone indicates short-range radar. Note the similarity of the images, including the guardrails.



Figure 5. NCAP scenario using AVSimulation SCANer software

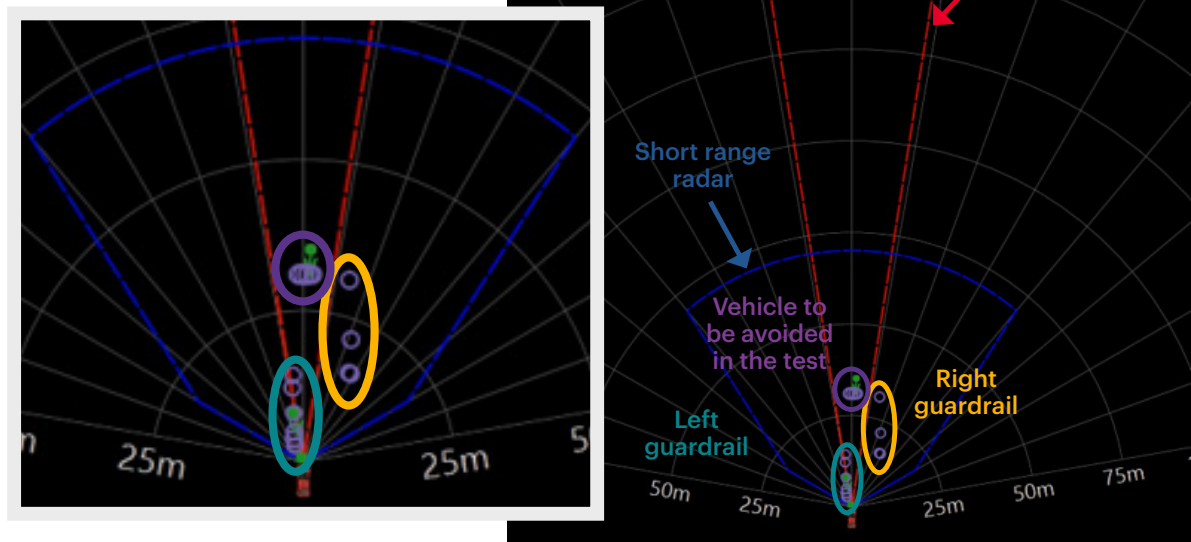


Figure 6. Here is the same scene as shown in the simulation, translated to a radar image. Open circles are emulations, and green arrows are detected objects

Today, a simple RTS system would be able to emulate a 2-car scenario. However, adding reflections of the guardrails creates a real-world scene to ensure a more accurate validation of AV systems.

NCAP Scenario 2: Vulnerable road users

Another critical area to test, as called out by NCAP, is VRUs. In addition to assessing how well cars protect their occupants, NCAP tests assess how well they protect VRUs — pedestrians and cyclists — with whom vehicles might collide. Like AEB above, the idea is to notify the driver of an impending collision and provide autonomous braking if the driver does not respond quickly enough.

These tests enable automotive OEMs to assess injury risk. The focus is on the vehicle's ability to protect pedestrians and cyclists rather than the driver and passengers. As with all the NCAP tests, there is a rating system. Cars that perform well can gain additional points if they have an AEB system, which recognizes pedestrians and cyclists.

Figure 7 indicates a two-object scenario that could emulate the AEB system test. However, what if a pedestrian or cyclist comes out from between parked cars? Or if several pedestrians are crossing or running back and forth? This also points out a weakness of many target simulation systems: not being able to emulate an object close to the vehicle and its radar under test.

