

Article

Vehicle-in-Virtual-Environment (VVE) Method for Autonomous Driving System Development, Evaluation and Demonstration

Xincheng Cao, Haochong Chen, Sukru Yaren Gelbal, Bilin Aksun-Guvenc * and Levent Guvenc 

Automated Driving Lab, Ohio State University, 1320 Kinnear Rd, Columbus, OH 43212, USA;
cao.1375@osu.edu (X.C.); chen.9286@osu.edu (H.C.); gelbal.1@buckeyemail.osu.edu (S.Y.G.);
guvenc.1@osu.edu (L.G.)

* Correspondence: aksunguvenc.1@osu.edu

Abstract: The current approach to connected and autonomous driving function development and evaluation uses model-in-the-loop simulation, hardware-in-the-loop simulation and limited proving ground use, followed by public road deployment of the beta version of software and technology. The rest of the road users are involuntarily forced into taking part in the development and evaluation of these connected and autonomous driving functions in this approach. This is an unsafe, costly and inefficient method. Motivated by these shortcomings, this paper introduces the Vehicle-in-Virtual-Environment (VVE) method of safe, efficient and low-cost connected and autonomous driving function development, evaluation and demonstration. The VVE method is compared to the existing state-of-the-art. Its basic implementation for a path-following task is used to explain the method where the actual autonomous vehicle operates in a large empty area with its sensor feeds being replaced by realistic sensor feeds corresponding to its location and pose in the virtual environment. It is possible to easily change the development virtual environment and inject rare and difficult events which can be tested very safely. Vehicle-to-Pedestrian (V2P) communication-based pedestrian safety is chosen as the application use case for the VVE in this paper, and corresponding experimental results are presented and discussed. A no-line-of-sight pedestrian and vehicle moving towards each other on intersecting paths with different speeds are used in the experiments. Their time-to-collision risk zone values are compared for determining severity levels. The severity levels are used to slow down or brake the vehicle. The results show that V2P communication of pedestrian location and heading can be used successfully to avoid possible collisions. It is noted that actual pedestrians and other vulnerable road users can be used very safely in this approach.

Keywords: autonomous driving; connected driving; vehicle-in-virtual-environment; model-in-the-loop simulation; hardware-in-the-loop simulation



Citation: Cao, X.; Chen, H.; Gelbal, S.Y.; Aksun-Guvenc, B.; Guvenc, L. Vehicle-in-Virtual-Environment (VVE) Method for Autonomous Driving System Development, Evaluation and Demonstration. *Sensors* **2023**, *23*, 5088. <https://doi.org/10.3390/s23115088>

Academic Editor: Ikhlas

Abdel-Qader

Received: 16 April 2023

Revised: 23 May 2023

Accepted: 25 May 2023

Published: 26 May 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

There have been many well-established research developments over the years on active safety and ADAS systems, such as those in [1–5]. These have been followed by research on robust and energy-preserving control, such as [6,7], and more recently by research on autonomous driving, such as [8–14]. Research on traffic and energy usage improvement has been reported in references such as [15–19], while research on safety improvements has been reported in references such as [9,20–24]. As a result of this and other similar past research, autonomous vehicles, which are self-driving and do not need an operator, are expected to be available soon. Indeed, there are several limited-scale deployments of driverless robo-taxis that are being operated in well-structured, geo-fenced areas with warm weather conditions year-round [25]. Unfortunately, most of the development and evaluation of driverless vehicles is taking place on public roads where the other road users are involuntarily taking part in the development of beta-level AV software. This approach is in contrast with the well-known V-diagram approach of extensive model- and

hardware-in-the-loop evaluation followed by extensive testing in controlled environments such as proving grounds [26].

The usual automotive OEM and supplier development procedure for advanced driver assistance systems involves extensive model-in-the-loop (MIL) simulation, followed by hardware-in-the-loop (HIL) simulation and controlled testing in a proving ground to fully develop the system and its software before public road testing, using a manufacturer's license plate, with several highly attentive and experienced test engineers being present in the vehicle at all times. This final public road testing is carried out for fine tuning of the algorithms and controllers for improved performance in their series production implementation. The well-known V diagram approach of development, evaluation, update and re-evaluation is used during each stage of this well-established development approach [27].

After this introduction in Section 1, this paper presents a literature review of related work in Section 2. Our proposed approach to the Vehicle-in-Virtual-Environment (VVE) is introduced in Section 3. A more detailed explanation of how the VVE is implemented is presented in Section 4, using basic manual driving and path-following inside a virtual environment using a real AV. Subsequent sections focus on an application use case to illustrate how the VVE method works. Section 5 discusses this application use case on the evaluation of pedestrian safety using Vehicle-to-Pedestrian (V2P) communication. Test results and their discussion are presented in Section 6. The paper ends in Section 7 with conclusions and recommendations for future work.

2. Related Work

MIL simulation tools for effective testing of connected and automated driving technologies are presented in [28]. Other actors on the road are added using co-simulation of a microscopic traffic simulator such as SUMO in MIL simulation [15,28]., Worst-case scenario analysis and accelerated analysis can be performed using the many available AV MIL simulation programs for focusing on rare and extreme events [29]. While MIL simulation is very useful, the low- and high-level control systems, the dynamics of the host vehicle and other significant road actors are modeled using software. In order to be more realistic in the evaluation of the low- and high-level control systems and their limitations on computation, memory, latency, etc., these physical systems are incorporated as actual devices into the simulation in HIL simulators [30,31]. The physical electronic control units in a HIL simulator have to run in real time [20]. While HIL simulation increases the fidelity of the AV development and evaluation, the actual vehicle dynamics are still missing and have to be emulated using software. If high-fidelity vehicle dynamics models are used, time-consuming and expensive validation of the model is needed [32].

The VVE approach eliminates this time-consuming vehicle dynamics validation and the loss of fidelity due to the use of a vehicle model by using the actual vehicle moving inside a virtual environment. The VVE approach also eliminates the loss of fidelity due to not using the physical low- and high-level control units, as these are already in the actual vehicle and are consequently part of the VVE system. The virtual environment is the most significant part of the VVE approach, as it enables the testing of many different realistic scenarios and rare and extreme events. A chosen realistic environment with the actual vehicle and possibly other nearby road actors in the loop can be used in the VVE.

Autonomous vehicles used inside the city urban environments rely on scan-matching-based localization using three-dimensional point cloud maps. Even though highly accurate localization based on RTK GPS is possible, this is not preferred, as safe operation requires the autonomous vehicle to localize itself correctly with respect to the road and the surroundings. This is achieved by map-matching of lidar scans [33]. Well-known readily available algorithms such as Normalized Distribution Transform (NDT) and Iterative Closest Point (ICP) can be used in real time for this map-matching-based localization [14,34–36]. Unfortunately, map-matching-based localization cannot be re-created in a proving ground, as the surrounding buildings, trees, infrastructure, etc. that are used as landmarks cannot be replicated. As a partial solution, researchers have built replicas of building blocks within

small controlled testing areas [37]. However, this approach only applies to that small building block, and the very large variety of surroundings that an AV will encounter in real practice cannot be used in the development and evaluation cycle. Physically changing the building block for each different environment is not feasible, as it is very costly and time-consuming. It is also very difficult to re-create the extensive traffic and weather combinations in this approach.

The solution that is currently being used is carrying out the final stage of development on public roads. This is an unsafe approach, putting the lives of all other road users at risk. This public road development approach is also a very inefficient method since it takes a very long time, and very many miles need to be driven to encounter the required rare but extreme situations. The solution proposed in this paper is to replace this unsafe, costly and inefficient public road testing phase with the Vehicle-in-Virtual-Environment (VVE) method of connected and autonomous driving function development, evaluation and demonstration [38].

3. Proposed Approach

The current approach of public road development of autonomous driving functions is illustrated in Figure 1. Note that this is also how an AV operates in the real world. AV sensors, including a modem or similar communication device used for connectivity, collect data about the surrounding environment. While point cloud lidar data is illustrated in the top left of Figure 1, lidar, cameras, radar, GPS and on-board-unit (OBU) modems are generally also used, as shown in the bottom left of Figure 1. Sensor data processing and situational-awareness algorithms, along with decision-making algorithms, are used to generate the higher-level trajectory planning or local modifications in order to accommodate other traffic- or infrastructure-based constraints at a higher level of control. An electronic control unit with CAN connectivity to the throttle, brake and steering actuators implements and executes the lower-level controls to follow the required trajectory. The resulting motion of the AV changes its pose (position and orientation) in the driving environment as illustrated in the top right part of Figure 1, where the AV is about to enter a roundabout.

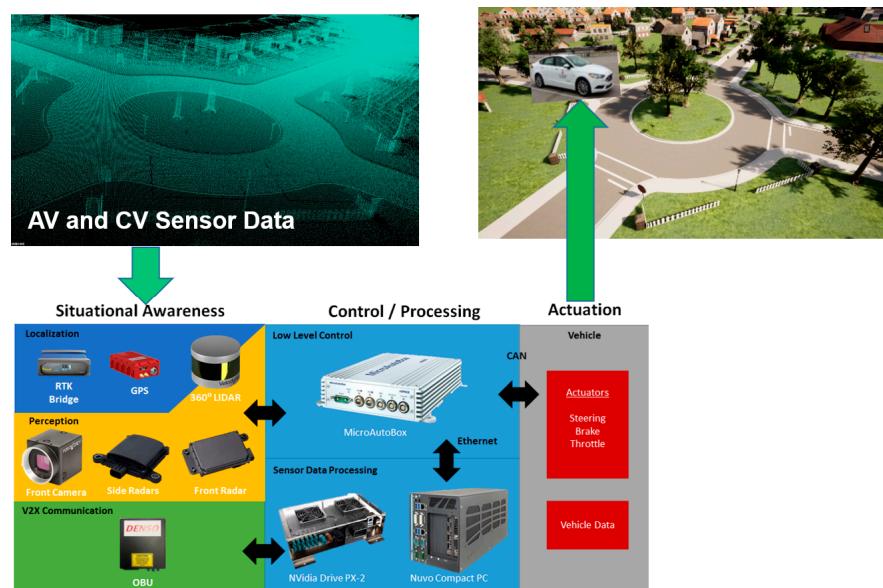


Figure 1. Public road development of AV driving functions.

