machines

Review

A Survey on Data-Driven Scenario Generation for Automated Vehicle Testing

Jinkang Cai [1], Weiwen Deng [1], Haoran Guang [1], Ying Wang [2,*], Jiangkun Li [1] and Juan Ding [3]

1 School of Transportation Science and Engineering, Beihang University, Beijing 102206, China
2 School of Computer Science and Technology, Jilin University, Changchun 130015, China
3 PanoSim Ltd., Jiaxing 314000, China
* Correspondence: wangying_jlu@jlu.edu.cn

Abstract: Automated driving is a promising tool for reducing traffic accidents. While some companies claim that many cutting-edge automated driving functions have been developed, how to evaluate the safety of automated vehicles remains an open question, which has become a crucial bottleneck. Scenario-based testing has been introduced to test automated vehicles, and much progress has been achieved. While data-driven and knowledge-based approaches are hot research topics, this survey is mainly about Data-Driven Scenario Generation (DDSG) for automated vehicle testing. Rather than describe the contributions of every study respectively, in this survey, methodologies from various studies are anatomized as solutions for several significant problems and compared with each other. This way, scholars and engineers can quickly find state-of-the-art approaches to the issues they might encounter. Furthermore, several critical challenges that might hinder DDSG are described, and responding solutions are presented at the end of this survey.

Keywords: automated vehicles; scenario-based testing; Traffic Safety

check for updates

Citation: Cai, J.; Deng, W.; Guang, H.; Wang, Y.; Li, J.; Ding, J. A Survey on Data-Driven Scenario Generation for Automated Vehicle Testing. *Machines* **2022**, *10*, 1101. https://doi.org/10.3390/machines10111101

Academic Editor: Antonios Gasteratos

Received: 13 October 2022
Accepted: 17 November 2022
Published: 21 November 2022

1. Introduction

Traffic accident reduction has always been a hot topic in industry and academia [1–3]. Automated Driving (AD) is a promising approach to reducing accidents [4]. The rapid development of AD has been witnessed in the past decade. Some companies, such as Baidu, Tesla, and Audi, claim that they have developed high-level automated vehicles (AVs) [5,6]. Before being accepted by the market, high-level AVs must be thoroughly validated and evaluated to prove that they are safer than human driving, or at least as same [7]. However, testing AVs is still challenging [8]. On the one hand, AVs are becoming increasingly more complicated. For example, some Machine Learning (ML) technologies have been applied to AD for tasks including object recognition and classification and have achieved good results [9]. However, many of them, such as Convolutional Neural Networks (CNN), and Reinforcement Learning (RL), are not fully interpretable, which makes it significantly challenging to test AVs based on traditional methods, such as mileage-based testing [10,11]. On the other hand, there are many occasional scenarios in the physical world where AVs are required to drive [12]. The number of scenarios an AV might encounter in natural driving traffic is theoretically infinite. Ref. [13] indicates that millions of miles of road testing may be required to prove the reliability of a driverless vehicle, which is unfeasible and unaffordable.

Scenario-Based Testing (SBT) is a well-investigated and promising method for AV testing. There are many projects and initiatives about the SBT of AVs, including Pegasus [14], euroFOT [15], AdaptIVe [16], Sakura [17], StreetWise [18], and so on. SBT has already been utilized in software engineering [19,20]. The purpose of SBT is to prove that the System Under Test (SUT) can work as designed or at least safely without getting itself hurt or jeopardizing the safety of other traffic participants in the whole Operational Design Domain (ODD) [6].

One of the cornerstones of SBT is the scenario database, including diverse and critical scenarios. There are two kinds of methods to obtain it, data-driven or knowledge-based. The most significant difference between them is their reliance on expert knowledge. Knowledge-based methods usually require substantial expertise to generate needed scenarios, such as ontology-based methods [21,22]. On the contrary, data-driven methods generate scenarios primarily by exploiting information contained in source data. It is worth noting that "data-driven" does not mean expert knowledge is not needed in all relevant processes of scenario generation. Actually, specialist knowledge is utilized in almost all data-driven methods or as complementary to them, more or less [23,24]. While knowledge-based SBT has been proven to be a practical approach in some circumstances [22], it is limited to the domain designed by experts [25], and it cannot derive SUT performance in natural traffic [15]. Therefore, this survey focuses on Data-Driven Scenario Generation (DDSG).

Data-driven scenario generation for AV testing has become a hot topic, and much progress has been made. It would be helpful for researchers and engineers to read surveys summarizing critical points of relevant studies, especially the ones published in the past three years. A problem-oriented survey is provided to make it easier for researchers and engineers to find solutions for their problems of data-driven scenario generation quickly. Solutions for the same issues are grouped and compared. For example, dimension reduction techniques, such as Principal Component Analysis (PCA), and t-distributed Stochastic Neighbor Embedding (t-SNE), are employed in many studies. A comparison of them could help scholars and engineers make the best choice. It is worth noting that while this survey is mainly about data-driven approaches for scenario generation, some studies taking advantage of expert knowledge to enhance data-driven scenario generation are also considered in this survey.

There are already some reviews about SBT of AVs [12,26–28]. SBT of AVs is one of the topics of [28], but the latest literature is not considered. The approaches for SBT are taxonomized in [26] without some essential topics not discussed in detail, such as the improvement of the original Accelerated Evaluation (AE) [29] and the construction of metrics considering several aspects. In [27], methods for SBT are categorized into three classes: coverage-oriented, unsafe-scenario-oriented, and indicator-estimation oriented. And these methods are analyzed based on simulation results. However, most of these reviews group related studies based on the overall methodologies rather than the specific problems they try to solve.

The motivation of this paper is to introduce, analyze, and compare state-of-the-art methodologies for Data-Driven Scenario Generation (DDSG) and solutions to related sub-problems. Furthermore, pointing out some remaining problems in DDSG. Scholars and engineers can make the best choice for AV testing among the existing methods or dig deeper to tackle the remaining problems in scenario-based testing (SBT). Therefore, the contributions of this survey include four parts:

1. State-of-the-art methodologies used for DDSG, such as Reinforcing Learning (RL), Accelerated Evaluation (AE), and so on, are generally introduced. The generation of customized scenarios for the VUT is also covered by this survey, which cannot be found in existing reviews.
2. Solutions to sub-problems involved in these methodologies are described in detail. These sub-problems include source data collection, scenario identification, and criticality metrics used for scenario evaluation.
3. Some remaining problems are pointed out, and responding potential solutions are provided.

2. Framework

Some crucial problems must be solved to generate needed scenarios, including collecting authentic source data, identifying interested scenarios hidden in source data, generating scenarios for AV testing, evaluating derived scenarios, and so on. Since methodologies for

DDSG might diverge a lot from each other, based on these to-be-solved problems, a typical framework of DDSG is obtained as shown in Figure 1, which includes mainly four steps:

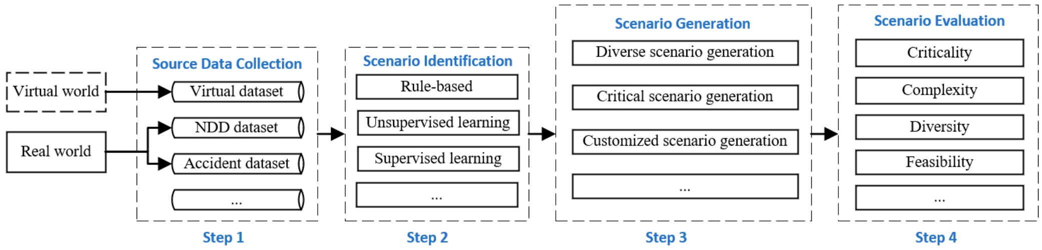

Figure 1. A typical framework of the data-driven scenario generation methodology.

Step 1: Source Data Obtaining. DDSG mainly relies on information extracted from data instead of expert knowledge. Therefore, source data is crucial to DDSG [26]. Source data can be collected through physical or virtual approaches. The former includes collecting Natural Driving Data (NDD) and accident data et al. Sometimes, data from the real world is integrated into relevant laws, regulations, or standards, which can also be used as data sources. With the developing of simulation techniques, the fidelity of the virtual world used for AV testing is improving gradually, leading to the increasing importance of the later ones [30,31].

Step 2: Scenario Identification. Scenario identification aims to identify or detect scenarios that have already happened and are hidden in source data. Based on these identified scenarios, parameter ranges and distributions of logical scenarios can be elicited. On the other hand, some accident scenarios can be directly used for AV testing, such as accident and near-collision scenarios [32].

Step 3: Scenario Generation. Many companies have collected a large amount of NDD [28]. However, on the one hand, the number of scenarios existing in the real world is theoretically infinite. It is almost an impossible mission to collect enough data in which all possible parameter combinations in ODD can be directly identified [33]. Moreover, most scenarios recorded by NDD are dull, and critical scenarios are rare [15]. As for accident databases, the diversity of identified scenarios cannot be secured. Therefore, it is necessary to generate scenarios that are rarely observed in the natural world but are critical for AVs.

Step 4: Scenario Evaluation. Based on pre-defined metrics, the quality of scenarios can be measured, which will be employed to quantify SUT performance. This survey focuses on the criticality of scenarios.

After analyzing hundreds of studies, this survey focusses on several hot topics, including source data collection, scenario identification, diverse scenario generation, critical/challenging scenario generation, customized scenario generation, and criticality metrics of scenarios. The remains of this survey are arranged as follows: Section 2 provides some basic term definitions. Several source data collection methods are described in Section 3. Sections 4 and 5 introduce essential procedures and methodologies for scenario identification and generation, respectively. At the end of Sections 3–5, conclusions of relevant sections are made. Finally, conclusions about existing methodologies for DDSG are made, and several challenges are pointed out in Section 6.

3. Definitions

3.1. Scene, Scenario, and ODD

Although many studies about SBT of AVs exist, the definitions of scenario and scene remain open [34]. Considering rationality and popularity, relative term definitions in Pegasus [14] are adopted in this survey. In Pegasus, a scenario is a temporal sequence of scenes. A scene is a snapshot of the environment, including dynamic entities such as vehicles and pedestrians and static entities such as traffic signs and lights [35].

Based on the level of abstraction, scenarios can be classified into three categories, functional, logical, and concrete scenarios [20]. Functional scenarios are described by natural languages, such as cut-in and car-following scenarios, and have the highest level of abstraction. Parameter ranges and distributions can be found in logical scenarios, which have less abstraction than functional scenarios. Each parameter has an exact value in concrete scenarios, meaning concrete scenarios are less abstract than logical and functional scenarios. Concrete scenarios selected to execute are called test cases [36]. If with no specific notation, scenarios mentioned in the following refer to concrete scenarios.

ODD is the operational design domain where the SUT should work as designed [6]. For a better description of ODD, a 5-layer model [22] is developed for highway scenarios, which includes Road-level (Layer 1), Traffic Infrastructure (layer 2), Temporary Manipulation of L1 and L2 (layer 3), Objects (Layer 4), and Environment (Layer 5). Furthermore, [37] extends it by adding the sixth layer, Digital Information. Furthermore, a modified 6-layer model is presented in [38] to describe traffic scenarios in urban appropriately, which includes Road Network and Traffic Guidance Objects (Layer 1), Roadside Structures (Layer 2), Temporary Modification of L1 and L2 (Layer 3), Dynamic Objects (Layer 4). Environment Conditions (Layer 5), Digital Information (Layer 6).

3.2. Critical, Challenging Scenarios

There are many terms used in different studies to describe scenarios with different characteristics, including critical scenarios, challenging scenarios, boundary scenarios, complex scenarios, corner cases, edge cases, etc. However, there are no standard definitions for them now. While usually applied as synonyms [39–41], differences between these terms can be found in some studies.

The term criticality can be found in many studies (see Section 7), but its definition in SBT remains an open question. In [42], critical metrics are surrogate measures for analyzing conflict potential or the severity of microscopic scenarios. In [43], criticality indicates the temporal or spatial closeness to a potential collision in a driving scene/scenario, or the magnitude of the dynamic driving reaction required to prevent an accident. Furthermore, a critical metric quantifies the criticality of a scene or scenario. Since the definitions described in [43] are more comprehensive, this survey adopts them. In terms of critical scenarios, the definition provided by [44] is adopted: critical scenarios are scenarios in which the ego vehicle leads or nearly leads to collisions. The criticality of a scenario can only be obtained after scenario execution.