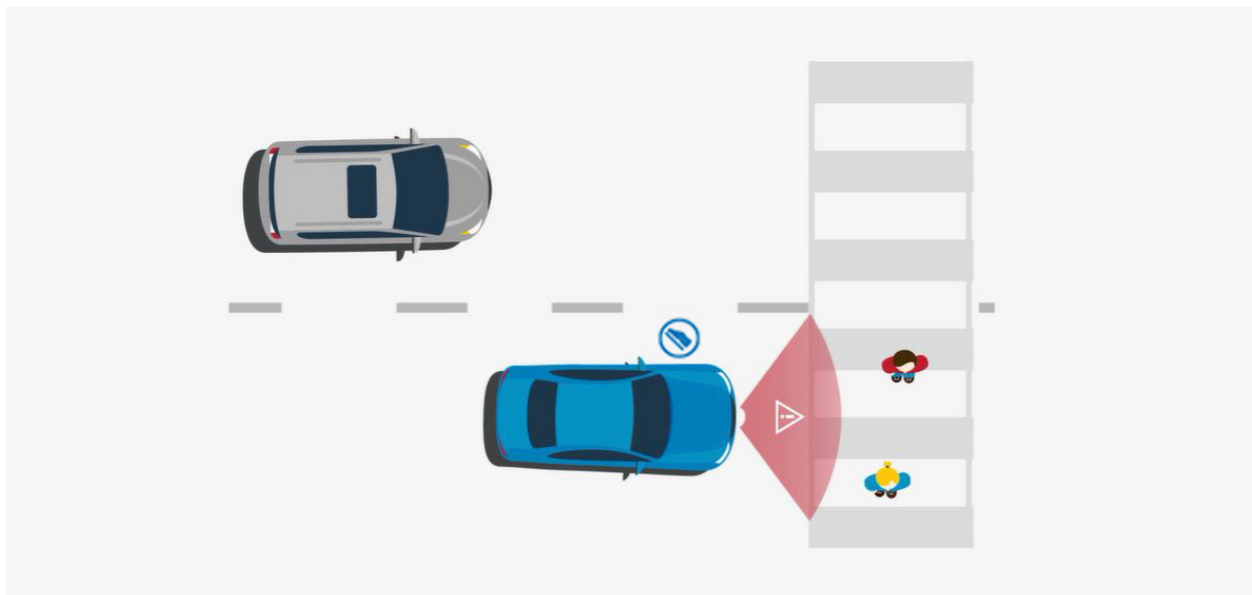


Figure 7. NHTSA diagram of VRU NCAP scenario (Image courtesy of NHTSA website)

This scenario is important as there are many other similar situations where resolution and close distance can make a difference between life and death. Envision a scenario where the test vehicle is approaching an intersection, intending to turn right (in a region where a right turn on red is legal), after thoroughly checking for pedestrians and oncoming automobiles. As the vehicle slowly approaches, its camera, long-range radar, and lidar sensors are of little use. Instead, it must employ its lateral short-range radar. This radar has a wide field of view and a high resolution, and it must be able to detect objects that are critically close to the vehicle.

Suppose the pedestrian crossing the street happens to be a mother pushing a baby stroller. In that case, the radar must detect both the stroller and the mother and stop the vehicle on time, even if the distance between the stroller and the vehicle is less than 4 m — which, in a real scenario, it would be.



VRU scenario experiment

This experiment involves IPG Automotive's CarMaker and one of its NCAP scenario packages for VRUs. In this scenario, a pedestrian runs across the road from between two parked cars from right to left, to the front of the ego vehicle, as shown in Figure 8. Again, the viewpoint is from the ego vehicle or the vehicle that has the radar under test. The ego vehicle should autonomously apply its brakes.

Similar to the first experiment, the Figure 9 is a view of both the expected and detected radar targets and provides the user with a real-time validation of the radar-based ADAS / AV algorithm's reaction to the scenario. Again, note the similarity of the images, including the details of parked cars and extra pedestrians.