The VVE approach is illustrated in Figure 2. All of the perception, localization and communication sensor data feeds are disconnected using a connection box added to the vehicle. All of the sensor data feeds are instead connected to simulated data from a highly realistic surrounding-environment model which can easily be changed. The strength of the VVE method is in easily being able to use different environments, as opposed to the

real-building-block approach. A separate edge computer with a powerful GPU/CPU combination runs the simulated environment in real time and produces the required AV and CV sensor feeds. These are fed to the low-level and high-level controls in the Control/Processing part of Figure 2, where the high-level trajectory planning and local updates and trajectory-following controls of the AV work as before, but using the simulated sensor data. The low-level controls send the actuator commands, and the AV moves as before, but this time in a large and empty area such as a large parking lot or the vehicle dynamics area in a proving ground. The motion of the AV in the large parking lot is tracked using the actual RTK GPS sensor of the vehicle, which determines the new pose in the real-time simulated environment also. This procedure is illustrated in the top right corner of Figure 2.

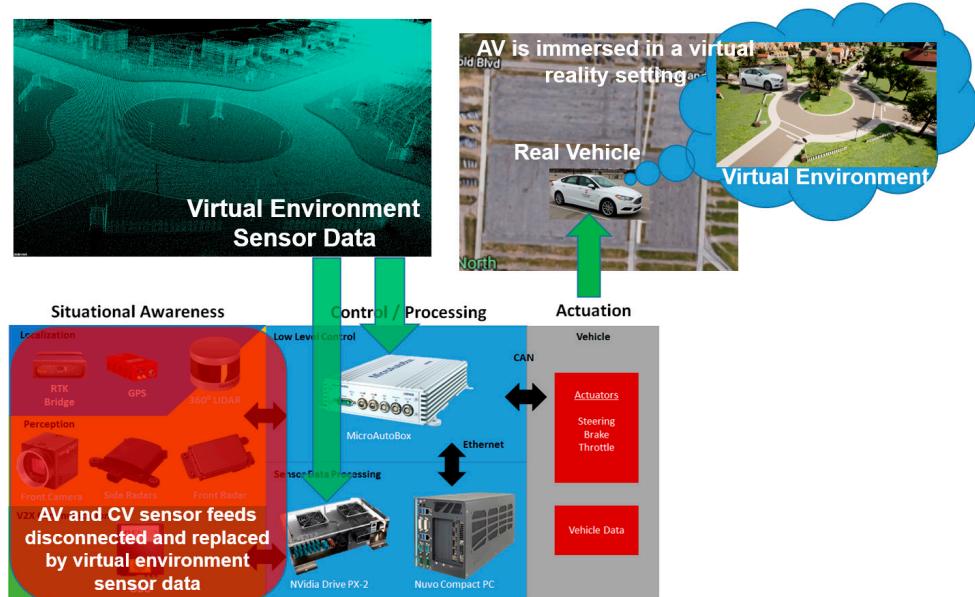


Figure 2. Vehicle-in-Virtual-Environment.

The vehicle itself is immersed in a virtual reality setting. It is as if the AV now has a VR headset or augmented reality glasses and is seeing the virtual environment even though it is in a large and flat parking lot or similar test area. Note that it is possible to run the real-time environment simulation in the cloud using the VVE as a service instead of using edge computing with a very powerful simulation computer in the vehicle. It is also possible to have multiple users share the same virtual environment simultaneously, even though they are at different locations, meaning different empty parking lots. This feature allows a very safe method of remotely located teams running and sharing the same AV development and evaluation environment. Other traffic is added realistically using real-time microscopic traffic co-simulation or as programmed non-player characters in the simulation. While the examples in this paper use Unreal Engine for the simulation environment and CARLA for the AV simulator in Unreal, any of the available 3-D environment-rendering engines and AV simulators can be used as long as they can run in real time and produce realistic AV sensor data and, if needed, CV sensor data. Real CV data for other team members sharing the environment can also be generated by using another modem or communication device. The AV sensor data is converted to the format that is expected by the AV controllers for seamless operation [39]. If the simulation environment is larger than the empty parking lot used, roundabouts are added to the environment to change the direction of the vehicle at the endpoints of the test area and the AV will move to the next building block(s) in the virtual environment.

A high-fidelity evaluation, development and demonstration method for self-driving that uses the actual autonomous vehicle(s) in a safe but realistic manner is currently not available. This causes high costs and long development times, with the risk of unacceptable

performance in the form of fatal accidents, too many near misses and too-frequent need for operator override. These problems and deficiencies delay the widespread market introduction of self-driving cars and reduce public trust in the global autonomous car industry. Even though this gigantic industry may see a large financial loss due to mistrust in the technology, testing and development is still mainly taking place on public roads. The reasons for this are that the currently used development and evaluation methods rely heavily on pure simulation in the form of model-in-the-loop, hardware-in-the-loop, and vehicle-in-the-loop (still inside a lab environment). This current approach keeps the moving vehicle out of the loop. This is significant, as the actual vehicle dynamics are missing and have to be emulated using software. Simple vehicle dynamics models will result in low fidelity of the simulation evaluation. If high-fidelity vehicle dynamics models are used, time-consuming and expensive validation of the model is needed. In addition, even high-fidelity vehicle models have limitations in accurately representing the highly coupled, nonlinear and complex dynamics of a real vehicle.

The VVE approach eliminates both the time-consuming vehicle dynamics validation for a high-fidelity model and the loss of fidelity due to the use of a simple model by using the actual vehicle that is moving inside a virtual environment. The classical proving-ground testing does not have the surrounding building, infrastructure, vegetation and other-traffic environments that are needed to fully test this technology. Buildings and trees, for example, are landmarks that are significant in the localization of the AV. Attempts at creating replicas of city blocks are useful. However, this approach can only partially replicate a small, fixed environment and does not solve the problem [37].

The Vehicle-in-Virtual-Environment (VVE) method proposed here takes care of all the problems associated with the current state-of-the-art methods and products by physically driving the actual vehicle in an immersed-reality environment while feeding it realistic autonomous driving system sensor signals. As a result, the AV is fully tested in all possible combinations of environment, other traffic, vulnerable road user, weather condition and fault situations while being in a very safe actual environment with no collision risk. The advantages of the VVE method over current approaches discussed above are tabulated in Table 1. The VVE approach is expected to be a game-changer for the autonomous vehicle industry, legislators, user groups and the public, as it will significantly decrease development cost and development time while improving product safety. The VVE is a safe approach for the development and evaluation of autonomous driving functions since it takes place in a large and empty area, thereby reducing the risk of collisions, as opposed to public road testing. Development cost and time are reduced significantly as rare and extreme events can be programmed into the VVE. In contrast, long miles of public road driving are required for such events to happen in public road testing. The safety of autonomous driving functions will be improved by using the VVE approach, as they will be evaluated and developed to handle these rare and extreme events.

The cost of the VVE product is also expected to be lower than that of hardware-in-the-loop simulators that are widely used for automotive software development, and significantly cheaper as compared to proving ground or controlled-city-block testing. Deployers of technology such as Smart Columbus will be able to evaluate a deployment in any geo-fenced urban area they choose. They will be able to immediately see the effect of unexpected situations in the VVE evaluation. Technology companies will be able to easily demonstrate how their system would operate in a planned deployment site such as that in [40] without having to physically go there and spend months of mapping, testing and bug fixing. Certification agencies and local governments will be able to use this tool to fully test vendor technologies before certification and for accident reconstruction and analysis.

Table 1. Advantages of the VVE over current methods. Red color denotes undesirable or difficult property while green denotes desirable and yellow in between.

Method Comparison	MIL/HIL	Proving Ground	Building Block	Public Road	VVE
Implementation	soft/hard [15,20,28–33,39]	hard [20,33,39]	hard [20,33,37]	hard [20,25]	hard/soft [38]
Adaptability to Different Scenarios	relatively easy [15,20,28–33,39]	difficult [20,33,39]	very difficult [20,33]	not possible [20,25,26]	easy [38]
Vehicle Model	high fidelity [15,20,28–33,39]	real vehicle [20,33,39]	real vehicle [20,33,37]	real vehicle [20,25]	real vehicle [38]
Safety	safe [15,20,28–33,39]	controlled experiment necessary [20,33,39]	controlled experiment necessary [20,33,37]	not safe [20]	safe [38]
Cost	high [15,20,28–33,39]	very high [20,33,39]	very high [20,33]	very high [20]	moderate [38]
Repeatability	high [15,20,28–33,39]	high [20,33,39]	high [20,33]	Low [20]	high [38]
Time to Implement	moderate [15,20,28–33,39]	long [20,33,39]	long [20,33]	very long [20,25]	moderate [38]

4. The VVE Method

Our current VVE architecture implemented in the ego AV is illustrated in Figure 3. Motion in the actual empty parking lot and motion in the corresponding virtual world are correlated with each other in Figure 3. As noted before, while our current architecture uses an Unreal Engine rendering of the virtual test environment and the CARLA AV simulator, any of the currently available three-dimensional surrounding-environment modeling and AV simulation tools can be used. At the beginning of the VVE run, the vehicle in the empty parking lot is placed at a desired reference position corresponding to the initial position of the virtual vehicle in the virtual environment. Both vehicles start at the same orientation. The information captured on the real vehicle is recorded and sent to the virtual environment in a way that keeps both virtual and real vehicles synchronized. Thus, any position change in the physical vehicle is reflected in the virtual environment. At each new position and orientation in the virtual environment, sensor data is collected in the simulation computer and sent to the AV computer system.