Challenging scenarios and complex scenarios are usually used as alternatives to each other [44]. In [45,46], challenging and complex scenarios refer to significantly difficult scenarios for ego vehicles to pass safely. They insist that how challenging or complex a scenario is can be determined before scenario execution. Ref. [47] regards scenarios that are very difficult to master as challenging scenarios and assumes that an increasing difficulty of a scenario leads to a surge of failure possibility of the SUT. In this survey, challenging or complex scenarios are the ones that are challenging for the SUT to pass without directly or indirectly leading to a crash. It is worth noting that challenging or complex scenarios are not always critical, which can only be determined after the execution.

In [28], boundary scenarios are those whose execution results are in the proximate area around the boundary between safe and unsafe. Behavior mode boundaries are considered guardians for searching critical scenarios in [48]. As for corner and edge cases, [49] believes only corner cases with unusual or novel conditions can be considered edge scenarios. In this survey, the definition presented by [27] is adopted, which is that boundary scenarios/cases are the ones that exist around the performance boundaries of the SUT, around which a small change to scenario parameters might lead to significantly different execution results, such as leading to a collision or not. Corner and edge scenarios/cases are the ones that are extremely rare in the real world. The slight difference between corner cases and edge cases is ignored. For more term definitions used for SBT, [49,50] are advised.

4. Source Data Collection

The authenticity and coverage of source data are crucial to the quality of a scenario database. Data from the physical world, such as Natural Driving Data (NDD) and accident datasets, are exploited in many studies. In the meantime, some studies use simulation-based methods to acquire source data. This section introduces methods of obtaining NDD, accident data, and virtual data, and some conclusions are given at the end of this section.

4.1. Natural Driving Data

Natural Driving Data (NDD) is traffic data collected in the physical world. There are two main approaches for NDD collection: the floating vehicle-based and the fixed sensor-based. Floating vehicles (FVs) are usually equipped with many sensors, such as lidars, millimeter-wave radars, inertial navigation systems, global positioning systems, and cameras. Theoretically, FVs can collect NDD at any place where FVs can go at any time. By fusing data from different sensors, high-quality data can be collected. Some big companies have gathered lots of NDD using FVs, including Baidu, Waymo, Volvo, and so on [51]. Well-founded projects such as Pegasus, SPMD project [52], et al. also have their own NDD databases. However, it is costly to maintain a large fleet of FVs. Due to the tremendous investment required, many related companies or institutes do not share/publicize the source data, such as Pegasus and TNO [53].

In [54], floating vehicles equipped with two laser scanners, two front cameras, and IMU/GUSS are used to collect traffic data in China, as shown in Figure 2. 3D-semantic labels and 3D-bounding boxes in ApolloScape dataset make it a hot dataset for object detection and orientation estimation algorithms. Since this dataset includes many urban traffic scenes, it can also be used to generate scenarios for AV testing.

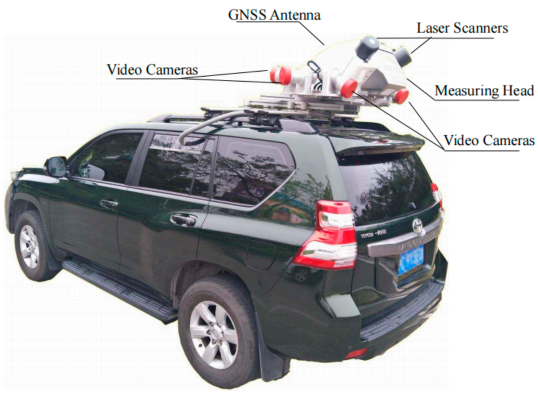

Figure 2. A floating vehicle used for NDD collection [54].

Fixing sensors on roadside infrastructures or drones to collect NDD is a comparatively more affordable method. Drones equipped with cameras are mentioned in many studies [55–58]. With the help of computer vision technology, trajectories of traffic participants can be extracted precisely. In the highD dataset [55], mean errors of lateral and longitudinal positions between computer-based and human-based methods are less than 0.03 m. However, compared with FV-based methods, the observed area and duration of the fixed sensor-based method are pretty limited and are sensitive to harsh circumstances. For example, if cameras are the only employed instruments, extreme weather, such as foggy or rainy, could make it significantly challenging to extract accurate NDD from the recorded videos. However, if several types of sensors are applied for data collection, the negative influence of unfavorable weather can be mitigated.

In [59], to collect traffic data in an intersection in Aschaffenburg, Germany, 14 8-layer lidars and eight cameras with different viewpoints are installed on traffic lights and lamp posts more than 5 m above the ground.

Traffic scenarios in NDD are naturally feasible, and SUT performance in natural traffic can be derived from that in scenarios generated based on NDD (see Section 5). Since the number of scenarios that might happen in natural driving traffic is almost infinite, an enormous amount of NDD is needed for AV testing. However, it requires heavy investments to collect NDD based on FVs, while fixed sensors can only monitor a specific area. While many large-scale NDD datasets are not publicly available, some institutes share their NDD data for non-commercial purposes. In [12], 25 public NDD datasets are compared in several aspects, including bird's-eye or first-person view, containing or not containing data in different weather, notation types, etc. Ref. [60] summarizes 37 NDD datasets and 22 kinds of commercial or simulation software for AV testing. For the completeness of this section, 27 traffic datasets are provided in Table 1, including the latest ones.

Table 1. 27 NDD dataset. "Trajectory" indicates if trajectories of all traffic agents are explicitly available in the dataset.

Number	Dataset	OpenSource	Method	Sensor	Trajectory
1	100-car [61]	Yes	FV-based	Camera, GPS, radar	No
2	A*3D [62]	Yes	FV-based	Lidar, Camera	No
3	ApolloScape [54]	Yes	FV-based	Camera, Lidar, GPS/IMU	No
4	Argoverse [63]	Yes	FV-based	Lidar, Camera	Yes
5	Bdd100k [64]	Yes	FV-based	Camera, Lidar, GPS/IMU	No
6	CamVid [65]	Yes	FV-based	Lidar, Camera	No
7	Cityscapes [66]	Yes	FV-based	Camera	No
8	CitySim [67]	Yes	Fixed sensor-based	Camera	Yes
9	Five Roundabouts [68]	Yes	Fixed sensor-based	Lidar, Camera	Yes
10	H3D [69]	Yes	FV-based	Cameras, LiDAR and GPS/IMU	No
11	InD [56]	Yes	Fixed sensor-based	Camera	Yes
12	INTERACTION [70]	Yes	Fixed sensor-based	Camera	Yes
13	KAIST [71]	Yes	FV-based	Camera, Lidar, GPS/IMU	No
14	KITTI [72]	Yes	FV-based	Camera, Lidar, GPS/IMU	No
15	Ko-PER [73]	Yes	Fixed sensor-based	Lidar, Camera	Yes
16	Lyft Level 5 [59]	Yes	FV-based	Lidar, Camera	No
	NGSIM [58]	Yes	Fixed sensor-based	Camera	Yes
17	nuScenes [59]	Yes	FV-based	Radar, Lidar, Camera, GPS/IMU	No
18	Oxford RobotCar [74]	Yes	FV-based	Camera, Lidar, GPS/IMU	No
19	RondD [57]	Yes	Fixed sensor-based	Camera	Yes
20	SPMD [52]	Yes	FV-based	VAD, ASD, RSD, et.al.	No
21	Stanford Drone [75]	Yes	Fixed sensor-based	Camera	Yes
22	BDDDD [76]	Yes	Fixed sensor-based	Camera	Yes
23	TRAF [77]	Yes	FV-based	Camera	Yes
24	TDCDBD [78]	Yes	FV-based	Camera	No
25	TAF-BW [79]	Yes	FV-based	Camera	Yes
26	Udacity [80]	Yes	FV-based	Camera	No
27	Waymo Open [81]	Yes	FV-based	Cameras, LiDAR and GPS/IMU	No

4.2. Accident Data

Each accident scenario identified in accident databases involves at least one crash, which means accident scenarios are naturally challenging to AVs [82]. Theoretically, all approaches of NDD collection can be applied to accident data gathering. In [83], many accident data collection methods adopted in developed and developing countries are introduced and compared. Considering the rarity of accidents and the affordability of relevant instruments, accident reports and videos are most widely used for accident data extraction [84,85]. However, accident reports or videos might ignore some essential details, which means some remedies might be necessary [86]. Many countries have invested a lot in accident data collection, resulting in many famous accident databases, such as

CIDAS [87] and NAIS [88] from China, GIDAS [89] from German, GES [90] from the US, ASSESS [91] from Europe, OTS [92] and STATS19 [92] from the UK. Several accident datasets are analyzed in [93].

However, as shown in Table 2, some valuable accident datasets are not available for most researchers, such as In-depth accident databases from German and China [89,94]. The Crash Investigation Sampling System (CISS) database of the National Highway Traffic Safety Administration is the only in-depth accident database freely available to the public [95]. Moreover, accident scenarios only fill a small part of ODD, indicating that they cannot be leveraged to prove AV safety in the whole ODD.

Table 2. 14 Accident datasets. "In-Depth" indicates an in-depth accident dataset.

Number	Dataset	Open Source	In-Depth	Region	Source of Raw Data
1	US-Accidents [96]	Yes	No	USA	MapQuest Traffic and Microsoft Bing Map Traffic
3	CIDAS [97]	No	Yes	China	accident report
4	Dubai [98]	No	No	Dubai	accident report
5	GIDAS [99]	No	Yes	German	accident report
6	KIDAS [100]	No	Yes	Korea	accident report
7	Korean Freeway [101]	No	No	Korea	accident report
8	NAIS [88]	No	Yes	China	accident report
9	GES [67]	Yes	Yes	USA	accident report
10	OSM [102]	Yes	No	Global	accident report
11	SHUFO [103]	No	Yes	Shanghai	accident report
12	Singapore [104]	No	No	Singapore	accident report
13	UKIDAS [105]	No	Yes	UK	accident report
14	CADP [106]	Yes	No	Global	Video

4.3. Virtual Data

In the virtual world, the positions and behaviors of traffic participants can be manipulated as desired, and complex scenarios can be produced at a low cost. With the advancement of simulation technology, it becomes possible to gather virtual but high-fidelity data in simulation experiments.

Microscopic traffic simulation is an efficient method to quickly get a large amount of virtual data for DDSG. In [31], accident scenario data is obtained by microscopic traffic simulation on SUMO [107]. To create a more realistic traffic circumstance, a digital twin of the natural static environment in a district is reconstructed on Unity in [108]. Then virtual vehicles, pedestrians, and other traffic participants are added to the virtual world to generate high-fidelity traffic scenarios. Reviews of traffic simulation technologies and popular simulators can be found in [109,110].

The behavior models employed in microscopic traffic simulations are usually conservative and lack complicated interactions among each other, which are significantly essential for AVs [111]. To this end, it is an excellent option to replace the crucial virtual traffic participants (such as drivers, pedestrians, et al.) with real ones while other elements remain virtual. To investigate the interaction between human drivers and automated vehicles in dilemma areas, [30] equips human drivers with Virtual Reality (V.R.) instruments to immerse them in the virtual world. Similarly, in [112], Mixed Reality (M.R.) technology is adopted to collect pedestrian-vehicle interaction data. Augmented Reality (AR) instruments are integrated with a scenario generation methodology in [113] to simulate a natural traffic environment. However, simulation efficiency is somewhat limited since not all elements in these experiments are virtually and automatically generated.

There are several advantages of generating source data based on the simulation environment. First, the generation of virtual traffic data is more time efficient. Different from collecting traffic data in the real world, the number of virtual traffic data generated within one second could be significant given sufficient computing resources. Second, it is less labor-intensive. Third, semantic information is 100% right, which is of great value for the perception system test of an AV. However, there might always be a simulation-to-real gap [114]. Rendering realistic traffic scenarios will be a hot research topic in the next decade.