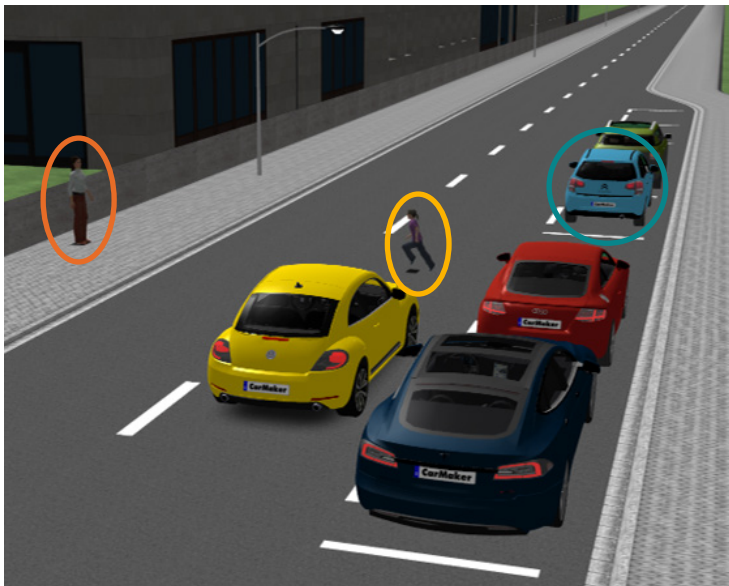


Figure 8. The pedestrian is less than 4 meters in front of the radar on the vehicle under test (the yellow car)

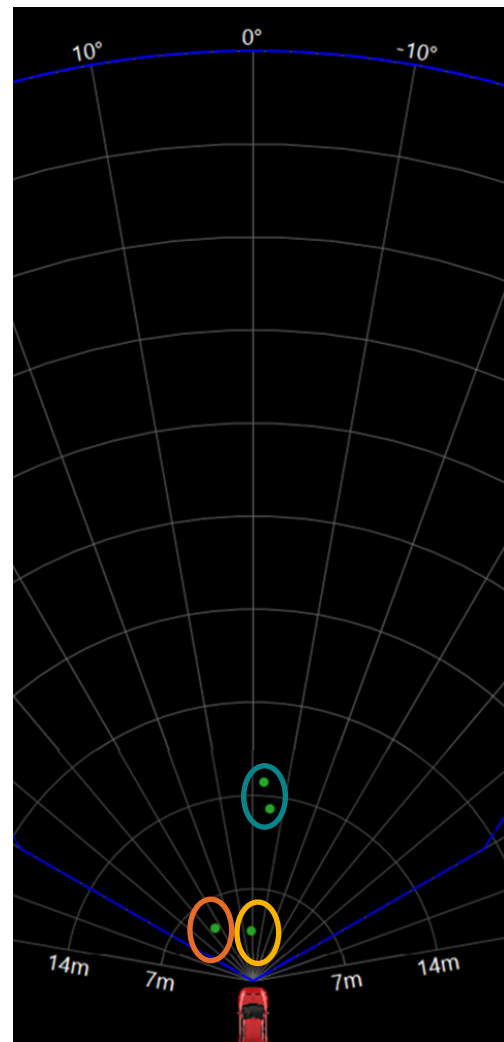


Figure 9. A radar image showing the 3D modeling of VRU scenario. It is a representation of the vehicles on the side of the road, the pedestrian in the middle, and the bystander on the side of the road — in green dots

Today, with a simple 2 or even a 4-point radar target simulation system automotive OEMs would not be able to add the complexity of multiple pedestrians with multiple parked cars.

Closing the Technology Gaps with Innovations

To emulate the examples in a lab, automotive OEMs need to translate the output of simulation software into real signals to stimulate the radar modules.

How does it work?

The following technology concepts explain how automotive OEMs could emulate the scenes for lab testing.

Point clouds

Point clouds describe a data set of points representing objects or sets of objects. The points that are coordinates from the x-, y-, and z-axes enable a large amount of spatial information to be in one set. 3D laser scanners, lidar, and radar technologies often produce and reference point clouds. For the sake of this paper, the point cloud is coming from the 3D scenario simulator.