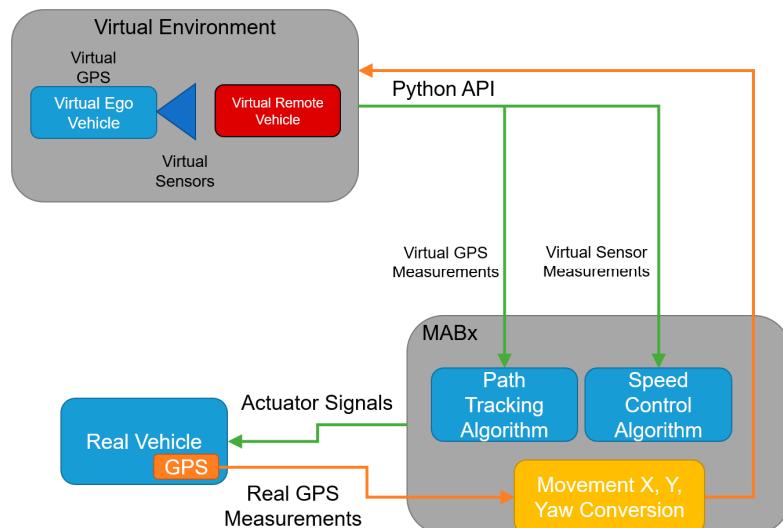


Figure 3. Correlation of motion in the real and virtual environments for a simple trajectory tracking application.

The actual vehicle used in this paper is shown in Figure 4. The simulation computer, the perception sensor computer, the low-level control computer and the GPS processing unit are shown in the trunk of this vehicle in Figure 4. The dSpace microautobox unit is a generic electronic control unit with CAN and Ethernet connections and acts as the low-level controller. The calculated trajectory or trajectory modification is tracked within the steering and speed controller implemented in this low-level control unit, which sends the drive-by-wire CAN commands for throttle, brake and steering to the AV drive-by-wire interface. An RTK GPS unit is used to keep track of position and orientation changes, which are then conveyed to the virtual world simulation to read perception sensors at the new virtual world location. Figure 5 shows the planned path in the virtual world on the right and the actual AV in the parking lot that is immersed in that environment and following that path on the left.

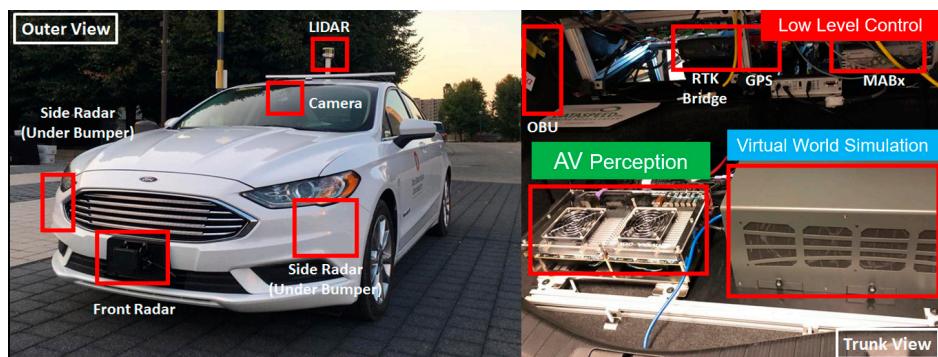


Figure 4. Research level AV used and the relevant components in its trunk.



Figure 5. Actual AV in parking lot and planned path in the virtual world. A is the starting point and B is the end point of the route.

Safe operation of the VVE is ascertained as follows. The first safety guard is the operation of the VVE in a large empty area which eliminates the risk of a collision. Currently, there is a safety operator in the driver's seat. The vehicle used is drive-by-wire and has manual safety overrides. The safety operator can always override the VVE operation by hitting the override button or by applying throttle/brake/steering. Since the VVE experimentation takes place in a geo-fenced area, the actual GPS location of the vehicle is used to automatically stop vehicle motion beyond the boundaries of operation.

Please note that the virtual world representation on the dashboard screen of Figure 5 is used only to illustrate how the method works. In the actual implementation, there is no need for a dashboard screen representation. The human user in the driver's seat will not use a VR headset, as this is dangerous and also not needed. The person in the driver's seat is a safety operator. A VR or augmented-reality headset will only be used by other passengers in the AV if the VVE is used for demonstration purposes. The VVE method operates in a large flat area in order to make sure that there are no collision possibilities.

5. Pedestrian Safety Using V2P

The application use case for the VVE that is used in this paper is pedestrian safety using V2P communication. The V2P vulnerable road user safety mobile phone app developed in our earlier work in reference [21] is used here for the communication between the AV and pedestrian. Development of pedestrian and vulnerable road user safety systems on public roads is not recommended due to safety issues. The alternative approach of using a mannequin on a movable platform for controlled testing in a proving ground is useful but is very limited in scope, considering the many different situations that occur in real life when AVs and CVs encounter and interact with vulnerable road users. The VVE method is an excellent choice here because, along with software-based vulnerable road users, it is also possible to use real vulnerable road users that share the same virtual environment and move at displaced and safe locations while the AV in the empty parking lot will perceive them to be on a collision risk path. This section, therefore, starts with Vehicle-to-Everything (V2X) and V2P communication and proceeds with how to implement the V2P-based VVE testing.

Vehicle connectivity research has seen rapid advancement in recent years. From the perspective of safety, connectivity can handle some traffic scenarios that are traditionally challenging, such as pedestrian motion detection under NLOS (no-line-of-sight) conditions. Refs. [21,41–43] provide some examples of pedestrian motion tracking and collision risk assessment implementations through cellular, Bluetooth or Wi-Fi connections, for example. There are two groups of technologies being used currently for V2X communication: Wireless-Local-Area-Network (WLAN)-based solutions and cellular-based solutions (C-V2X). WLAN-based technologies are based on the IEEE 802.11p standard [44]. Ref. [45] offers a performance evaluation of IEEE 802.11p and IEEE 1609 WAVE (another standard built upon IEEE 802.11p) standards in the sense of capacity and delay and concludes that the traffic prioritization schemes work well and that stable connections in high-density traffic is possible. The most notable technology used in this branch is Dedicated Short Range Communication (DSRC), where direct communication among vehicles and infrastructure can be established. It operates in the 5.9 GHz band with a bandwidth of 100 MHz in the U.S., and its devices have an operation range of 1 km [41].

Cellular-based technologies are another popular area of V2X connectivity. These technologies are developed under 3GPP (3rd Generation Partnership Project) and include a wide range of protocols such as GSM (Global System for Mobile)/2G, UMTS (Universal Mobile Telecommunications System)/3G, LTE (long-term evolution)/4G, and 5G NR (5G New Radio)/5G cellular networks, with 6G on the way. Ref. [42] provides an overview of 3GPP standards and offers a technical comparison between 3GPP functionalities and IEEE 802.11p standards. Refs. [46,47] also provide a brief introduction of GSM network topology. In recent years, LTE- and 5G-based solutions have been explored more, as they offer significant advantages over previous-generation cellular networks. Ref. [48] tests and compares LTE and 5G NSA (non-standalone) networks under V2X application and observes significantly better performance of 5G NSA as compared to LTE in the sense of response time and packet loss.

Apart from the aforementioned two main groups of solutions, some other connectivity options exist. Wi-Fi is a wireless connection protocol based on earlier variations of the IEEE 802.11 standard, but it is not a suitable option for V2X applications due to its varying data rate under different conditions [46]. ZigBee is a communication scheme based on IEEE 802.15.4 [44] and is another possible alternative for V2X connectivity. Ref. [46] tests the handshake time of different ZigBee channels. Bluetooth is another short-range wireless communication option and is explored by many recent works. Ref. [49] describes an Android application that tracks real-time vehicle motions and uses Bluetooth to transmit information received on DSRC devices to connected mobile phones. Refs. [46,50] analyze the handshake time of Bluetooth connection under noisy Wi-Fi conditions. Ref. [21] introduces a mobile phone application that broadcasts PSM (personal safety messages) between vehicle and pedestrian via Bluetooth low-energy connection using the advertising mode.

An extension of this last Bluetooth BLE communication app between two mobile phones will be used here as it has performed very well in recent deployments.

The pedestrians or vulnerable road users run the app in their mobile phones, which broadcast their location information using PSM to nearby vehicles, where another mobile phone or Bluetooth device running the software listens to this information and uses it to determine collision risk with the pedestrian or vulnerable road user. If the collision risk is high and the vehicle and pedestrian or vulnerable road user are close, the AV applies autonomous braking to avoid an accident. This V2P communication is illustrated in Figure 6. It should be noted that C-V2X and over-the-cloud connectivity can also be used to obtain similar results and can be tested using the VVE method. Figure 7 shows the VVE implementation architecture for developing, evaluating and demonstrating V2P-based vulnerable road user safety. Experimental results are presented and discussed in the next section.

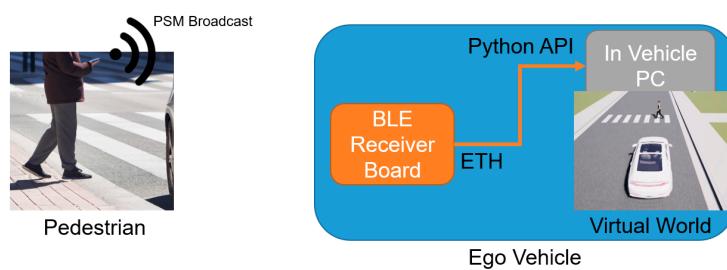


Figure 6. Mobile phone BLE based communication between vulnerable road user and AV.

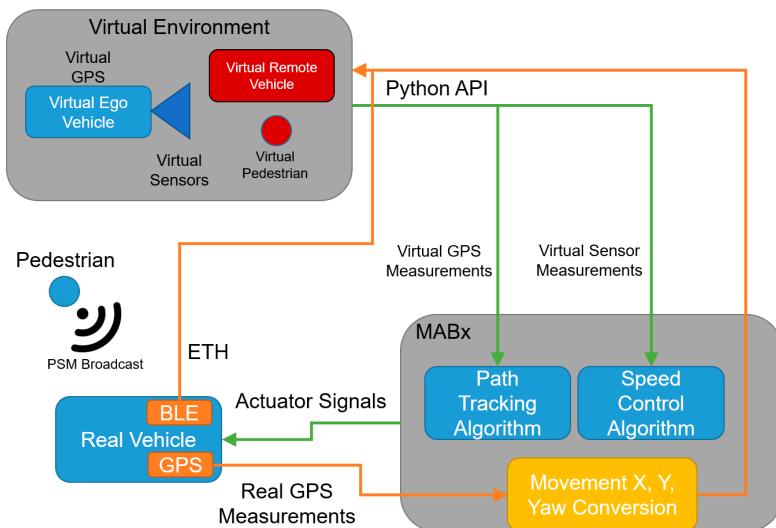


Figure 7. V2P-based vulnerable road user safety implementation of VVE architecture.