4.4. Conclusions of Source Data Collection

This section introduces methods for collecting NDD, accident data, and virtual scenario data. NDD is naturally feasible, and SUT performance in natural traffic can be derived based on SUT performance in scenarios generated based on NDD (see Section 6). However, it is not easy to gather enough NDD without heavy investment. Accident data can be used to extract accident scenarios. Each of these accident scenarios involves at least one accident and is challenging for AV testing. Considering the efforts needed for collecting scenario data in the real world, it would significantly contribute to the community if big companies share their traffic data, such as Alphabet, Tesla, Didi, and Baidu. It must be pointed out that it is not possible or feasible to identify scenarios that cover all space of the ODD only using data collected in the physical world.

In the virtual world, almost all elements needed for a scenario, including behaviors of all traffic participants, weather, roadside buildings, and so on, can be simulated and manipulated. With the development of simulation technologies, the efficiency of generating high-fidelity scenario data is increasing continually. It is reasonable to believe that virtually generated data would play an increasingly significant role in DDSG.

5. Scenario Identification

Scenarios identified in source data can be applied to derive ranges and distributions of scenario parameters. The critical ones among them can be directly used for the SBT of AVs. For example, critical scenarios generated by adding noise to accident scenarios are utilized by Waymo to test AVs [32]. Highly efficient scenario identification methods are needed to mind a large amount of source data. This section summarizes studies about scenario identification, and a comparison of different approaches is made at the end of this section. It is worth noting that while the topic of this survey is DDSG, considering rule-based methods can be used to label data for scenario identification based on Unsupervised Machine Learning (UML), studies about rule-based methods are also considered in the section.

5.1. Region of Interest

Before scenario identification, spatial and temporal ranges of engaging scenarios should be determined to ignore factors having almost no influence on ego and eliminate unnecessary data. However, a scenario's duration and Region Of Interest (ROI) are still open to discussion. Ref. [26] suggests that a duration of about 10 seconds is enough for most scenarios.

For the ROI of highway scenarios, an eight-vehicle model, as shown in Figure 3, is proposed only to consider vehicles near ego [115]. For specific values of L_f and L_b, 60 m is advised in [115]. Ref. [116] argues that ROI should be relative to the longitudinal velocity of ego, and an ROI considering safety distance is presented, as shown in Figure 4. A time gap of 1.8 s is suggested in [45,117]. Ref. [118] holds similar opinions and argues that remaining lateral distance should also be considered.

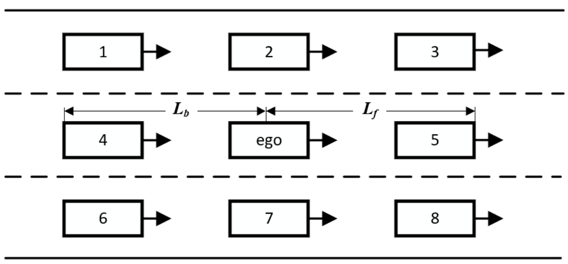

Figure 3. DOI based on distance [115].

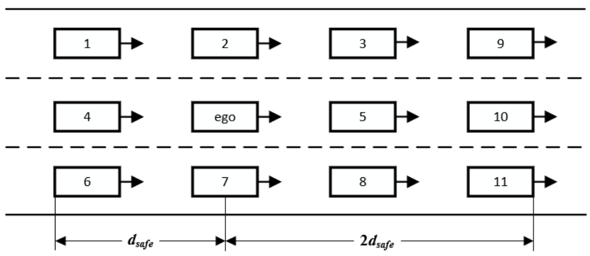

Figure 4. DOI based on safety distance [116].

5.2. Feature Dimension Reduction

For different scenarios number of parameters may vary a lot. To evaluate the performance of AVs in the whole ODD, complicated scenarios involving multiple parameters are necessary. Dimension reduction is essential to overcome dimension explosion and make it more feasible to identify or generate scenarios based on features in the latent space in the following steps. While there are many techniques for DR [115], three of them are frequently used for DDSG, including Principal Component Analysis (PCA) [119], Singular Value Decomposition (SVD) [120], t-Distributed Stochastic Neighbor Embedding (t-SNE) [121].

PCA is a linear analysis technique that transfers original parameters to independent principal components by mapping parameters from the original to a latent space without losing too much variance [115]. PCA is widely adopted because it is robust and costs a low level of computational power [122]. Inspired by PCA, [123] proposes Principal Feature Analysis (PFA) which maps the most important features rather than all features to the latent space. In [124], PCA is applied to transform high-dimensional trajectories to low-dimensional feather vectors, and then the noise is added to the feather vectors to generate critical scenarios. However, since PCA works based on the contribution of parameters to the variance of the result, it belongs to a statistical method. This usually makes the feather vector not physically interpretable [125].

While SVD is one of the key parts of PCA, SVD itself can also be applied for dimension reduction. To reduce dimensions for scenario description, SVD is adopted in [126] to map the original scenario parameters to a space with fewer dimensions. Furthermore, the Possibility Density Function (PDF) of the resulting low-dimension parameters is estimated by Kernel Density Estimation (KDE) and sampled to generate new scenarios. Although PCA and SVD are both linear analysis techniques without supervision, there is a significant difference between them [127]. The goal of PCA is to maximize the variance of the original parameters, while SVD aims to minimize reconstruction error. Furthermore, PCA needs no iteration, while SVD does.

t-SNE is a visualization and dimension reduction tool based on possibility differences [128]. High-dimension parameters are mapped to a lower-dimension (usually 2 or 3) space by t-SNE without losing significant structure in the original data. In other words, adjacent points in the original space are still near each other with a high possibility in the new space, which PCA cannot achieve. To cluster trajectories in different scenarios, [129] combined Density-Based Spatial Clustering of Applications with Noise (DBSCAN) [130] with PCA, SVD, and t-SNE, respectively. Simulation results show t-SNE based DBSCAN outperforms the others. However, every DR method has its edges. For more details about dimension reduction, [131] are suggested.

5.3. Rule-Based Methods

Rule-based methods identify scenarios mainly based on pre-defined constraints or rules, such as thresholds, parameter ranges, etc. For simple scenarios that involve no complicated maneuvers or many participants, several thresholds or rules defined by experts might be enough for scenario identification. Based on pre-defined thresholds of decelerations, the lateral distance between ego and lane lines, or the time before or after the maneuver, braking, turning, lane-change, and cut-in scenarios are identified in [23,132,133].

In Rule-based approaches designed to identify complicated scenarios, scenarios are usually interpreted as combinations of elementary blocks, which could be states, events, activities, or maneuvers. Different temporal sequences of these building blocks belong to different function scenarios [134–136].

In [137], activities are the elementary blocks of scenarios. There are three lateral activities (acceleration, cruising, and deceleration) and three longitudinal activities (lane following, turn left, and turn right). Every trajectory can be regarded as a combination of these activities. Rules-based scenario templates are designed by experts and include the relative locations of traffic participants and their activities and speed. Gap-closing and cut-in scenarios in NDD are detected. The authors of [137] claim that this approach can identify all scenarios in NDD. But no example involving complex maneuvers is provided.

In [138], a maneuver is defined as the intentional transfer between different states of a participant. A layered maneuver model for urban scenario identification is designed, which includes vehicle state maneuvers, infrastructure maneuvers, and object-related maneuvers. Vehicle state maneuvers include acceleration, keeping velocity, deceleration, and reversing. Infrastructure maneuvers include following lanes, lane-change, approaching a junction, and so on. Object-relate maneuvers include the following and approaching objects, et al. Different combination patterns of these maneuvers are categorized into different functional scenarios. A field experiment of left turning in a junction is presented to validate this methodology. Then random sampling is used to generate scenarios based on the scenario parameter ranges extracted from the field data. However, it is difficult for this method to detect scenarios involving maneuvers not considered in the pre-defined maneuver model.

Rule-based models are enhanced by an unsupervised clustering technique in [45]. First, data in highD is clustered by a Hierarchical Clustering Algorithm (HCA) to identify scenarios that include at least one challenger. Second, the maneuvers of challengers are classified by a rule-based decision tree. Third, based on trajectories, the maneuvers of challengers are classified into one of the six functional scenarios in the PEGASUS Project. This method is scenarios in highD dataset, and 67,455 concrete scenarios belonging to the pre-defined six functional scenarios are extracted.

The studies mentioned above have shown that the rule-based method can effectively identify scenarios in source data. However, the rules designed by different experts might vary, and it demands a lot of expertise, and complicated scenarios are still challenging for rule designers. Moreover, it is difficult for rule-based methods to identify scenarios that are unknown to experts, which, to some extent, limits the diversity of the resulting scenarios.

5.4. Unsupervised Machine Learning

Rule-based methods are quite effective in detecting known scenarios with pre-defined rules. However, more powerful tools are needed to search for scenarios that have unknown patterns or are too complicated to be described by rule-based models. Machine learning is an effective tool for pattern recognition and has been applied in many fields [139]. Unsupervised Machine Learning (UML) identifies patterns without expert knowledge. The methods frequently used for scenario identification are unsupervised clustering techniques, such as K-Means, K-Medoids, and Hierarchical Clustering Algorithms (HCA) [45,115,140,141]. There are two major problems with detecting scenarios using UML. First, the durations of scenarios from the same function or logical scenario might vary. Second, some important features might hide in a latent instead of explicit space.

For the first problem, techniques that can transfer time series to feature vectors are advised, such as Dynamic Time Warping [115] and Variational Autoencoder [142]. Ref. [142] maps four parameters of car-following scenarios, including the speed of the ego and the following vehicle, car-following distance, and ego acceleration, to a 2-dimensional latent space by Variational Autoencoder (VAE), which can process temporal sequences. Furthermore, the distribution of one of the two latent variables, named z, is analyzed based on NGSIM and highD, respectively. Results of numerical experiments indicate that z contains enough information for scenario identification.

The first problem can also be solved by decomposing the scenario as a combination of states and actions. In [53], signals are sliced to search event or activity indicators such as extrema, inflections, or saddle points. Then, signal slices are clustered by an agglomerative HCA. This method can be used for data labeling. A similar approach can also be found in [141].

For the second problem, source data are usually mapped to a latent space and then processed by UML to extract information in the latent space. The combination of a dimension reduction algorithm and a clustering algorithm can be found in [115,143–145]. In [145], PCA and K-Means are used to identify representative traffic scenarios in 72,336-h urban-motorway NDD. First, all source data are divided into two datasets based on the driving direction. Then, each dataset is processed by PCA, and two principle components accounting for more than 90% variance of all eigenvalues for each dataset are obtained. Finally, K-Means is applied to cluster similar scenarios in the latent space generated by PCA. Four representative scenarios are identified.

The introduction and analysis above show that, based on UML techniques, especially clustering techniques, implicit and explicit patterns of scenarios can be automatically extracted for scenario identification. Unlike rule-based methods, UML-based ones could obtain unknown scenarios, contributing to the coverage of the resulting scenarios. However, some patterns identified by UML might be uninterpretable, resulting in unreasonable results.

5.5. Supervised Machine Learning

Supervised Machine Learning (SML) techniques, such as Support Vector Machine (SVM), Recurrent Neural Networks (RNN), can identify scenario patterns based on labeled source data. In [25], scenarios are represented as state sequences. Rules are designed to label data with state tags. Different tag sequences represent different scenarios. Tag sequences can be used as scenario templates to identify scenarios in source data. The supervised classifier trained by a small amount of labeled data could label other unlabeled data automatically. In [146], A rule-based method is applied to tag source data with state labels, and logical scenarios are described as label sequences. An RNN model is trained by these labeled scenario data and then exploited to detect scenarios in unlabeled source data. This methodology is evaluated by a case of detecting lane-change scenarios in an NDD dataset [146].

Unlike rule-based and UML-based methods, SML combines expert knowledge with machine learning techniques, achieving high efficiency in scenario identification without losing too much interpretability. It is worth noting that with inappropriate labels, SML cannot obtain reasonable scenarios, like UML methods.

5.6. Conclusions of Scenario Identification

This section introduces solutions to three problems: filtering out unnecessary scenario data, reducing feature dimensions, and identifying scenarios recorded in source data. Limiting the spatial and temporal ranges allows scenario data of interest to be extracted from the source data. After that, three widely used dimension-reducing techniques are introduced and compared for high-dimensional scenarios. Finally, Rule-based, UL-based, and SL-based methods are introduced.