Point clouds add details to the scene and ensure that the algorithm you are testing can distinguish between two objects that are close together. While a traditional radar target simulator (RTS) will return one reflection, independent of distance, radar scene emulation increases the number of reflections as the target gets closer. This type of dynamic resolution varies the number of points that represent an object as a function of distance.

To display the point clouds, the setup needs to fulfill two hardware aspects:

- ray tracing
- wall of pixels

What is a rixel?

Rixels are RF transceivers small enough to fit into a chip-sized unit. Each one is like a pixel on a TV screen.

By putting eight of them on one board, and stacking multiple boards next to each other, the matrix of rixels creates a high-resolution wall.

This is analogous to a high-definition screen with pixels that display different colors and brightness. Similarly, rixels “display” distance, velocity, and object size.

Ray tracing

Ray tracing technology extracts the required information for sensors, such as radar or cameras, during testing (Figure 10).

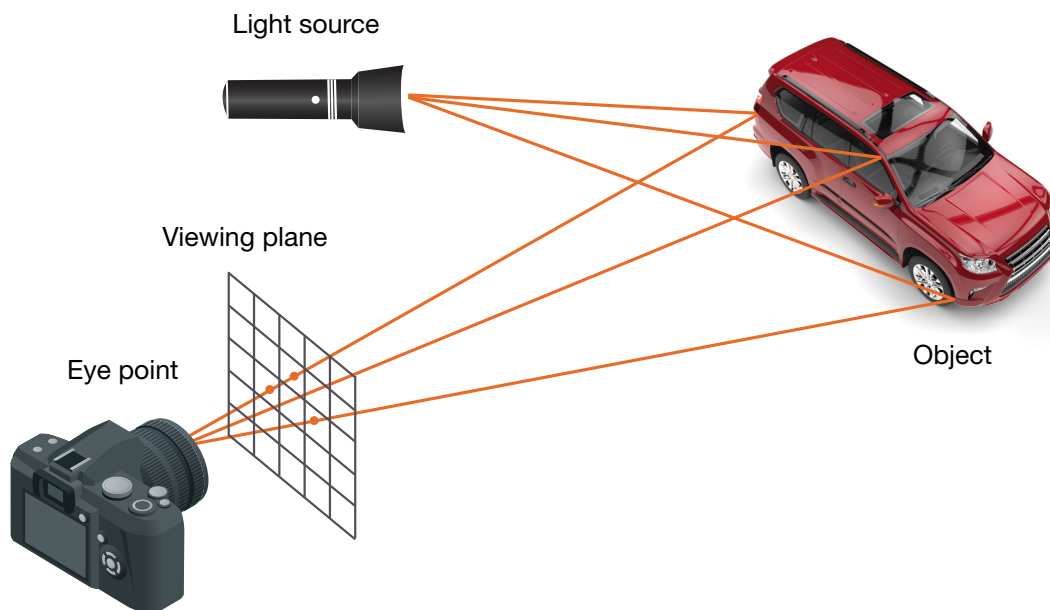


Figure 10. Ray tracing technology helps extract required information for sensors in an autonomous vehicle

The concept of ray tracing goes back to computer graphics and the ability to render digital 3D objects on 2D screens by simulating the physical behavior of light. Placing objects in a stimulus-response system renders them visible. In this diagram, the light source illuminates the object, with light rays reflected and scattered in multiple directions. Only those that converge into the user's eye point are mapped on the viewing plane. The screen's resolution is given by the viewing plane's characteristics and the object's finer or coarser meshing. The object rendering includes material properties and other relevant information, such as object color and brightness.

Although this explanation is specific to the visible spectrum of light, the same principle applies to any stimulus-response rendering algorithm based on LOS radiation — for example, radar vision. The light source is the radar transmitter, the relevant material properties concern radar radiation reflectivity, and spatial velocity translates into Doppler effects.