6. Experimental Protocol

This section presents the experimental protocol for a proof-of-concept demonstration of V2P functionality in a virtual environment using pedestrian safety through V2P communication as an example. The CARLA simulator and Unreal Engine are selected as the AV simulator and environment modeling tool, respectively, for this demonstration. The collision risk estimation routine that is used is illustrated in Figure 8 [21]. The vehicle and pedestrian headings are first compared, and if their future paths intersect one another, the intersection point becomes the potential collision point. A collision zone is then established around this collision point, in this case as a rectangular area of size $6\text{ m} \times 6\text{ m}$. Based on the current heading and speed of the vehicle and the pedestrian, Time-To-Zone (TTZ) can be calculated separately from the perspective of both the vehicle and the pedestrian. The two TTZ values are then compared to each other, and if their difference is small enough,

collision is deemed highly probable, as the vehicle and the pedestrian are expected to arrive at the collision zone at the same time. For this implementation, the TTZ difference is compared to a threshold value T_s , in this case chosen as 1.5 s, to determine if the situation is potentially dangerous. Once a situation is deemed dangerous, automatic braking will be applied to the vehicle to avoid possible collision.

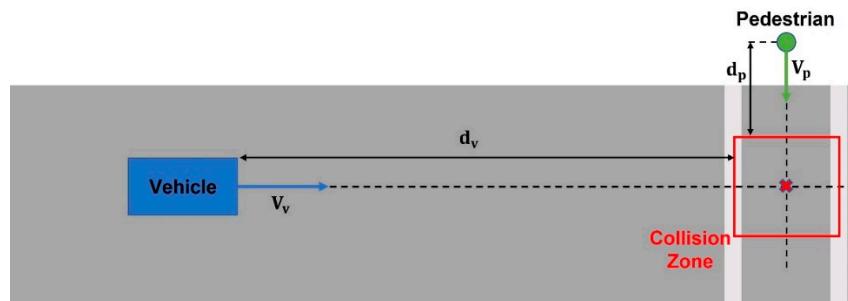


Figure 8. Collision risk estimation. The vehicle and pedestrian paths intersect in the collision zone shown with a red square.

To accommodate different situations, a three-level severity classification is implemented as shown in Figure 9. Once the TTZ difference is within the chosen threshold, the TTZ value for the vehicle is used to determine the severity level of the possible collision. In general, a smaller TTZ value indicates a shorter headway time-to-collision, and hence requires harder braking. In this case, the TTZ threshold value used to differentiate level 1 and 2 severity is chosen as 2.3 s, and the TTZ threshold value used to differentiate level 2 and 3 severity is chosen as 1.5 s. It should again be noted that the threshold values and collision zone sizing can easily be modified to accommodate various settings such as different vehicle dynamic models and road conditions.

Severity Level 1	Sufficient distance to collision zone, Mild braking necessary.
Severity Level 2	Close to collision zone, Moderate braking necessary.
Severity Level 3	Very close to collision zone, Emergency braking necessary.

Figure 9. Severity levels.

It should be noted that the VVE method takes place in a large and empty flat area without any other road actors. There is a safety operator at the driver's seat at all times. The vehicle is not allowed to move beyond this large and empty flat area. The vehicle does not operate on a public road during VVE development. There is no collision risk. Hence, NHTSA's FMVSS Considerations for Vehicles with Automated Driving Systems and ANSI and ISO standards on interlocked guarding, interlocked controls on equipment are not applicable or needed for the VVE method.

7. Results of Experiments and Discussion

We first present a traffic scenario as displayed in Figure 10. The ego vehicle approaches an intersection, where a pedestrian intends to cross. Another vehicle is parked at the intersection in a neighboring lane, blocking the line-of-sight (LOS) between the ego vehicle and the pedestrian. This is a typical traffic case, and the no-line-of-sight (NLOS) condition makes it difficult for the ego vehicle's onboard sensors to detect the pedestrian. In Figure 11, simulation results are presented for the worst-case scenario, where V2P connection is not implemented, and the pedestrian decides to quickly run across the intersection as the ego vehicle approaches, necessitating emergency braking. It can be observed, however, that the ego vehicle fails to decelerate for the crossing pedestrian whom it does not detect, and collision becomes imminent. Figure 12 demonstrates this process in more detail. It can be observed that due to the lack of V2P connectivity, braking is not applied even when the TTZ difference drops below the threshold value that should trigger braking actions, and

hence the vehicle speed keeps increasing until the collision event occurs shortly after the pedestrian enters the collision zone.

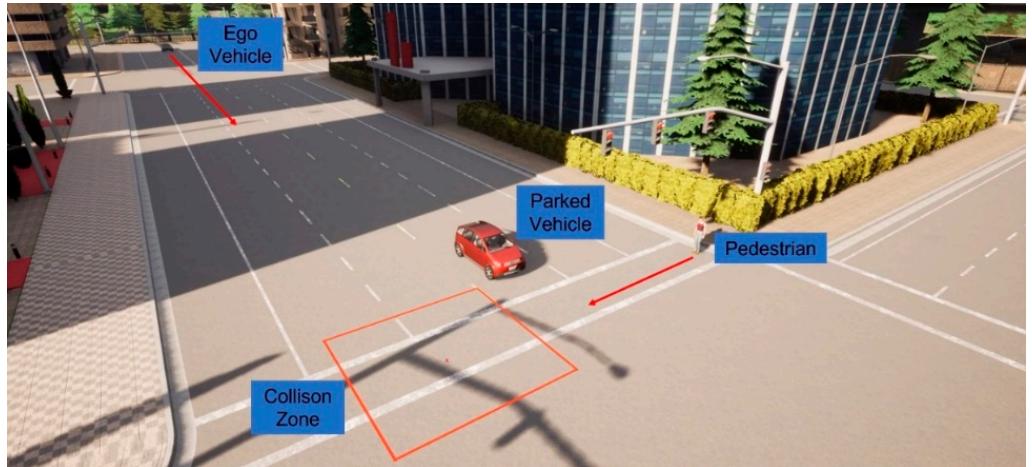


Figure 10. NLOS intersection scenario with suddenly darting pedestrian.

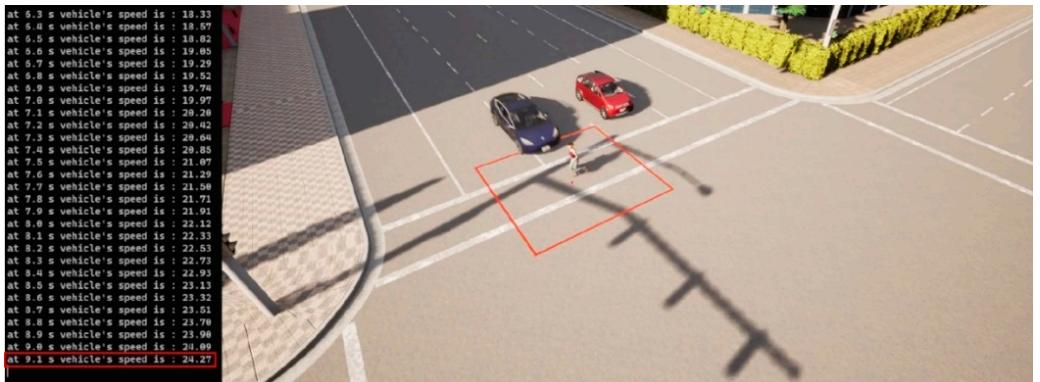


Figure 11. Automatic braking not engaged, collision imminent. Red square shows the collision risk zone.

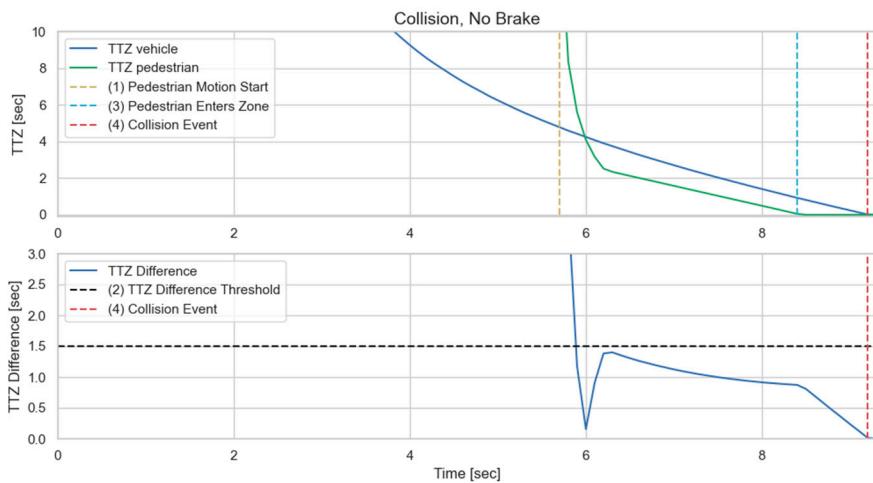


Figure 12. Time history of TTZ values and TTZ difference; collision imminent.

We then implement the V2P communication-based autonomous braking scheme introduced above and run the experiment again. The results are presented in Figures 13 and 14.

It can be observed that the ego vehicle begins to engage the brake before it is able to establish a LOS with the pedestrian and is able to eventually come to a stop before colliding with the pedestrian. In this case, a level 3 severity is needed, and maximum braking is applied to avoid collision. In order to demonstrate the functionality of the three-level severity design, two more cases are experimented. In the case displayed in Figures 15 and 16, the pedestrian walks slowly across the intersection and the ego vehicle has ample time to react. As a result, only a level 1 severity is needed, and the ego vehicle only needs to apply minor braking to stop and avoid collisions. In the case displayed in Figures 17 and 18, the pedestrian runs across the intersection while the ego vehicle is still somewhat far away, allowing the ego vehicle to avoid collision by applying moderate braking action triggered by a level 2 severity classification.