Rule-based, UML, and ML techniques are described for the last problem. Rule-based methods identify scenarios relying on rules designed by experts, which means that the performance of rule-based methods highly depends on experts. It would be pretty tricky to achieve good results for complex scenarios without time for adjusting the rules. UML techniques can identify previously unknown scenarios with no expert knowledge. However, the results obtained by UML might be uninterpretable or even feasible. SML-based methods can be regarded as a tradeoff between rule-based and UML-based methods. The training data for SML must be labeled, which can be achieved with or without expert knowledge. And implicit patterns in the training data can be extracted by SML. But UML-based methods also have similar disadvantages compared with rule-based and UML-based ones. Experts may be needed for data labeling, and the resulting scenarios may be unreasonable.

To make good use of source data, it will be a trend to apply a rule-based method to identify known scenarios and exploit UML to find unknown scenarios. After that, a feasibility analysis is needed to filter out unreasonable scenarios. Furthermore, based on all identified scenarios, some labels, such as maneuver types, road topologies, weathers, et al., could be tagged to a certain amount of the source data. Finally, A high-performance scenario detector is trained based on the labeled data and applied to detect all other known or unknown scenarios.

6. Scenario Generation

6.1. Diverse Scenario Generation

Coverage and criticality of scenarios are essential to prove AVs' safety in ODD. Scenarios covering large spaces in ODD can be used to test AVs in various circumstances. Critical scenarios can quickly find faults in the SUT, helping developers and engineers improve AVs' safety. Almost all scenarios detected in accident databases are challenging, but their diversity is limited. Therefore, it is necessary to generate diverse, critical, and customized scenarios for the SBT of AVs.

6.1.1. Random Sampling

Random sampling can generate concrete scenarios based on parameter ranges and distributions in logical scenarios. Monte Carlo might be the most popular random sampling technique applied for DDSG. Theoretically, random sampling can be used to produce all possible concrete scenarios in ODD if logical scenarios are available.

A 60-hour traffic video is recorded in [15] to generate car-following scenarios. Based on car-following scenarios identified from the recorded video, parameter distributions are fitted using Kernel Density Estimation (KDE). These parameters are velocity reduction, total braking time, and end velocity. Ten thousand concrete scenarios are generated by random sampling on parameter distributions.

The most significant advantage of DDSG based on NDD using random sampling is that performance of the SUT in natural traffic can be derived based on SUT performance in the generated scenarios, if the distributions of scenario parameters are given [147]. However, it is significantly inefficient to execute numerous scenarios while only a small proportion of them is critical [148]. For this reason, many studies use random sampling as a baseline to show the superiority of other methodologies, such as Reinforce Learning and Accelerated Evaluation [29,147,148].

6.1.2. Combinatorial Testing

Combinatorial Testing (CT), also named N-wise testing, is a widely used tool in software testing in which different value combinations of inputs are regarded as different scenarios [149]. CT is proposed based on the hypothesis that most errors in a system happen because of the interaction of influencing factors [150]. Kuhn and Reilly analyzed test reports and found that 70% of errors of Mozilla are caused by the interaction of two factors and 90% by the interaction of three factors [151]. While CT can be implemented for knowledge-based or data-driven scenario generation [21,84,152–154], The latter is the topic of this section.

CT is utilized in [153] to find errors in the Lane Deviation Warning (LDW) system. First, influencing factors are parameterized, and all continuous parameters are discretized. Second, scenarios are generated based on CT with the different significance of parameters being considered. A metric named "complex index" is defined to measure the complexity of each scenario. Third, concrete scenarios generated by CT are clustered by an HCA based on weighted distance. Scenarios in each cluster are stitched together, deriving several continuous scenarios. Finally, the continuous scenarios are executed on a virtual test platform. Simulation results show that the more complex a scenario is, the more possible errors of the SUT are found, and for a scenario with a complexity of 0.3, there is more than a 70% chance of making the LDW system fail.

CT can generate complex scenarios given parameter ranges, and the N-wise coverage metric can be applied to measure the coverage of the obtained scenario database. However, CT can only generate scenarios based on discrete values, which means parameters with continuous ranges must be discretized. For high-dimension scenarios, this might lead to a dimension explosion. Therefore, CT is mainly adopted to generate scenarios for low-level AVs.

6.1.3. Mutation Testing

Mutation Testing (MT) is a method for evaluating the quality of test sets. In MT, mutants of original test cases are generated based on pre-defined mutation operators. One original test case can derive one or more mutants. If SUT performance in a mutant scenario differs significantly from that in the original test case, the mutant will be killed. If not, the mutant stays alive. The smaller the proportion of the killed mutants among all mutants is, the better the original test set is.

Ref. [155] is a typical study of generating scenarios based on MT and Feature-Interaction-Coverage-Sampling (FIC-Sampling), as shown in Figure 5. The methodology in [155] can be divided into eight steps. In step 1, scenario templates named feature models are designed. Test cases can be obtained by modifying parameters in scenario templates. In step 2, FIC-Sampling is adopted to select parameters for concrete scenarios. FIC-Sampling is a technique similar to CT. For convenience, scenarios generated in the second step are called father scenarios. In step 3, mutants of scenarios generated in the second step are generated by pre-defined mutant operators and are called son scenarios. In step 4, father and son scenarios are all executed, and SUT performance in these scenarios is recorded. In step 5, if the performance in a son scenario is quite different from the performance in the corresponding father scenario, the son scenario will be killed and, if not, saved. The mutant metric is the ratio of the number of killed and saved son scenarios. In step 6, if the pre-defined stop conditions are meted, iteration stops, and the son scenarios generated in the last iteration are regarded as the most diverse scenarios. If not, son scenarios become parent scenarios and iterate steps 3 to 6. Two high-performance AEB systems are tested in [155], resulting in 11,145 scenarios from 30 high-quality scenario sets, better than random sampling and experts. Since FIC-Sampling can be regarded as a variant of CT, the methodology described in [155] can be considered a combination of MT and CT. The performance of this methodology highly depends on the pre-designed scenario templates, which can not only effectively avoid generating unreasonable scenarios but also stop the generation of unknown but valuable scenarios simultaneously.

MT can also be used to obtain diverse path planners. For a path planner, the weight combinations in the cost function determine its style, which may consider several aspects, such as safety, comfort, energy efficiency, etc. Different weight combinations lead to different driving styles. In [156], MT mutates the weights of cost functions, which guide the path planners to find the optimal trajectories for AVs. In the experiment, 42 path planners with different weight combinations are generated, which can be used to obtain more diverse maneuvers of vehicles around the VUT. Inspired by the Genetic Algorithm (GA), a variant of MT, SceGene, is proposed by [111]. In SceGene, scenario features are encoded as genes on chromosomes. Iteration processes, such as crossover, mutation, and selection, are activated to generate new scenarios based on initial scenarios. A microscopic driving model is adopted to repair scenarios that are not valid. This methodology is exploited to generate 1000 diverse merging scenarios in simulation experiments. The authors of [111] claim that SceGene can automatically generate diverse and realistic traffic scenarios. However, the repair rate of the merging scenario is 34.4%, indicating the unstable performance of this method. The repair rate is calculated by (repaired number)/(total number).

Similar to CT, the boosting number of scenario parameters is catastrophic for MT because many iterations would be needed to achieve an excellent mutant score. The iterative processes in MT require many computation resources, and with no guiding processes for critical scenario generation, many of the resulting scenario are not critical for the VUT.

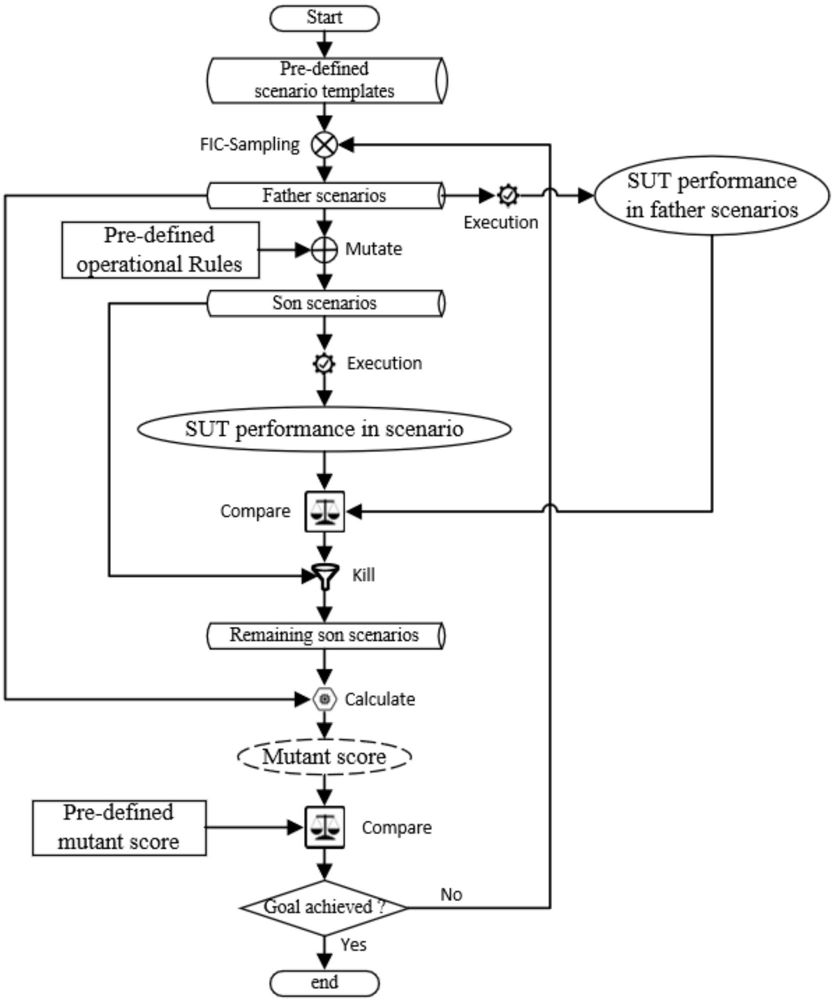

Figure 5. A flowchart of scenario generation based on MT [155]. "Father scenario" and "Son scenario" are terms used in this survey for convenience.

6.2. Critical Scenario Generation

6.2.1. Accelerated Evaluation

Considering most scenarios generated by random sampling are not safety critical, an original Accelerated Evaluation (AE) based on statistical sampling is proposed by [29]. In [29], unusual behavior models of the other primary vehicles are obtained by sampling from an accelerated distribution $f^*(x)$, which is obtained by skewing the original distribution $f(x)$ in NDD by Importance Sampling (IS). Simulation results show that the efficiency of AE is 300 to 100,000 times better than road testing in generating critical traffic scenarios. Therefore, AE can be divided into the following six steps.

1. Collect a large amount of NDD.
2. Identify target scenarios.
3. Fit the original Probability Density Function (PDF) $f(x)$ of each scenario parameter.
4. Skew the original PDF, deriving a modified PDF $f^*(x)$, which will lead to more radical behaviors of traffic agents or rare scenarios.
5. Random sampling is conducted based on the modified PDF $f^*(x)$ to generate accelerated scenarios, and then applied to test the VUT.
6. The accelerated scenarios are statistically skewed back to obtain the performance of the VUT in natural traffic.

In step 3, the original PDF $f(x)$ can be fitted by Piecewise mixture distribution models [157] or KDE [147,158] to reduce the gap between the real one and the statistical one. In step 4, Importance Sampling (IS) is innovatively used in [29] to make step 6 feasible. The hyperparameters needed for IS can be obtained based on random searching or other search algorithms, such as the cross-entropy method [159] and GA [160].