Wall of rixels

To translate the information extracted from ray tracing to something that a radar sensor could detect, we have

- created each vehicle as an object
- assigned directions and speeds to each object in the simulation

The RSE test array contains a wall of RF front ends, or rixels, that echo back the signal modulated by the parameters needed for the system under test (SUT) to detect scene elements. A 64-by-8 array of rixels creates a dynamic radar environment, covering more cases in less time than systems that rely on mechanically moving parts. In addition, it is more stable, predictable, repeatable, and reliable.

Miniaturized rixels are invisible to the radar sensor by design and activated by 3D simulation software, replacing mechanical movement altogether. Each rixel in the array emulates an object's distance and echo strength. As objects get closer, multiple reflections allow for improved detection and differentiation of objects.

Validate crucial functionality down to 1.5 meters

Many test cases — including AEB, FCW, LDW, and LKA — require emulating objects very close to the SUT. For example, vehicles are typically less than 2 meters apart in every direction when approaching a stoplight. In a moving scenario, a two-wheeler — a bicycle, motorcycle, scooter — could swerve into the lane, or a pedestrian might suddenly step into the roadway. The green circle in Figure 11 indicates the distance of 1.5 meters, representing the minimum emulation distance — the ability to emulate objects close to the ego vehicle — to test important safety features. It is challenging to recreate scenarios like these in a lab environment.

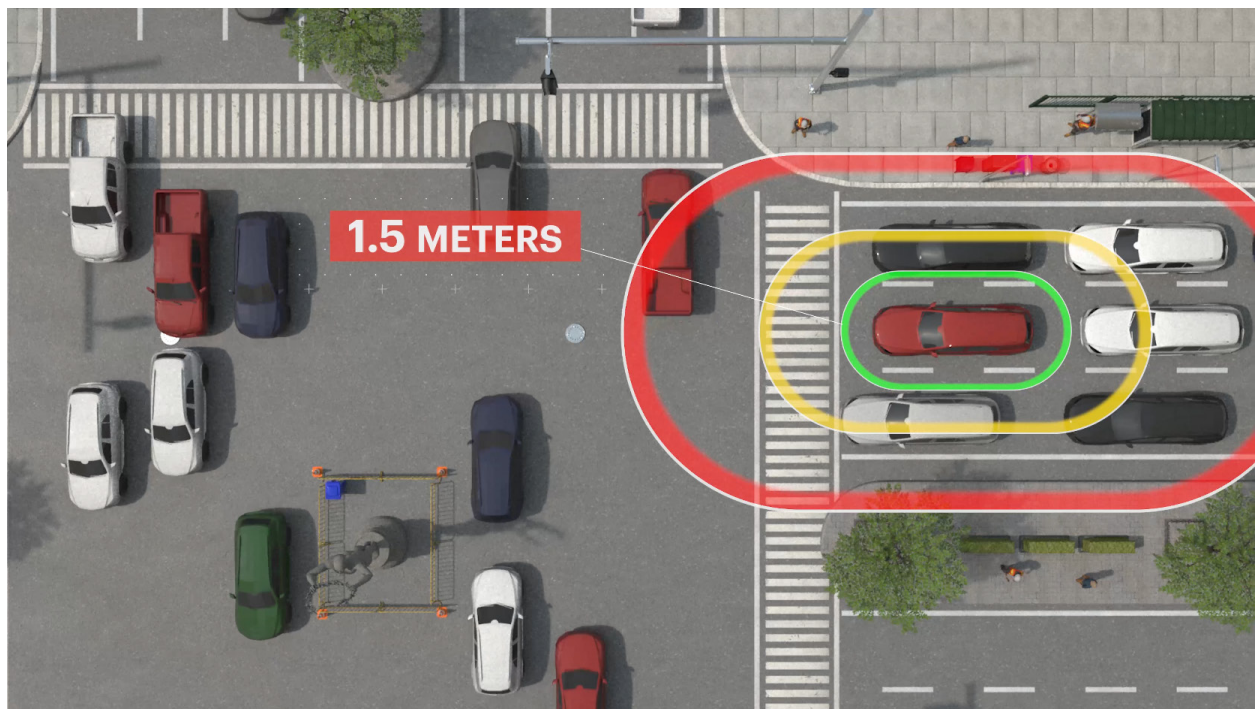


Figure 11. The green circle indicates the distance of 1.5 meters, representing minimum emulation distance