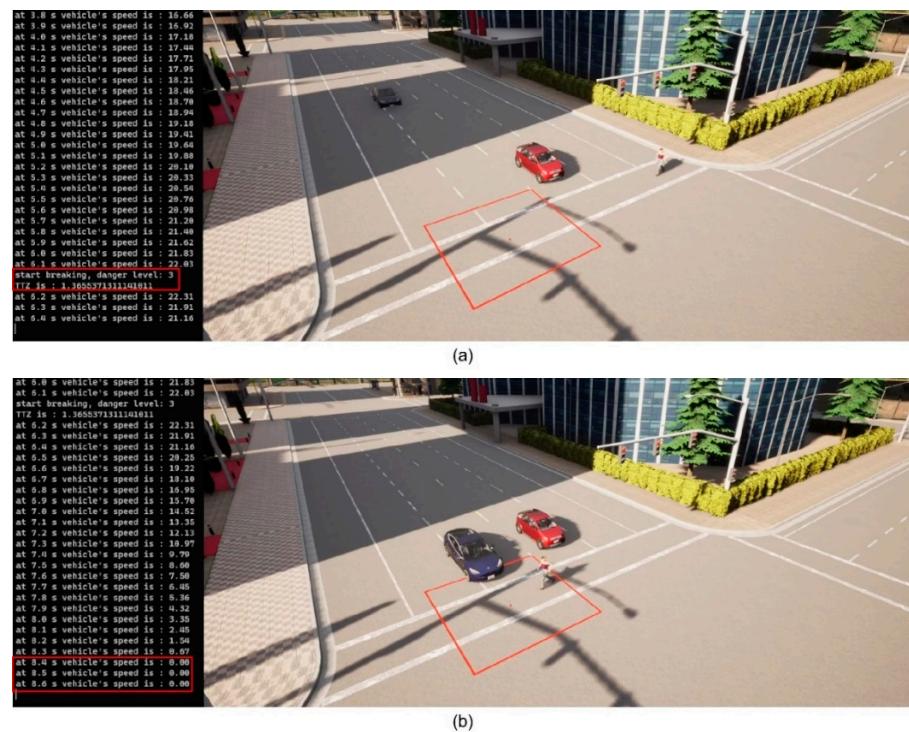


Figure 13. (a) Automatic braking engaged, severity level 3; (b) Vehicle stopped, collision avoided. The red square is the collision risk zone.

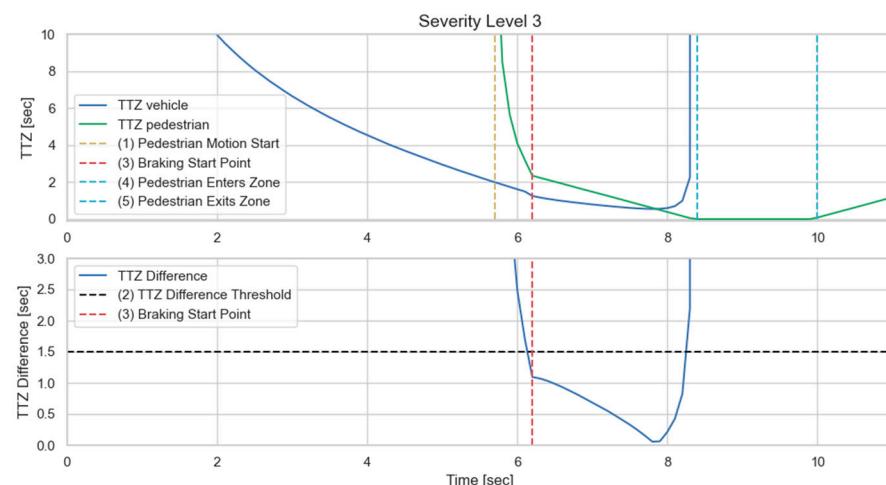
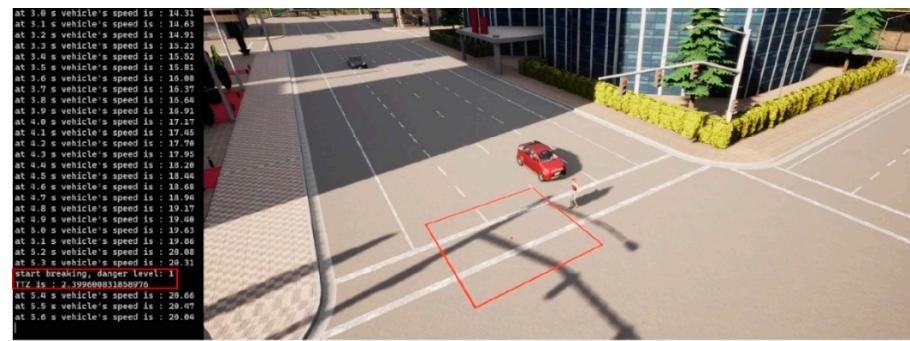
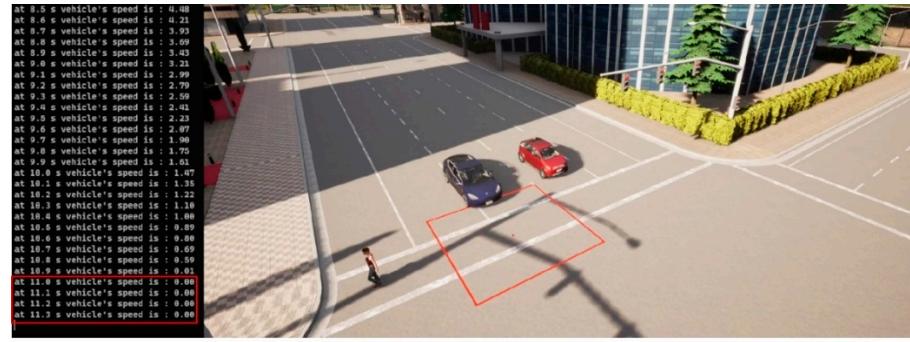


Figure 14. Time history of TTZ values and TTZ difference; severity level 3.



(a)



(b)

Figure 15. (a) Automatic braking engaged, severity level 1; (b) Vehicle stopped, collision avoided. The red square is the collision risk zone.

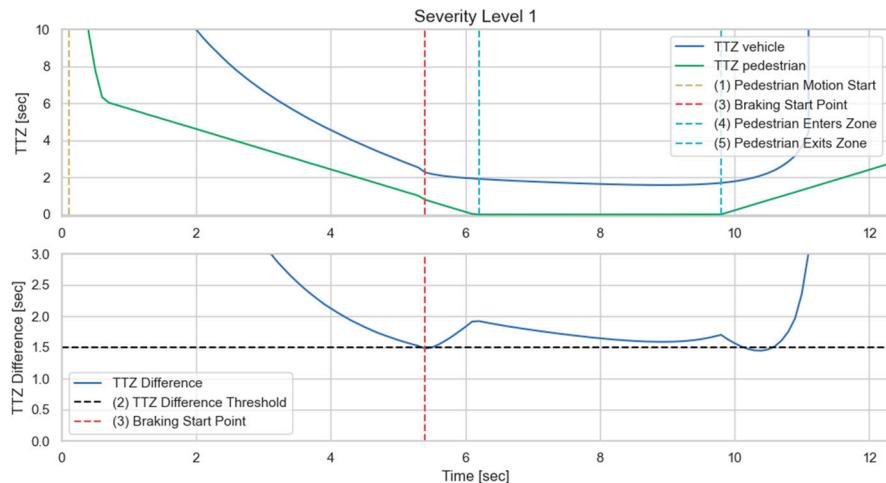
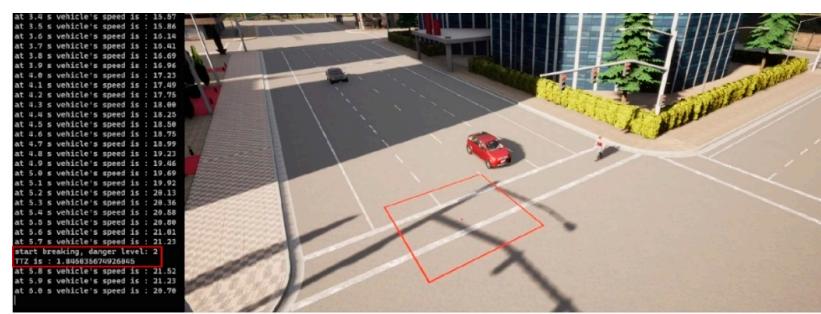


Figure 16. Time history of TTZ values, TTZ difference; severity level 1.



(a)

Figure 17. Cont.

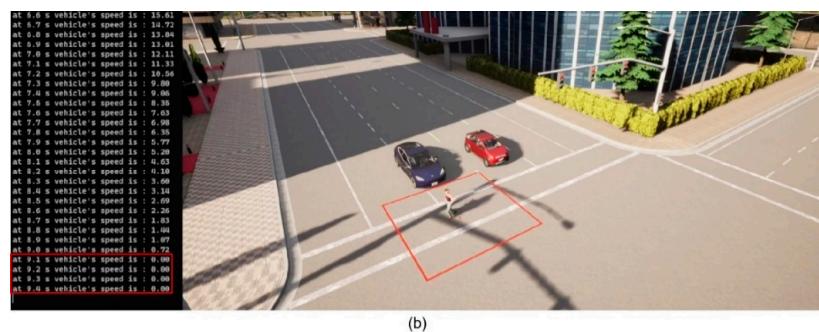


Figure 17. (a) Automatic braking engaged, severity level 2; (b) Vehicle stopped, collision avoided. The red square is the collision risk zone.