In [159], the lane-change scenario, as shown in Figure 6, is described by three parameters, $v(t_{LC}), v_L(t_{LC}),\ R_L(t_{LC})$. t_{LC} is the time when the lane marker is crossed by the center line of the Lane-Change Vehicle (LCV). v_L and v are the velocities of LCV and the VUT, respectively. R_L is the distance gap between the LCV and the LKV. Time to collision $\mathrm{TTC}_L = \frac{R_L}{\dot{R}_L}$ is used as the criticality metric of the lane-change scenario. Since $v = v_L - \dot{R}_L$, then a lane-change scenario can be described by a vector $x = \begin{bmatrix} v_L & \mathrm{TTC}_L^{-1} & R_L^{-1} \end{bmatrix}$. The distribution of v_L in NDD is adopted. The function of IS is to skew the Probability Density Function (PDF) of TTC_L^{-1} and R_L^{-1}, which are $f_{\mathrm{TTC}_L^{-1}}$ and $f_{R_L^{-1}}$, resulting in two modified PDFs of them $f^*_{\mathrm{TTC}_L^{-1}}$ and $f^*_{R_L^{-1}}$. The similarity between the original PDF f and its modified PDF f is measured by the likelihood in Equation (1). Then, aiming for the best L close to 1, The cross-entropy method is used to search for the appropriate hyperparameters of IS, $\vartheta_{\mathrm{TTC}_L^{-1}}$ and $\vartheta_{R_L^{-1}}$. Experiment results show that test efficiency in lane-change scenarios is improved by IS-based AE by 200 to 20,000 times compared with natural driving testing.

$$L\left(R_L^{-1} = x, \mathrm{TTC}_L^{-1} = y\right) = \frac{f_{\mathrm{TTC}_L^{-1}}(x) \cdot f_{R_L^{-1}}(y)}{f^*_{\mathrm{TTC}_L^{-1}} \cdot f^*_{R_L^{-1}}} \tag{1}$$

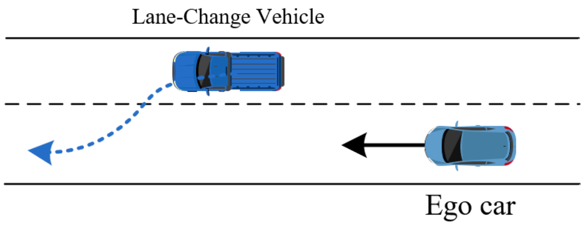

Figure 6. Lane-change scenario [159].

It should be noted that the core of AE is obtaining the accelerated $f^*(x)$, which means that all approaches that can obtain the accelerated $f^*(x)$ can be used for AE [148,161,162]. An AE strategy based on kernel methods is proposed by [148]. The function of kernel methods is mapping scenario parameters to a latent space. This strategy includes six steps. First, some initial scenarios are generated by random sampling and executed, resulting in the initial training set in the original space $\{(X_i, Y_i)\}_{i=1}^n$, where X is the scenario parameter set, and Y is a binary set that indicates if the responding scenarios are critical or not. Second, the training set is mapped by a kernel function $\phi(x)$ to a latent space, resulting scenario in the latent space, $\{(\phi(X_i), Y_i)\}_{i=1}^n$. Third, a Support Vector Machines (SVM) classifier is trained by latent scenarios to find the linear boundary between critical and non-critical scenarios in the latent space. Forth, another set of random scenarios $\left\{\left(\phi\left(X_j\right)\right)\right\}_{j=1}^{\mathrm{m}}$ are generated and mapped by $\phi(x)$. Fifth, GMM is used to approximate the PDF of $\phi(X_i)$, and leads to an approximation model $\widetilde{f}(\phi(X_i))$, from which the modified distribution of $\phi(X_i)$, $\widetilde{f}^*(\phi(X_i)$, is derived. Sixth, $f^*(X_i)$ is acquired by adjusting the marginal distribution of $\widetilde{f}^*(\phi(X_i)$ to make it closer to the area dominated by critical latent scenarios. The SVM classifier identifies these critical latent scenarios, and its accuracy needs to be improved by iterations. Simulation experiments about cut-in scenarios are carried out to validate and evaluate this strategy. Simulation results imply kernel model-based AE is 60,000 times better than random sampling. The biggest disadvantage of this method is that many iterations are needed to train the SVM classifier with random latent scenarios, which consumes a lot of computing resources. Therefore, this strategy can be improved further by detecting the performance boundary in the latent space based on fewer scenario executions.

Since the accelerated distribution function $f^*(x)$ is the unbiased estimation of the original estimation $f(x)$, the VUT performance in the generated scenarios can be skewed back to that in natural traffic. However, obtaining high-accuracy $f(x)$ usually requires lots of effort for data collection. Moreover, the generated scenarios based on AE are not always feasible, although feasibility analysis can eliminate this problem [127].

6.2.2. Search Algorithms

Finding the most critical scenarios is a Worst Case Searching (WCS) problem, which can also be regarded as an optimization problem. Many search algorithms have been used for WCS, including Genetic Algorithm (GA) [163], Particle Swarm Optimization (PSO) [164], multi-arm searching [165], Bayesian optimization algorithm [166], Rapidly-exploring random trees [167], Differential Evolution Algorithm (DEA) [168], and so on. A toolchain for critical scenario generation based on search algorithms is introduced in [44]. There are three keys for scenario generation based on search algorithms: define an appropriate fitness function, find a balance between exploration and exploitation, and reduce the search space as much as possible.

Fitness functions are crucial to searching algorithm-based WCS [168]. Based on pre-defined fitness functions, search algorithms can adaptively search for the worst case/scenario. Drivable area minimization is an efficient method to generate critical scenarios for motion planners. The drivable area indicates the solution space where the VUT operates appropriately without leading to any collision. Therefore, the drivable area is not relevant to the performance of the VUT. In [169], drivable areas are minimized by adjusting the initial state parameters of traffic participants using Evolutionary Algorithms (EA), such as DE and PSO. Simulation experiments involving multiple vehicles are carried out to generate critical highway and intersection scenarios. Experiment results show that by iteratively adjusting maneuvers of more than 10 traffic agents, both DE and PSO can find scenarios with little drivable space for the ego in both highway or intersection areas, and collisions are found in several highway scenarios. In contrast, no collision is produced in the intersection scenario, although several scenarios with small drivable areas are generated.

To our knowledge, no general metric can comprehensively quantify all properties of all scenarios. For example, Time-To-Collision (TTC) might be enough to measure the safety of car-following scenarios, but it is not suitable for intersection scenarios. A fitness function may consider several aspects by including several metrics. Combining metrics concerning different aspects of different aspects might be a good option to find scenarios with various characteristics (see Section 7).

How to obtain the best fitness function is still an open question [170]. Several templates for designing fitness functions are provided in [171]. However, given a suitable fitness function, it might still be challenging for search algorithms to find the most critical scenarios because of the high-dimension ODD and the increasingly complicated AVs. Therefore, a search strategy must balance exploration and exploitation, and the search space should be reduced as much as possible to find the best target more efficiently [163,172].

To enhance local searching ability, [163] combines a GA with a local fuzzer to generate the most critical trajectories of Non-Player Characters (NPCs) to minimize the safety of the SUT, and some faults of Baidu Apollo are found in a simulation experiment. A two-stage method is proposed in [173]. In the first stage, the optimization algorithm obtains diverse collision scenarios. In stage two, the purpose is to find the best parameter combination for path planners to avoid collision in the most critical scenarios found in stage one. A five-module method is proposed in [165]. These five modules include the Exploration and Exploitation Module based on the multi-arm bandit method, the Parameter Moving Probability Determination Module for ensuring the change of influencing factors, the Step Size Determination Module for searching size controlling, and the Memory Function Module for avoiding repeat searching. Simulation experiments are carried out to generate critical car-following scenarios. Experiment results show that 174 critical scenarios are generated based on the methodology proposed in [165] with 1251 scenarios executed.

Performance boundaries are the ones near which scenarios tend to lead to apparent changes in AV behaviors. Boundary scenarios near performance boundaries are valuable for AVs, and they can guide the generation of critical scenarios by reducing the search space [167,174–176]. In [174], decision tree classification and multi-objective population-based search algorithms are combined to find performance boundaries of critical regions. The critical scenarios found by the search algorithm can increase classification accuracy. Simultaneously, critical regions identified by the classifier can enhance the efficiency of critical scenario searching. An experiment testing an AEB system shows that 731 distinct, critical scenarios are found within 24 h.

To generate scenarios with realism, severity, and exposure considered, a heuristic algorithm is applied in [40]. First, scenarios in NDD are identified and classified into two sets, the critical and normal sets. After that, parameter distributions are fitted, and dependencies between parameters are obtained by regression analysis. Then, heuristic algorithms are applied to find the critical and high-coverage scenarios in the normal and high-coverage sets. The risk potential field and the diversity of parameter combinations are utilized to measure the criticality and exposure of scenarios, respectively. A new metric quantifying the difference between two scenarios is proposed to eliminate duplicate scenarios. This method is applied to an unpublished dataset, and 229 critical and 2065 high-coverage scenarios are found based on the critical and high-coverage datasets, respectively.

Although search algorithm-based approaches have been proven effective in some studies [111,167,174–176], numerous iterations are unavoidable and consume a lot of computation resources, especially for high-dimensional critical scenario generation.

6.2.3. Reinforcement Learning

Similar to the search algorithm-based method, guided by a pre-defined reward function, Reinforcement Learning (RL) searches for critical scenarios adaptively without prior knowledge about the SUT [11], as shown in Figure 7. A DDSG methodology based on Neural Architecture Search is proposed in [177]. A policy-gradient RL algorithm is applied in Neural Architecture Search. The longitudinal safe distance based on Responsibility Sensitive Safety (RSS) method is used to quantify the criticality of the situation in which ego should be responsible for the possible accident. A reward function is specifically designed, which gives a bonus to parameter combinations leading to a crash.

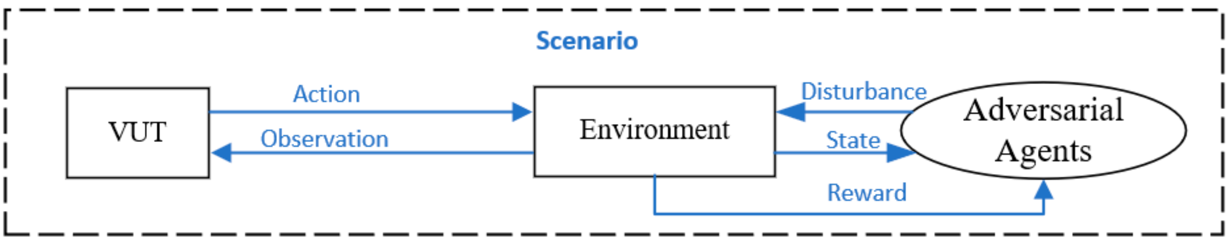

Figure 7. A flowchart of critical scenario generation based on RL [11].

Q-learning is applied in [178] to search for critical scenarios. The reward function is a criticality metric that considers distance headway, Time-To-Collision (TTC), and the longitudinal acceleration required for collision avoidance. The numerical experiment shows that the proportion of critical scenarios is 37.13% more than that of randomly generated scenarios. RL-based on Long and Short-Term Memory (LSTM) architecture is applied in [179] to find the worst perception scenario. This method is evaluated by several experiments utilizing three datasets, EVB, KITTI, and BDD100K.

Adaptive Stress Testing (AST) based on deep RL is proposed by [180] to test decision-making systems. AST searches for the worst scenarios by adaptively changing the behaviors of the agents around the VUT to maximize the pre-defined reward. Simulation results indicate that deep RL is more efficient than a Monte Carlo Tree Search (MCTS) algorithm. However, iterations involved in AST require many high-fidelity simulations, which is costly and time-consuming. A methodology combining AST and the backward searching algorithm is proposed to tackle this problem [181]. First, critical scenarios in the low-fidelity simulation environment are generated by AST. Then these critical scenarios are utilized as expert demonstrations of the backward searching algorithm to find critical scenarios in the high-fidelity simulation environment. Simulation results show that the percentage of high-fidelity simulations is reduced to 4.7% without losing many rewards.

Like search algorithm-based methods, RL-based methods also belong to the falsification approach. If the VUT fails in a scenario, a conclusion that the VUT is not safe enough can be made. However, if no critical scenarios are found, it is not reasonable to claim that the VUT is safe. Furthermore, RL-based methods usually need many high-fidelity simulations, which needs sufficient computing efforts.

6.2.4. Others

A model-driven adversarial testing strategy is presented by [182], in which adversarial trajectories for NPCs are continuously generated based on a pre-defined anchor-template hierarchy structure, and a lower-level controller is designed to track these trajectories. Unlike other iteration-based methods, the VUT is tested continuously in one scenario. However, the duration of searching critical scenarios based on this method would cost much more time than one iteration of other methods, such as RL-based testing. Experiments involving highway scenarios, including several traffic agents and one VUT, are carried out to prove the effectiveness of this strategy. The simulation results show that the strategy proposed by [182] can find critical scenarios within 15 seconds. But no baseline is provided to measure the efficiency of this strategy.