Enable Next-Generation Vehicle Autonomy with In-Lab Full-Scene Emulation

In the automotive space, OEMs are racing toward goals of zero accidents, zero emissions, and zero congestion. These are complex problems that demand innovative solutions in all aspects of automotive design and test. The robustness of autonomous driving algorithms depends on how comprehensive the testing is.

A Shift in ADAS / AV Testing

Keysight's Radar Scene Emulator (RSE) enables OEMs in the automotive industry to test autonomous driving systems with radar sensors faster and with highly complex, multitarget scenes. RSE allows you to create scenarios with up to 512 objects, at distances as close as 1.5 meters from the vehicle. The scenarios can also have dependent attributes, including speed, direction, distance from the vehicle, and angle. The RSE can emulate objects as far away as 300 meters and as near as 1.5 meters. Object velocities can range from -400 to 400 kilometers per hour.

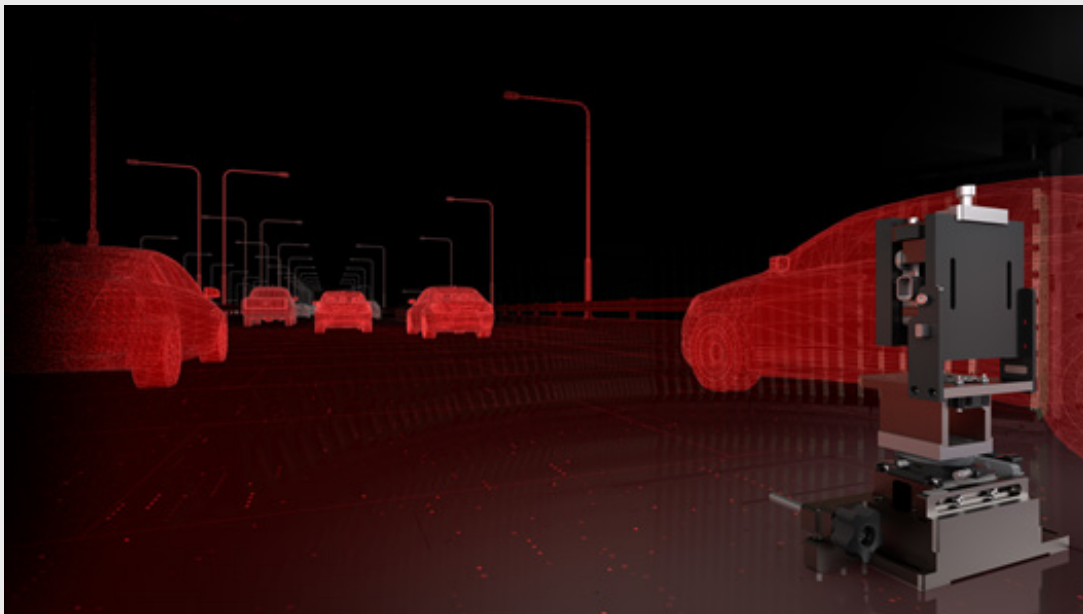


Figure 12. Complex scenario represented by the Keysight RSE

Scene Emulation Earlier in the Lab Accelerates ADAS / AV Testing

Automotive OEMs must shift testing of complex driving scenarios to the lab, eliminating the need to drive millions of miles and dramatically accelerating the speed of testing. They can accelerate the insights from ADAS or AD algorithms by thoroughly testing decisions earlier in the cycle against complex, repeatable, high-density scenes, and with stationary objects or objects in motion.