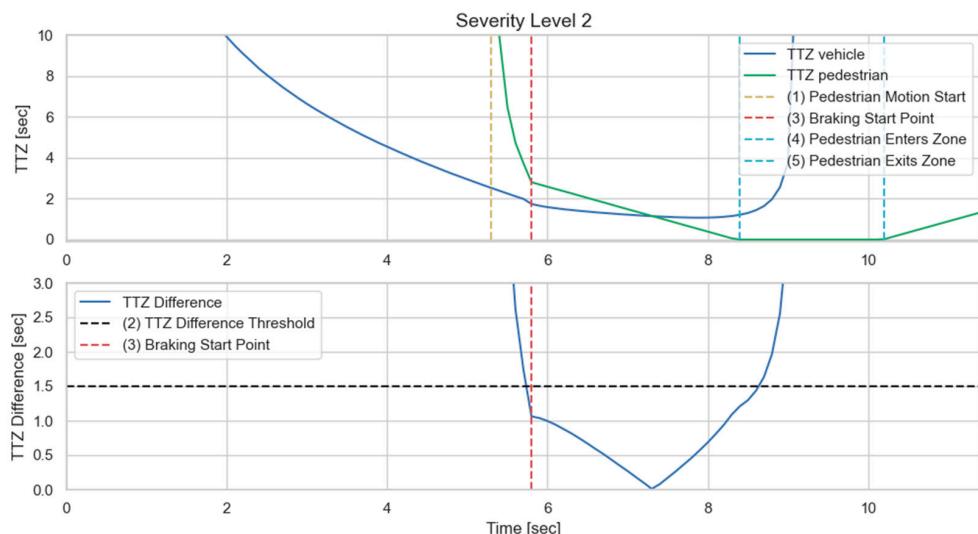


Figure 18. Time history of TTZ values and TTZ difference; severity level 2.

In addition to the above-mentioned scenario, we also present another commonly encountered traffic situation, as displayed in Figure 19. In this case, the vehicle takes a right turn at the intersection, where the pedestrian intends to cross. Before the vehicle finishes the turn, no line-of-sight can be established with the pedestrian due to object occlusions. V2P connectivity is, thus, necessary for collision avoidance functions. Figures 20 and 21 demonstrate the simulation results for a worst-case scenario setup, where the pedestrian begins to cross the intersection as the vehicle is performing the turning maneuver. It can be observed that the V2P algorithm detects the collision risk and classifies it as severity level 3, and the vehicle responds with emergency braking, successfully stopping before it reaches the collision zone.

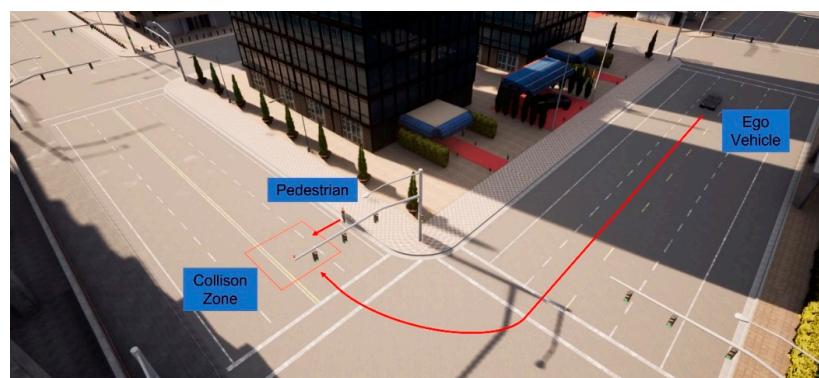


Figure 19. Alternative NLOS intersection scenario with suddenly darting pedestrian.

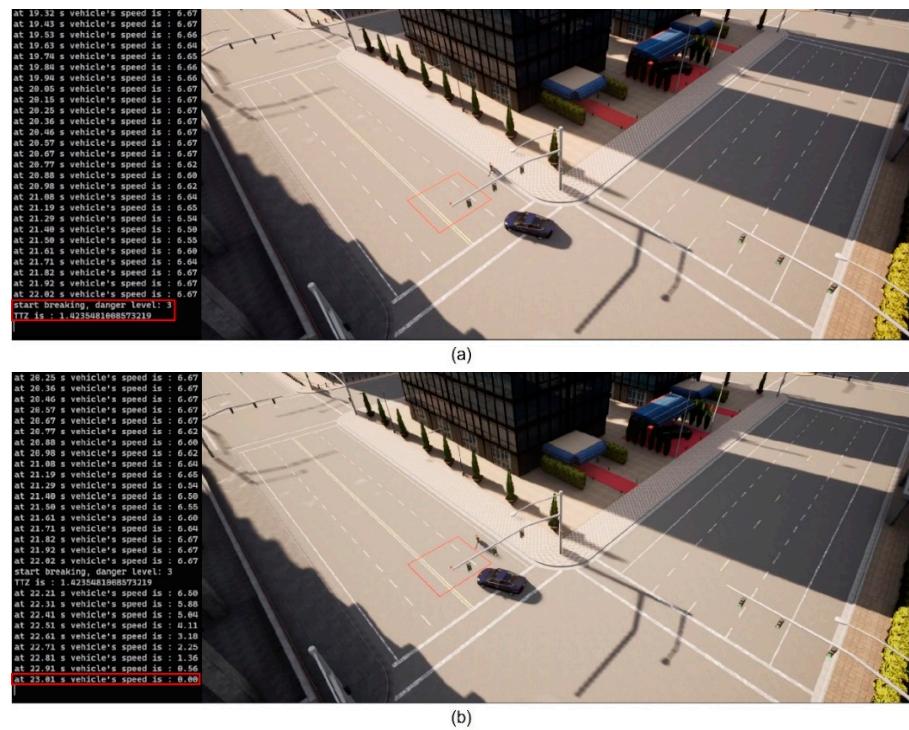


Figure 20. Turning vehicle (a) Automatic braking engaged, severity level 3; (b) Vehicle stopped, collision avoided. Red square is the collision risk zone.

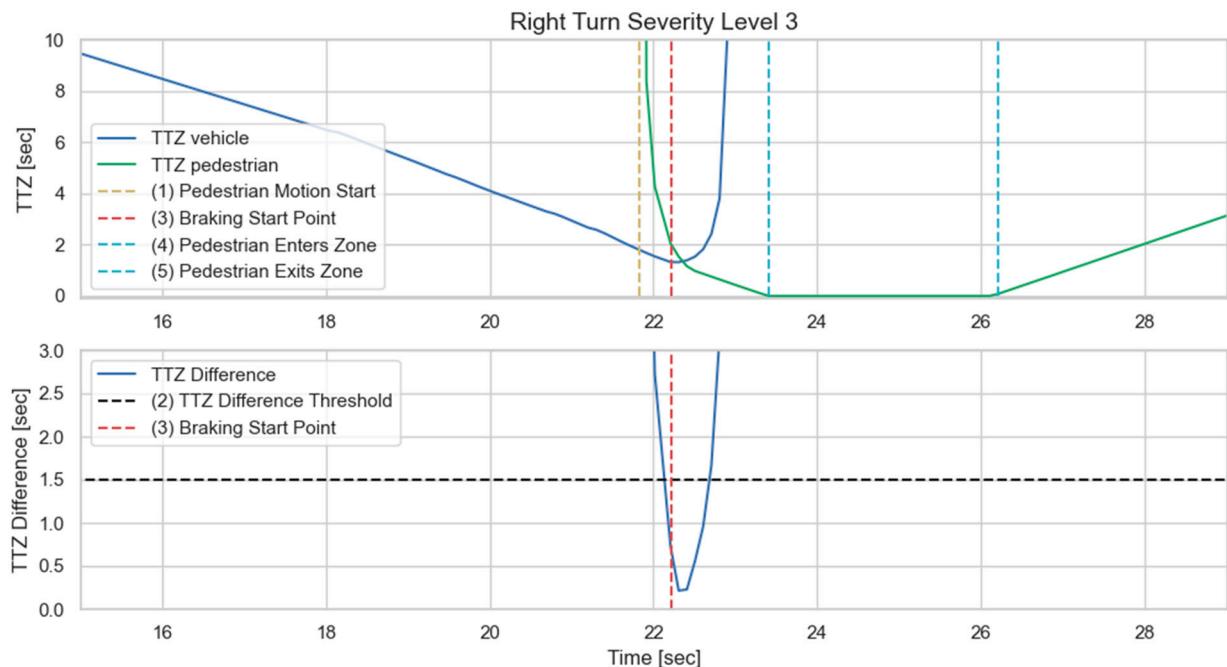


Figure 21. Time history of TTZ values and TTZ difference; severity level 3 for turning vehicle.

In the VVE experiments presented above, the vehicle and pedestrian are at two close but different locations with no possibility of a real collision, as illustrated in Figure 22. A mobile phone is placed in the vehicle, and another mobile phone is in the pedestrian's possession and they both use a V2P communication app that sends PSM data of the pedestrian to the vehicle. The vehicle is in an open space, a parking lot, so that it can maneuver, while the pedestrian is at another safe location. Both mobile phones are connected to the same CARLA environment, and their sensor data are fed into the environment. Collision

risk is calculated in the environment and the appropriate level of braking command is sent to the vehicle to facilitate the braking action in the parking lot. As a result, it is possible to realistically and safely test different vehicle and pedestrian interactions, including dangerous ones.