6.3. Customized Scenario Generation

If the VUT is unavailable, a surrogate model of the VUT is usually utilized to generate critical scenarios. Five SMs are utilized and compared in [183], including RBF, Kriging, QP, IDW, XGB, and SVR. In simulation experiments, a modified IDM is used as the VUT. Simulation results indicate that different SMs lead to scenarios with different characteristics, and IDW-based SM is the best. It generates a large percentage of critical scenarios while exploiting only 2.5% of the test resource used by random searching. A multi-start optimization approach assisted by a seed-filling technique is described in [184] and applied in [185] for critical scenario searching based on a GPC-based SM. First, a lot of random scenarios are generated and simulated. Then these initial scenarios are labeled with critical or non-critical tags. Then they are used as the training data set for GPC. Simulation experiments of the car-following scenario are presented to evaluate this method, which is described by three parameters, including the speed of the ego car, the speed of the leading car, and the aperture angle of the radar sensor of the ego car. Simulation results show that the boundary between critical and non-critical scenarios is successfully found. However, it will need a large number of random samples for high-dimension scenarios. Therefore, it is necessary to find a method that can derive a high-performance SM without consuming too many computational resources.

There are some studies proposing methodologies for SM optimization. Most of them follow the process depicted in Figure 8. The method mentioned in [184] is further improved by an adaptively enhanced GPR-based surrogate model [166]. Ref. [166] presents an Adaptive Testing Scenario Library Generation (ATSLG) methodology using a Bayesian optimization scheme. Gaussian Process (GP) plays a crucial part in this methodology. There are two roles for GP in [166]. Gaussian Process Regression (GPR) estimates the dissimilarity between the surrogate model and the VUT. Gaussian Process Classification (GPC) estimates observations of unexecuted scenarios. GPR is applied to classify the parameter space into two sets, suboptimal scenarios and optima scenarios. The unexecuted scenarios whose observation GPR models cannot estimate accurately with high possibility are called informative scenarios. In each iteration, informative scenarios are selected and executed to provide information to improve GPR models and GPC models. This method is applied to enhance the AE method in [185] by adaptively find the most informative scenarios based on an optimized SM. Simulation results show that 18 times more critical cut-in scenarios are generated. Similar techniques can also be found in [183,186,187].

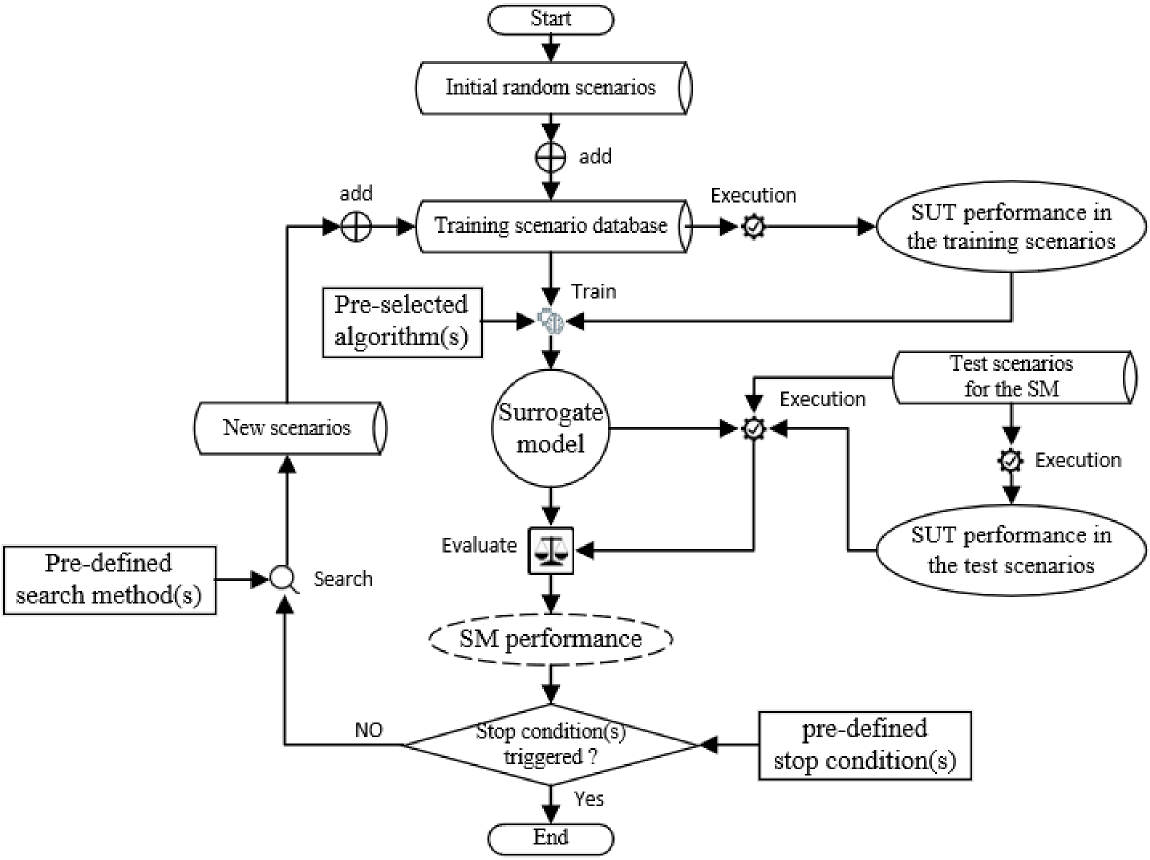

Figure 8. A schematic diagram for SM optimization.

In [188], a surrogate model based on Artificial Neural Networks (ANN) is iteratively improved by iteratively generating diverse samples. Experiments of generating car-following scenarios involving three vehicles are described, and TTC is adopted as the criticality metric. Simulation results show that after 2560 scenario executions, the MSE between the results estimated by the ANN and the real ones is 0.0097.

ANN-based SM can better model high-dimension logical scenarios when enough training data are provided. The GP-based SM usually performs better when the training set size is limited. As illustrated above, some studies claim that their methods can generate customized scenarios effectively and reduce testing efforts significantly. However, no cases involving complicated AV systems are described or applied in the industry.

6.4. Conclusions of Scenario Generation

The tasks of this section are to introduce methodologies for generating diverse, critical, and customized scenarios. A qualitative comparison of the strategies mentioned above is presented in Table 3. The section on customized scenario generation is mainly about obtaining a good SM to assist in the generation of high-quality scenarios. Therefore, the strategies for customized scenario generation are not included in Table 3.

Table 3. Qualitative comparison of scenario generation strategies. "Diversity": Capable of generating diverse scenarios that can fill the whole ODD. "Criticality": Capable of finding a large percentage of critical scenarios. "Knowledge": a certain amount of expert knowledge is needed. "Iteration": Some iterations are required. "Naturality": testing results can be mapped to that in natural traffic. "Scalability": Capable of generating high-dimensional critical scenarios without consuming many resources. "Y": Yes. "N": No. "Y/N": Yes, for some cases, and NO, for others. For example, random sampling can generate natural traffic scenarios if sampling on the parameter distributions in natural traffic, and vice versa.

Type	Method	Diversity	Criticality	Knowledge	Iteration	Naturality	Scalability
Diversity-Oriented	Random Sampling	Y	N	N	N	Y/N	Y
	CT	Y	Y	N	N	N	Y
	MT	Y	N	Y	Y	N	N
Criticality-Oriented	AE	N	Y	Y/N	Y/N	Y	Y
	Search Algorithms	N	Y	Y/N	Y	N	N
	RL	N	Y	Y/N	Y	N	N

As for generating diverse scenarios, random sampling can achieve the best diversity, and testing results can be mapped to natural traffic. But scenarios generated based on NDD have only a small percentage of critical scenarios. CT discretizes continuous scenario parameters and generates scenarios by combining different parameters' discrete values. A greater granularity of parameter discretization leads to more scenarios requiring more computing resources. MT generates scenarios relying on pre-defined mutation rules, which experts must design. Moreover, the resulting scenarios, including non-critical ones, would be generated and executed for AV testing, leading to unnecessary wasting of efforts. The biggest drawback of diversity-oriented methods is that many boring scenarios might be generated and executed, significantly reducing test efficiency. It is a promising research topic to reduce the execution of boring scenarios without scarifying coverage.

AE, search algorithm-based, and RL-based methodologies aim to generate critical scenarios. Unlike the other two methodologies, AE can achieve naturality and criticality at the same time [147]. However, expert knowledge might be crucial to tune vital parameters for some statistical sampling techniques, such as IS. Since the original parameter PDF is essential to AE-based methods, the completeness of source data is essential to AE. Search algorithms have been proven effective in some cases, but the involved iteration processes often require much computation, especially for high-dimensional critical scenario generation. Although RL has been proven more efficient than search algorithms in generating specific scenarios [180], both need expert knowledge to design a fitness or reward function. It is a significant challenge and worth more attention to find the most critical scenario for the VUT based on limited sources.

Customized scenarios could be generated based on high-performance SMs to minimize unnecessary execution. While many surrogate models can be chosen, such as RBF, Kriging, QP, IDW, XGB, and SVR [183], each is suitable for certain circumstances. It might be good to build several surrogate models and choose the best one for the SUT based on their functional performance.

7. Criticality Metrics of Scenarios

Criticality metrics of scenarios are of great importance in DDSG. On the one hand, criticality metrics can be used to identify or generate scenarios for DDSG. On the other hand, the safety of the SUT can also be measured by criticality metrics. Criticality metrics used in DDSG are listed and compared in [43,189]. This survey categorizes criticality metrics into five classes: trajectory-based, maneuver-based, energy-based, uncertainty-based, and combination-based. They are described in the following sections.

7.1. Trajectory-Based

Trajectory-based criticality metrics are the ones that can be calculated based on the whole or part of the trajectories of traffic participants. Time headway (THW) [190], Gap Time (GT), and Distance Head Way (DHW) quantify the spatial or temporal gap between two traffic participants without considering their velocities. Time-to-Collision (TTC) is the time to collide if two participants keep their velocities unchanged. TTC is almost the most widely used criticality metric. However, there are many scenarios in which the velocities of traffic participants change, and more than two participants exist. Therefore, some TTC variants are designed to measure the safety of the whole scenario, such as Worst Time To Collision (WTTC) [191], Time To Closest Encounter (TTCE) [192], Time Exposed TTC (TET) [193], minimal normalized positive enhanced time-to-collision (mnpETTC) [184], Time Integrated TTC (TIT), Time to Zebra (TTZ) [194] et al.

Post Encroachment Time (PET) [195] is the time gap between one traffic agent leave and another one enters a conflict area, as shown in Figure 9 and Equation (2). PET is more suitable than TTC for the criticality quantification of intersection scenarios.

$$\mathrm{PET}(Veh_1, Veh_2, \mathrm{CA}) = t_{\mathrm{enter}}(Veh_1, CA) - t_{\mathrm{leave}}(Veh_2, CA) \tag{2}$$

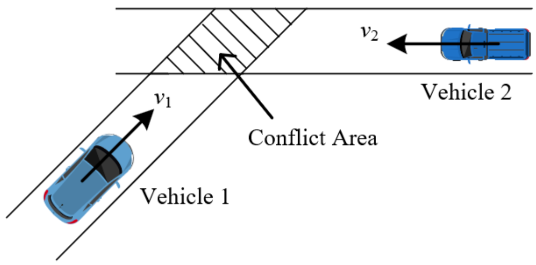

Figure 9. A schematic diagram of a conflict area [195].

However, TTC and its variants are available only after scenario execution. For scenarios unexecuted, it is impossible to be measured by trajectory-based metrics.

7.2. Maneuver-Based

Maneuver-based criticality metrics (MBSMs) are the ones that are relative to collision-avoiding maneuvers, such as braking and steering. Criticality metrics relative to braking behaviors include Time To Brake (TTB), Deceleration to Safety Time (DST), Brake Threat Number (BTN), Required Longitudinal Acceleration, Longitudinal Jerk, and so on [190,196]. Criticality metrics involving steering maneuvers include Time To Steer (TTS), Steer Threat Number (STN), Required Lateral Acceleration (RLA), Required Longitudinal Acceleration (RLA), Required Lateral Acceleration, Lateral Jerk, et al. In some MBSMs, all kinds of collision-avoiding maneuvers are considered, including Time to Maneuver (TTM), Required Acceleration (RA), and Time to Reaction (TTR).