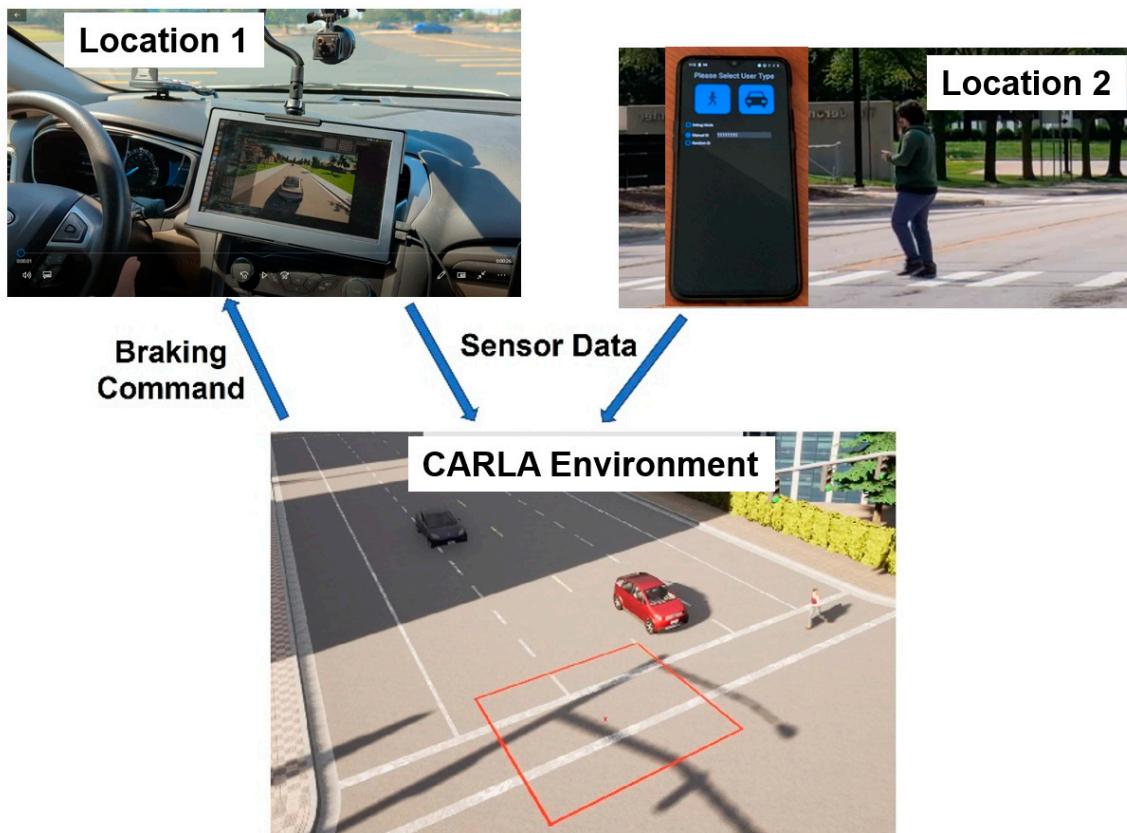


Figure 22. Experimental setup.

Simulation-based methods do not have the actual vehicle and pedestrian(s) within the test environment, which reduces their validity. Running such experiments with real pedestrians in a proving ground or on public roads should not be attempted as it is not safe for the pedestrians. The VVE method, in contrast, provides a safe and realistic method of testing autonomous driving operation such as detection (using V2P in this case) and autonomous braking in a safe manner with (a) real pedestrian(s). This was the main point that the authors wanted to demonstrate in this introductory paper on the VVE method. The number of experiments reported in this paper have been adjusted to demonstrate this point.

The architecture in the experiments of our paper used the CARLA AV simulator inside the Unreal Engine rendering of the virtual test environment. However, the VVE method can work with any AV simulator and virtual environment rendering software as long as these can be run in real time and can generate the raw sensor data required by the actual AV computing system.

8. Conclusions and Recommendations for Future Work

The VVE method was introduced in this paper as a safe, efficient and low-cost method of developing, evaluating and demonstrating connected and autonomous driving functions. The widespread use of VVE is expected to replace the current unsafe and time-consuming approach of public road development of AV driving functions. A path-following task was used to illustrate how the method works. V2P communication-based vulnerable road user safety was chosen as the application use case in this paper, and VVE runs were used to

demonstrate how the method can safely be employed with real pedestrians and an AV in a parking lot that are all immersed in the same realistic, three-dimensional environment. Results for non-line-of-sight pedestrians, including a sudden darting pedestrian, and a turning vehicle were used in the evaluations, demonstrating the efficacy of the VVE method. It is recommended that future work concentrate on more application use cases to demonstrate the full potential of the VVE method and help with its widespread adoption.

Author Contributions: Conceptualization, all authors; methodology, S.Y.G., X.C. and H.C.; software, S.Y.G., X.C. and H.C.; validation, X.C. and H.C.; formal analysis, all authors; investigation, all authors; resources, B.A.-G. and L.G.; data curation, all authors; writing—original draft preparation, all authors; writing—review and editing, B.A.-G. and L.G.; visualization, all authors; supervision, B.A.-G. and L.G.; project administration, B.A.-G. and L.G.; funding acquisition, B.A.-G. and L.G. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: The authors thank and acknowledge the support of the Automated Driving Lab at the Ohio State University.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

ADAS	Advanced Driver Assistance Systems
C-V2X	Cellular V2X
DSRC	Dedicated Short Range Communication
HIL	Hardware-in-the-Loop
MIL	Model-in-the-Loop
V2P	Vehicle to Pedestrian communication
V2X	Vehicle to Everything communication
VVE	Vehicle-in-Virtual-Environment

References

1. Guvenc, B.A.; Guvenc, L.; Ozturk, E.S.; Yigit, T. Model Regulator Based Individual Wheel Braking Control. In Proceedings of the 2003 IEEE Conference on Control Applications, CCA 2003, Istanbul, Turkey, 25 June 2003; Volume 1, pp. 31–36.
2. Guvenc, L.; Guvenc, A.B. The Limited Integrator Model Regulator and Its Use in Vehicle Steering Control. *Turk. J. Eng. Environ. Sci.* **2002**, *26*, 473–482.
3. Oncu, S.; Karaman, S.; Guvenc, L.; Ersolmaz, S.S.; Ozturk, E.S.; Cetin, E.; Sinal, M. Robust Yaw Stability Controller Design for a Light Commercial Vehicle Using a Hardware in the Loop Steering Test Rig. In Proceedings of the 2007 IEEE Intelligent Vehicles Symposium, Istanbul, Turkey, 13–15 June 2007; pp. 852–859.
4. Lenzo, B.; Zanchetta, M.; Sorniotti, A.; Gruber, P.; De Nijs, W. Yaw Rate and Sideslip Angle Control Through Single Input Single Output Direct Yaw Moment Control. *IEEE Trans. Control Syst. Technol.* **2021**, *29*, 124–139. [[CrossRef](#)]
5. Bengler, K.; Dietmayer, K.; Farber, B.; Maurer, M.; Stiller, C.; Winner, H. Three Decades of Driver Assistance Systems: Review and Future Perspectives. *IEEE Intell. Transp. Syst. Mag.* **2014**, *6*, 6–22. [[CrossRef](#)]
6. Guvenç, L.; Guvenç, B.A.; Demirel, B.; Emirler, M.T. *Control of Mechatronic Systems*; IET Control, Robotics and Sensors Series; The Institute of Engineering and Technology: London, UK, 2017; ISBN 978-1-78561-145-2.
7. Boyali, A.; Guvenc, L. Real-Time Controller Design for a Parallel Hybrid Electric Vehicle Using Neuro-Dynamic Programming Method. In Proceedings of the 2010 IEEE International Conference on Systems, Man and Cybernetics, Istanbul, Turkey, 10–13 October 2010; pp. 4318–4324.
8. Claussmann, L.; Revilloud, M.; Gruyer, D.; Glaser, S. A Review of Motion Planning for Highway Autonomous Driving. *IEEE Trans. Intell. Transp. Syst.* **2020**, *21*, 1826–1848. [[CrossRef](#)]
9. Wang, H.; Tota, A.; Aksun-Guvenc, B.; Guvenc, L. Real Time Implementation of Socially Acceptable Collision Avoidance of a Low Speed Autonomous Shuttle Using the Elastic Band Method. *Mechatronics* **2018**, *50*, 341–355. [[CrossRef](#)]
10. Guo, J.; Kurup, U.; Shah, M. Is it Safe to Drive? An Overview of Factors, Metrics, and Datasets for Driveability Assessment in Autonomous Driving. *IEEE Trans. Intell. Transp. Syst.* **2020**, *21*, 3135–3151. [[CrossRef](#)]

11. Kuwata, Y.; Teo, J.; Fiore, G.; Karaman, S.; Frazzoli, E.; How, J.P. Real-Time Motion Planning With Applications to Autonomous Urban Driving. *IEEE Trans. Control Syst. Technol.* **2009**, *17*, 1105–1118. [[CrossRef](#)]
12. Yurtsever, E.; Lambert, J.; Carballo, A.; Takeda, K. A Survey of Autonomous Driving: Common Practices and Emerging Technologies. *IEEE Access* **2020**, *8*, 58443–58469. [[CrossRef](#)]
13. Kuutti, S.; Fallah, S.; Katsaros, K.; Dianati, M.; McCullough, F.; Mouzakitis, A. A Survey of the State-of-the-Art Localization Techniques and Their Potentials for Autonomous Vehicle Applications. *IEEE Internet Things J.* **2018**, *5*, 829–846. [[CrossRef](#)]
14. Meneses-Cime, K.; Guvenc, L.; Aksun-Guvenc, B. Optimization of On-Demand Shared Autonomous Vehicle Deployments Utilizing Reinforcement Learning. *Sensors* **2022**, *22*, 8317. [[CrossRef](#)]
15. Kavas-Torris, O.; Lackey, N.; Guvenc, L. Simulating the Effect of Autonomous Vehicles on Roadway Mobility in a Microscopic Traffic Simulator. *Int. J. Automot. Technol.* **2021**, *22*, 713–733. [[CrossRef](#)]
16. Kavas-Torris, O.; Cantas, M.R.; Gelbal, S.; Guvenc, B.; Guvenc, L. Fuel Economy Benefit Analysis of Pass-at-Green (PaG) V2I Application on Urban Routes with STOP Signs. *Int. J. Veh. Des.* **2021**, *83*, 258–279. [[CrossRef](#)]
17. Yang, Y.; Ma, F.; Wang, J.; Zhu, S.; Gelbal, S.Y.; Kavas-Torris, O.; Aksun-Guvenc, B.; Guvenc, L. Cooperative Ecological Cruising Using Hierarchical Control Strategy with Optimal Sustainable Performance for Connected Automated Vehicles on Varying Road Conditions. *J. Clean. Prod.* **2020**, *275*, 123056. [[CrossRef](#)]
18. Kavas-Torris, O.; Guvenc, L. A Comprehensive Eco-Driving Strategy for Connected and Autonomous Vehicles (CAVs) with Microscopic Traffic Simulation Testing Evaluation. Available online: <https://arxiv.org/abs/2206.08306/> (accessed on 6 May 2023).
19. Kavas-Torris, O.; Cantas, M.R.; Cime, K.M.; Guvenc, B.A.; Guvenc, L. *The Effects of Varying Penetration Rates of L4–L5 Autonomous Vehicles on Fuel Efficiency and Mobility of Traffic Networks*; SAE International: Warrendale, PA, USA, 2020.
20. Guvenc, L.; Aksun-Guvenc, B.; Zhu, S.; Gelbal, S.Y. *Autonomous Road Vehicle Path Planning and Tracking Control*; John Wiley & Sons: Hoboken, NJ, USA, 2021; ISBN 978-1-119-74794-9.
21. Gelbal, S.Y.; Cantas, M.R.; Guvenc, B.A.; Guvenc, L.; Surnilla, G.; Zhang, H. *Mobile Safety Application for Pedestrians Utilizing P2V Communication over Bluetooth*; SAE International: Warrendale, PA, USA, 2022.
22. Gelbal, S.Y.; Aksun-Guvenc, B.; Guvenc, L. Collision Avoidance of Low Speed Autonomous Shuttles with Pedestrians. *Int. J. Automot. Technol.* **2020**, *21*, 903–917. [[CrossRef](#)]
23. Emirler, M.T.; Güvenç, L.; Güvenç, B.A. Design and Evaluation of Robust Cooperative Adaptive Cruise Control Systems in Parameter Space. *Int. J. Automot. Technol.* **2018**, *19*, 359–367. [[CrossRef](#)]
24. Zhu, S.; Gelbal, S.Y.; Aksun-Guvenc, B.; Guvenc, L. Parameter-Space Based Robust Gain-Scheduling Design of Automated Vehicle Lateral Control. *IEEE Trans. Veh. Technol.* **2019**, *68*, 9660–9671. [[CrossRef](#)]
25. Templeton, B. Waymo and Cruise Have Both Hit 1M Miles with No Driver, But Waymo Publishes Detailed Safety Data. Available online: <https://www.forbes.com/sites/bradtempleton/2023/02/28/waymo-and-cruise-have-both-hit-1m-miles-with-no-driver-but-waymo-published-detailed-safety-data/> (accessed on 13 April 2023).
26. Tierno, A.; Santos, M.; Arruda, B.; Rosa, J. Open Issues for the Automotive Software Testing. In Proceedings of the 2016 12th IEEE International Conference on Industry Applications (INDUSCON), Curitiba, Brazil, 20–23 November 2016; pp. 1–8.
27. Guvenc, L.; Guvenc, B.A.; Emirler, M.T. Connected and Autonomous Vehicles. In *Internet of Things and Data Analytics Handbook*; John Wiley & Sons, Ltd.: Hoboken, NJ, USA, 2017; pp. 581–595. ISBN 978-1-119-17360-1.
28. Pariota, L.; Coppola, A.; Di Costanzo, L.; Di Vico, A.; Andolfi, A.; D’Aniello, C.; Bifulco, G.N. Integrating Tools for an Effective Testing of Connected and Automated Vehicles Technologies. *IET Intell. Transp. Syst.* **2020**, *14*, 1025–1033. [[CrossRef](#)]
29. Alghodhaifi, H.; Lakshmanan, S. Autonomous Vehicle Evaluation: A Comprehensive Survey on Modeling and Simulation Approaches. *IEEE Access* **2021**, *9*, 151531–151566. [[CrossRef](#)]
30. Chen, S.; Chen, Y.; Zhang, S.; Zheng, N. A Novel Integrated Simulation and Testing Platform for Self-Driving Cars With Hardware in the Loop. *IEEE Trans. Intell. Veh.* **2019**, *4*, 425–436. [[CrossRef](#)]
31. Gelbal, S.Y.; Tamilarasan, S.; Cantas, M.R.; Guvenc, L.; Aksun-Guvenc, B. A Connected and Autonomous Vehicle Hardware-in-the-Loop Simulator for Developing Automated Driving Algorithms. In Proceedings of the 2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC), Banff, AB, Canada, 5–8 October 2017; pp. 3397–3402.
32. Emirler, M.T.; Uygan, I.M.C.; Gelbal, S.Y.; Gozu, M.; Boke, T.A.; Aksun-Guvenc, B.; Guvenc, L. 2016, Vehicle Dynamics Modelling and Validation for a Hardware-in-the-Loop Vehicle Simulator. *Int. J. Veh. Des.* **2016**, *71*, 191–211. [[CrossRef](#)]
33. Li, X.; Doss, A.C.A.; Guvenc, B.A.; Guvenc, L. Pre-Deployment Testing of Low Speed, Urban Road Autonomous Driving in a Simulated Environment. *SAE Int. J. Adv. Curr. Prac. Mobil.* **2020**, *2*, 3301–3311. [[CrossRef](#)]
34. Takeuchi, E.; Tsubouchi, T. A 3-D Scan Matching Using Improved 3-D Normal Distributions Transform for Mobile Robotic Mapping. In Proceedings of the 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems, Beijing, China, 9–13 October 2006; pp. 3068–3073.
35. Magnusson, M.; Vaskevicius, N.; Stoyanov, T.; Pathak, K.; Birk, A. Beyond Points: Evaluating Recent 3D Scan-Matching Algorithms. In Proceedings of the 2015 IEEE International Conference on Robotics and Automation (ICRA), Seattle, WA, USA, 26–30 May 2015; pp. 3631–3637.
36. Suzuki, R.; Kataoka, R.; Ji, Y.; Umeda, K.; Fujii, H.; Kono, H. SLAM Using ICP and Graph Optimization Considering Physical Properties of Environment. In Proceedings of the 2020 21st International Conference on Research and Education in Mechatronics (REM), Cracow, Poland, 9–11 December 2020; pp. 1–5.

37. Xu, S.; Zidek, R.; Cao, Z.; Lu, P.; Wang, X.; Li, B.; Peng, H. System and Experiments of Model-Driven Motion Planning and Control for Autonomous Vehicles. *IEEE Trans. Syst. Man Cybern. Syst.* **2022**, *52*, 5975–5988. [[CrossRef](#)]
38. Vehicle-in-Virtual-Environment Method for Autonomous Driving System Development and Evaluation. Available online: <https://oied.osu.edu/technologies/vehicle-virtual-environment-method-autonomous-driving-system-development-and-evaluation> (accessed on 13 April 2023).
39. Cime, K.M.; Cantas, M.R.; Dowd, G.; Guvenc, L.; Guvenc, B.A.; Mittal, A.; Joshi, A.; Fishelson, J. *Hardware-in-the-Loop, Traffic-in-the-Loop and Software-in-the-Loop Autonomous Vehicle Simulation for Mobility Studies*; SAE International: Warrendale, PA, USA, 2020.
40. Cime, K.M.; Cantas, M.R.; Fernandez, P.; Guvenc, B.A.; Guvenc, L.; Joshi, A.; Fishelson, J.; Mittal, A. Assessing the Access to Jobs by Shared Autonomous Vehicles in Marysville, Ohio: Modeling, Simulating and Validating. *SAE Int. J. Adv. Curr. Pract. Mobil.* **2021**, *3*, 2509–2515. [[CrossRef](#)]
41. Hussein, A.; García, F.; Armingol, J.M.; Olaverri-Monreal, C. P2V and V2P Communication for Pedestrian Warning on the Basis of Autonomous Vehicles. In Proceedings of the 2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC), Rio de Janeiro, Brazil, 1–4 November 2016; pp. 2034–2039.
42. Li, C.-Y.; Salinas, G.; Huang, P.-H.; Tu, G.-H.; Hsu, G.-H.; Hsieh, T.-Y. V2PSense: Enabling Cellular-Based V2P Collision Warning Service through Mobile Sensing. In Proceedings of the 2018 IEEE International Conference on Communications (ICC), Kansas City, MO, USA, 20–24 May 2018; pp. 1–6.
43. Tahir, M.N.; Mäenpää, K.; Hippi, M. Pedestrian Motion Detection & Pedestrian Communication (P2I & V2P). In Proceedings of the 2020 International Conference on Software, Telecommunications and Computer Networks (SoftCOM), Split, Croatia, 17–19 September 2020; pp. 1–3.
44. 802.11p-2010; IEEE Standard for Information Technology—Local and Metropolitan Area Networks—Specific Requirements—Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment 6: Wireless Access in Vehicular Environments. IEEE: New York, NY, USA, 2010. [[CrossRef](#)]
45. Gräfling, S.; Mähönen, P.; Riihijärvi, J. Performance Evaluation of IEEE 1609 WAVE and IEEE 802.11p for Vehicular Communications. In Proceedings of the 2010 Second International Conference on Ubiquitous and Future Networks (ICUFN), Jeju Island, South Korea, 16–18 June 2010; pp. 344–348.
46. Gheorghiu, R.A.; Cormos, A.C.; Stan, V.A.; Iordache, V. Overview of Network Topologies for V2X Communications. In Proceedings of the 2017 9th International Conference on Electronics, Computers and Artificial Intelligence (ECAI), Targoviste, Romania, 29 June–1 July 2017; pp. 1–6.
47. Wang, X.; Mao, S.; Gong, M.X. An Overview of 3GPP Cellular Vehicle-to-Everything Standards. *GetMobile Mob. Comp. Comm.* **2017**, *21*, 19–25. [[CrossRef](#)]
48. Janeba, M.; Lehoczky, P.; Galinski, M.; Milesich, T.; Danko, J.; Kotuliak, I. Evaluation of LTE and 5G Qualitative Parameters for V2X Use Cases. In Proceedings of the 2022 IEEE Zooming Innovation in Consumer Technologies Conference (ZINC), Novi Sad, Serbia, 25–26 May 2022; pp. 165–169.
49. Ahmed, M.S.; Hoque, M.A.; Khattak, A.J. Demo: Real-Time Vehicle Movement Tracking on Android Devices through Bluetooth Communication with DSRC Devices. In Proceedings of the 2016 IEEE Vehicular Networking Conference (VNC), Columbus, OH, USA, 8–10 December 2016; pp. 1–2.
50. Gheorghiu, R.A.; Iordache, V.; Cormos, A.C. Analysis of Handshake Time for Bluetooth Communications to Be Implemented in Vehicular Environments. In Proceedings of the 2017 40th International Conference on Telecommunications and Signal Processing (TSP), Barcelona, Spain, 5–7 July 2017; pp. 144–147.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.