Based on trajectories generated by a path planner, some information about scenarios can be derived. Several criticality metrics based on trajectories are designed by [197], including SafePathInv, UnsafePercent, NarrowInv, AvgEffort, MinEffort, NarrowInv, and CriticalTime. SafePathInv is the inverse of the number of safe paths available to the VUT. UnsafePercent is the percentage of paths that will lead to a collision among all safe paths for the VUT. AvgEffort is the average effort the VUT needs to pay to pass the scenario safely. All safe paths are analyzed to obtain the absolute values of steering and acceleration to follow each safe path. MinEffort is the minimum effort required to follow a safe path. Critical Time is the minimal time available before the VUT is unable to avoid a crash. Numerical experiments indicate that most difficult scenarios in the NGSIM dataset have a score of 3.22, while a rule-based attacker designed by [197] can generate adversarial scenarios with a score of 3.95. However, this method heavily relies on the performance of the path planner. Different path planners may lead to different results.

Maneuver-based metrics do not rely on the performance of the VUT, which makes them suitable for measuring scenario criticality before executions. However, some maneuvers-based metrics, such as STN, BTN, SafePathInv, cannot separate similar scenarios with different criticality.

7.3. Energy-Based

If a collision is not avoidable, it is necessary to reduce the accident's severity. The kinematic energy released in a collision is directly relative to crash severity [198]. Therefore, the kinematic energy of traffic participants can be used as a criticality metric. In [31], the kinematic energy of the ego at the moment of avoiding the collision is regarded as part of the Scenario Risk Index (SRI).

The kinematic energy of a traffic agent is relevant to its weight, which is not always available. Therefore, energy-based metrics often work as a complement to other criticality metrics.

7.4. Uncertainty-Based

Many factors may contribute to the crash in a scenario, such as road friction and velocity variance of vehicles. Therefore, possibility-based metrics are proposed to capture these uncertainties. The temporal variation of estimated collision speed between a vehicle and a pedestrian in a crosswalk scenario is quantified by Pedestrian Risk Index (PRI) in [199]. Crash Potential Index (CPI) [200] is the average crash possibility if the required deceleration exceeds the maximal available deceleration in the scenario. The maximal available deceleration in the scenario is described as a distribution relevant to objective factors such as the road material and performance of the braking system. Parameters in CPI are obtained by calibration based on traffic data in an intersection in NGSIM dataset in [200]. No application case of this metric is described in [200].

In natural traffic, a minor behavior change of the traffic participants around the ego might lead to a fetal accident [201]. Ref. [202] insists that a scenario is critical if the VUT cannot predict the trajectory of at least one participant. In [203], all possible trajectories of NPCs are predicted, and Monte-Carlo simulations are carried out to estimate Time-To-Critical-Collision-Probability (TTCCP) to consider the uncertainty of their behaviors.

7.5. Combination-Based

Different metrics with different properties can measure scenarios from different perspectives. Five metrics are exploited in [170] to guide evolutionary algorithms to find scenarios with different properties, including criticality. Furthermore, as a general metric that can measure the safety of all scenarios does not exist now, combining different metrics is a good option.

On the one hand, several criticality metrics can be adaptively used to measure scenario criticality. An adaptive methodology to measure scenario safety is described in [204]. Several criticality metrics concerning different aspects are selected based on multidimensional criticality analysis. A situation awareness module is designed to identify the type of the current scenario. Applicable metrics in the current scenario are calculated, and those exceeding pre-defined thresholds are weighted and summed together, resulting in the safety of the current scenario. In [178], a combination-based safety, including longitudinal acceleration, time headway, and TTC, is applied as a reward function to guide critical scenario generation based on RL.

On the other hand, metrics concerning several other aspects except safety may be integrated. The production of exposure, severity, and controllability is used to quantify the risk of a scenario in [158,205]. Exposure is the expected happening possibility of a scenario. Severity is the expected possibility of collision if no backup operators are available. Controllability is the ratio of expected collision possibility with and without backup operators.

7.6. Conclusions of Criticality Metrics of Scenarios

Five classes of criticality metrics are considered in this section. The trajectory-based criticality metrics can be calculated given all trajectories of traffic participants in a scenario or all positions in a scene. Maneuver-based ones are proposed to measure the difficulty of avoiding an accident. Energy-based ones are applied to measure the severity of a crash. Uncertainty is the key to uncertainty-based criticality metrics. More uncertainty in a scenario would lead to more challenges for the SUT. Unlike other criticality metrics, a combination-based critical metric integrates several metrics concerning different aspects, resulting in a more comprehensive metric. However, to our knowledge, no single criticality metric can be utilized for all scenarios. Therefore, researchers are advised to design or adopt appropriate criticality metrics for different scenarios. A general and objective criticality metric for all scenarios does not exist by far.

8. Discussions and Conclusions

This work decomposes methodologies described in relevant studies into solutions to several fundamental problems about DDSG, including source data collection, scenario identification, scenario generation, and criticality metrics of scenarios. Involved techniques for similar problems are analyzed and compared with each other. Conclusions about each methodology mentioned in this survey can be found at the end of the related section. To avoid unnecessary repetition, they are not included in this section. Some hot research topics are summarized as follows:

1. Develop a toolchain that can generate good-quality traffic-scenario data on simulation platforms to reduce the efforts and investments for gathering source data in the real world.
2. Build a methodology that can effectively and efficiently identify known and unknown scenarios in source data without sacrificing feasibility to use all source data fully.
3. Use different methodologies to generate diverse, critical, and natural scenarios to meet different requirements in different development stages.
4. Find a strategy to obtain high-performance Surrogate Models (SM) based on limited resources.
5. Design a general criticality metric that can objectively quantify the criticality of all scenarios.

There are also some significant problems in AV testing, rather than only existing in DDSG, and the authors of this survey think it is necessary to point out. Here are some of the most crucial ones:

1. Simulation fidelity and computing power need improvement. Simulation with high fidelity is crucial to executing scenarios and can contribute to source data generation. Simulation technology has been widely utilized in SBT. However, on the one hand, many studies utilize simulation techniques to execute scenarios. On the other hand, no software companies claim that their software can replace experiments in the real world. If it is impossible to replace the real world with a virtual one, it will be helpful to quantify the gap between them, which can let us know how much we can trust the simulation results. Moreover, it is of great significance to use low-fidelity simulations to reduce high-fidelity simulations, which are more expensive and consume more time.
2. There are no conclusions on how many of what scenarios are enough for AV testing. There is an infinite number of scenarios in the physical world. It is impossible to test AVs in all of them. An embarrassing dilemma is that many studies propose many scenario-based methods to test AVs, and no one concludes how many of what scenarios are enough for AV testing. It is significantly vital to draw a terminal line for this endless Marathon.
3. Data sharing is crucial for AV testing. Safety-critical events hidden in NDD are crucial for DDSG. However, because of the Curse of Rarity (CoR) [206], a large amount of

NDD would significantly contribute to DDSG. While some open datasets are available for researchers, giant companies like Tesla, Baidu, Didi, et al. hold a large amount of NDD privately. Furthermore, because many functions are complete black boxes, it is hard for a researcher or an engineer to generate customized scenarios for the VUT. It is reasonable to believe that more comprehensive cooperation between industry and academia can tremendously enhance the development of SBT of AVs.

4. The unignorable gap between ideology and reality deserves more attention: While one of the aims of developing AVs is to reduce traffic accidents to zero, achieving zero accidents in practice is severely challenging. It might be good to mitigate the public expectation to an appropriate level to let more un-perfect but good AVs be tested in natural traffic. This way, AVs will evolve and collect more valuable data for researchers.

Author Contributions: Conceptualization, J.C., Y.W. and W.D.; methodology, J.C. and Y.W.; formal analysis, J.C. and H.G.; investigation, J.C., H.G. and J.L.; writing—original draft preparation, J.C., H.G., Y.W., J.L. and J.D.; writing—review and editing, J.C., Y.W. and W.D.; supervision, Y.W.; project administration, Y.W.; funding acquisition, W.D. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Jilin Province Science and Technology Project, grant number 20200501012GX, and National Key Research and Development Program (2016YFB0100904).

Data Availability Statement: Not applicable.

Acknowledgments: All authors acknowledge the funding from Jilin Province Science and Technology Project, National Key Research and Development Program. All authors acknowledge and appreciate all of the contributions of related anonymous reviewers to the improvement of this article.

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

References

1. Ju, Z.; Zhang, H.; Li, X.; Chen, X.; Han, J.; Yang, M. A Survey on Attack Detection and Resilience for Connected and Automated Vehicles: From Vehicle Dynamics and Control Perspective. *IEEE Trans. Intell. Veh.* **2022**, 1–24. [CrossRef]
2. Paradina, R.M.; Noroña, M.I. Applications and Challenges of Adopting Internet of Things (IoT) in Reducing Road Traffic Accidents. In Proceedings of the Second Asia Pacific International Conference on Industrial Engineering and Operations Management, Surakarta, Indonesia, 14–16 September 2021.
3. Ghorpade, P.; Nema, A. A Review on Accident Detection System Using Iot. *Int. Res. J. Mod. Eng. Technol. Sci.* **2022**, *4*, 4278–4283.
4. Sohrabi, S.; Khodadadi, A.; Mousavi, S.M.; Dadashova, B.; Lord, D. Quantifying the Automated Vehicle Safety Performance: A Scoping Review of the Literature, Evaluation of Methods, and Directions for Future Research. *Accid. Anal. Prev.* **2021**, *152*, 106003. [CrossRef] [PubMed]
5. Shadrin, S.S.; Ivanova, A.A. Analytical Review of Standard Sae J3016 taxonomy and Definitions for Terms Related to Driving Automation Systems for On-Road Motor Vehicles with Latest Updates. *Avtomob. Doroga Infrastrukt.* **2019**, *3*, 10.
6. Thorn, E.; Kimmel, S.C.; Chaka, M.; Hamilton, B.A. *A Framework for Automated Driving System Testable Cases and Scenarios*; Department of Transportation, National Highway Traffic Safety: Washington, DC, USA, 2018.
7. Roesener, C.; Fahrenkrog, F.; Uhlig, A.; Eckstein, L. A Scenario-Based Assessment Approach for Automated Driving by Using Time Series Classification of Human-Driving Behaviour. In Proceedings of the 2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC), Rio de Janeiro, Brazil, 1–4 November 2016; pp. 1360–1365.
8. Jesenski, S.; Rothert, J.; Tiemann, N.; Zöllner, J.M. Using Sum-Product Networks for the Generation of Vehicle Populations On Highway Sections. In Proceedings of the 2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC), Rhodes, Greece, 20–23 September 2020; pp. 1–7.
9. Fremont, D.J.; Sangiovanni-Vincentelli, A.L.; Seshia, S.A. Safety in Autonomous Driving: Can Tools Offer Guarantees? In Proceedings of the 2021 58th ACM/IEEE Design Automation Conference (DAC), San Francisco, CA, USA, 5–9 December 2021; pp. 1311–1314.
10. Kiran, B.R.; Sobh, I.; Talpaert, V.; Mannion, P.; al Sallab, A.A.; Yogamani, S.; Pérez, P. Deep Reinforcement Learning for Autonomous Driving: A Survey. *IEEE Trans. Intell. Transp. Syst.* **2021**, *23*, 4909–4926. [CrossRef]
11. Karunakaran, D.; Worrall, S.; Nebot, E. Efficient Statistical Validation with Edge Cases to Evaluate Highly Automated Vehicles. In Proceedings of the 2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC), Rhodes, Greece, 20–23 September 2020; pp. 1–8.

12. Ding, W.; Xu, C.; Lin, H.; Li, B.; Zhao, D. A Survey on Safety-Critical Scenario Generation from Methodological Perspective. *arXiv* **2022**, arXiv:2202.02215.
13. Kalra, N.; Paddock, S.M. Driving to Safety: How Many Miles of Driving Would It Take to Demonstrate Autonomous Vehicle Reliability? *Transp. Res. A Policy Pract.* **2016**, *94*, 182–193. [CrossRef]
14. Winner, H.; Lemmer, K.; Form, T.; Mazzega, J. PEGASUS—First Steps for the Safe Introduction of Automated Driving. In *Road Vehicle Automation 5*; Springer: Berlin/Heidelberg, Germany, 2019; pp. 185–195.
15. Roesener, C.; Sauerbier, J.; Zlocki, A.; Fahrenkrog, F.; Wang, L.; Várhelyi, A.; de Gelder, E.; Dufils, J.; Breunig, S.; Mejuto, P. A Comprehensive Evaluation Approach for Highly Automated Driving. In Proceedings of the 25th International Technical Conference on the Enhanced Safety of Vehicles (ESV) National Highway Traffic Safety Administration, Detroit, MI, USA, 5–8 June 2017.
16. Fahrenkrog, F.; Rösener, C.; Zlocki, A.; Eckstein, L. Technical Evaluation and Impact Assessment of Automated Driving. In *Road Vehicle Automation 3*; Springer: Berlin/Heidelberg, Germany, 2016; pp. 237–246.
17. Makano, H. C-ITS and Connected Automated Driving in Japan. *Routes/Roads* **2017**. Available online: https://trid.trb.org/view/1487881 (accessed on 1 October 2022).
18. Elrofai, H.; Paardekooper, J.-P.; de Gelder, E.; Kalisvaart, S.; den Camp, O.O. Scenario-Based Safety Validation of Connected and Automated Driving. In *Technical Report*; Netherlands Organization for Applied Scientifific Research: Hague, The Newtherlands, 2018. Available online: http://resolver.tudelft.nl/uuid:2b155e03-5c51-4c9f-8908-3fa4c34b3897 (accessed on 1 October 2022).
19. McMinn, P. Search-Based Software Testing: Past, Present and Future. In Proceedings of the 2011 IEEE Fourth International Conference on Software Testing, Verification and Validation Workshops, Berlin, Germany, 21–25 March 2011; pp. 153–163.
20. Menzel, T.; Bagschik, G.; Maurer, M. Scenarios for Development, Test and Validation of Automated Vehicles. In Proceedings of the 2018 IEEE Intelligent Vehicles Symposium (IV), Changshu, China, 26–30 June 2018; pp. 1821–1827.
21. Li, Y.; Tao, J.; Wotawa, F. Ontology-Based Test Generation for Automated and Autonomous Driving Functions. *Inf. Softw. Technol.* **2020**, *117*, 106200. [CrossRef]
22. Bagschik, G.; Menzel, T.; Maurer, M. Ontology Based Scene Creation for the Development of Automated Vehicles. In Proceedings of the 2018 IEEE Intelligent Vehicles Symposium (IV), Changshu, China, 26–30 June 2018; pp. 1813–1820.
23. Tenbrock, A.; König, A.; Keutgens, T.; Weber, H. The ConScenD Dataset: Concrete Scenarios from the HighD Dataset According to ALKS Regulation UNECE R157 in OpenX. In Proceedings of the 2021 IEEE Intelligent Vehicles Symposium Workshops (IV Workshops), Nagoya, Japan, 11–17 July 2021; pp. 174–181.
24. Kramer, B.; Neurohr, C.; Büker, M.; Böde, E.; Fränzle, M.; Damm, W. Identification and Quantification of Hazardous Scenarios for Automated Driving. In Proceedings of the International Symposium on Model-Based Safety and Assessment, Lisbon, Portugal, 14–16 September 2020; Springer: Berlin/Heidelberg, Germany, 2020; pp. 163–178.
25. Elspas, P.; Langner, J.; Aydinbas, M.; Bach, J.; Sax, E. Leveraging Regular Expressions for Flexible Scenario Detection in Recorded Driving Data. In Proceedings of the 2020 IEEE International Symposium on Systems Engineering (ISSE), Vienna, Austria, 12 October–12 November 2020; pp. 1–8.
26. Riedmaier, S.; Ponn, T.; Ludwig, D.; Schick, B.; Diermeyer, F. Survey on Scenario-Based Safety Assessment of Automated Vehicles. *IEEE Access* **2020**, *8*, 87456–87477. [CrossRef]
27. Sun, J.; Zhang, H.; Zhou, H.; Yu, R.; Tian, Y. Scenario-Based Test Automation for Highly Automated Vehicles: A Review and Paving the Way for Systematic Safety Assurance. *IEEE Trans. Intell. Transp. Syst.* **2021**, *23*, 14088–14103. [CrossRef]
28. Batsch, F.; Kanarachos, S.; Cheah, M.; Ponticelli, R.; Blundell, M. A Taxonomy of Validation Strategies to Ensure the Safe Operation of Highly Automated Vehicles. *J. Intell. Transp. Syst.* **2021**, *26*, 14–33. [CrossRef]
29. Zhao, D. Accelerated Evaluation of Automated Vehicles. Ph.D. Thesis, The University of Michigan, Ann Arbor, MI, USA, 2016.
30. Brown, B.; Park, D.; Sheehan, B.; Shikoff, S.; Solomon, J.; Yang, J.; Kim, I. Assessment of Human Driver Safety at Dilemma Zones with Automated Vehicles through a Virtual Reality Environment. In Proceedings of the 2018 Systems and Information Engineering Design Symposium (SIEDS), Charlottesville, VA, USA, 27 April 2018; pp. 185–190.
31. Yue, B.; Shi, S.; Wang, S.; Lin, N. Low-Cost Urban Test Scenario Generation Using Microscopic Traffic Simulation. *IEEE Access* **2020**, *8*, 123398–123407. [CrossRef]
32. Scanlon, J.M.; Kusano, K.D.; Daniel, T.; Alderson, C.; Ogle, A.; Victor, T. Waymo Simulated Driving Behavior in Reconstructed Fatal Crashes within an Autonomous Vehicle Operating Domain. *Accid. Anal. Prev.* **2021**, *163*, 106454. [CrossRef]
33. Damm, W.; Möhlmann, E.; Rakow, A. Traffic Sequence Charts for the Enable-s 3 Test Architecture. In *Validation and Verification of Automated Systems*; Springer: Berlin/Heidelberg, Germany, 2020; pp. 45–60.
34. De Gelder, E.; Paardekooper, J.-P.; Saberi, A.K.; Elrofai, H.; op den Camp, O.; Kraines, S.; Ploeg, J.; de Schutter, B. Towards an Ontology for Scenario Definition for the Assessment of Automated Vehicles: An Object-Oriented Framework. *IEEE Trans. Intell. Veh.* **2022**, *7*, 300–314. [CrossRef]
35. Ulbrich, S.; Menzel, T.; Reschka, A.; Schuldt, F.; Maurer, M. Defining and Substantiating the Terms Scene, Situation, and Scenario for Automated Driving. In Proceedings of the 2015 IEEE 18th International Conference on Intelligent Transportation Systems, Gran Canaria, Spain, 15–18 September 2015; pp. 982–988.
36. Steimle, M.; Menzel, T.; Maurer, M. A Method for Classifying Test Bench Configurations in a Scenario-Based Test Approach for Automated Vehicles. *arXiv* **2019**, arXiv:1905.09018.

37. Sauerbier, J.; Bock, J.; Weber, H.; Eckstein, L. Definition of Scenarios for Safety Validation of Automated Driving Functions. *ATZ Worldwide* **2019**, *121*, 42–45. [CrossRef]
38. Scholtes, M.; Westhofen, L.; Turner, L.R.; Lotto, K.; Schuldes, M.; Weber, H.; Wagener, N.; Neurohr, C.; Bollmann, M.H.; Körtke, F. 6-Layer Model for a Structured Description and Categorization of Urban Traffic and Environment. *IEEE Access* **2021**, *9*, 59131–59147. [CrossRef]
39. Damm, W.; Möhlmann, E.; Rakow, A. A Scenario Discovery Process Based on Traffic Sequence Charts. In *Validation and Verification of Automated Systems*; Springer: Berlin/Heidelberg, Germany, 2020; pp. 61–73.
40. Stepien, L.; Thal, S.; Henze, R.; Nakamura, H.; Antona-Makoshi, J.; Uchida, N.; Raksincharoensak, P. Applying Heuristics to Generate Test Cases for Automated Driving Safety Evaluation. *Appl. Sci.* **2021**, *11*, 10166. [CrossRef]
41. Zhong, Z.; Tang, Y.; Zhou, Y.; Neves, V.d.O.; Liu, Y.; Ray, B. A Survey on Scenario-Based Testing for Automated Driving Systems in High-Fidelity Simulation. *arXiv* **2021**, arXiv:2112.00964.
42. Gettman, D.; Head, L. Surrogate Safety Measures from Traffic Simulation Models. *Transp. Res. Rec.* **2003**, *1840*, 104–115. [CrossRef]
43. Junietz, P.M. Microscopic and Macroscopic Risk Metrics for the Safety Validation of Automated Driving. Ph.D. Theis, Technischen Universität Darmstadt, Darmstadt, Germany, 2019.
44. Song, Q.; Tan, K.; Runeson, P.; Persson, S. Critical Scenario Identification for Realistic Testing of Autonomous Driving Systems. 2022. Available online: https://assets.researchsquare.com/files/rs-1280095/v1_covered.pdf?c=1642707391 (accessed on 1 October 2022).
45. Ponn, T.; Breitfuß, M.; Yu, X.; Diermeyer, F. Identification of Challenging Highway-Scenarios for the Safety Validation of Automated Vehicles Based on Real Driving Data. In Proceedings of the 2020 Fifteenth International Conference on Ecological Vehicles and Renewable Energies (EVER), Monte-Carlo, Monaco, 10–12 September 2020; pp. 1–10.
46. Ponn, T.; Gnandt, C.; Diermeyer, F. An Optimization-Based Method to Identify Relevant Scenarios for Type Approval of Automated Vehicles. In Proceedings of the ESV—International Technical Conference on the Enhanced Safety of Vehicles, Eindhoven, The Netherlands, 10–13 June 2019; pp. 10–13.
47. Ponn, T.; Lanz, T.; Diermeyer, F. Automatic Generation of Road Geometries to Create Challenging Scenarios for Automated Vehicles Based on the Sensor Setup. In Proceedings of the 2020 IEEE Intelligent Vehicles Symposium (IV), Las Vegas, NV, USA, 19 October–13 November 2020; pp. 694–700.
48. Wang, X.; Zhang, S.; Peng, H. Comprehensive Safety Evaluation of Highly Automated Vehicles at the Roundabout Scenario. *IEEE Trans. Intell. Transp. Syst.* **2022**, 1–16. [CrossRef]
49. Zhang, X.; Tao, J.; Tan, K.; Törngren, M.; Sánchez, J.M.G.; Ramli, M.R.; Tao, X.; Gyllenhammar, M.; Wotawa, F.; Mohan, N. Finding Critical Scenarios for Automated Driving Systems: A Systematic Literature Review. *arXiv* **2021**, arXiv:2110.08664.
50. Steimle, M.; Menzel, T.; Maurer, M. Toward a Consistent Taxonomy for Scenario-Based Development and Test Approaches for Automated Vehicles: A Proposal for a Structuring Framework, a Basic Vocabulary, and Its Application. *IEEE Access* **2021**, *9*, 147828–147854. [CrossRef]
51. Badue, C.; Guidolini, R.; Carneiro, R.V.; Azevedo, P.; Cardoso, V.B.; Forechi, A.; Jesus, L.; Berriel, R.; Paixao, T.M.; Mutz, F. Self-Driving Cars: A Survey. *Expert Syst. Appl.* **2021**, *165*, 113816. [CrossRef]
52. Wu, Z.; Zhang, D.; Wang, Y.; Zhou, H.; Qi, H. Lane-Changing Behavior Recognition in the Connected Vehicle Environment. In Proceedings of the International Conference on Intelligent Transportation Engineering, Beijing, China, 12–13 November 2022; Springer: Berlin/Heidelberg, Germany, 2022; pp. 50–61.
53. Montanari, F.; German, R.; Djanatliev, A. Pattern Recognition for Driving Scenario Detection in Real Driving Data. In Proceedings of the 2020 IEEE Intelligent Vehicles Symposium (IV), Las Vegas, NV, USA, 19 October–13 November 2020; pp. 590–597.
54. Huang, X.; Cheng, X.; Geng, Q.; Cao, B.; Zhou, D.; Wang, P.; Lin, Y.; Yang, R. The Apolloscape Dataset for Autonomous Driving. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, Salt Lake City, UT, USA, 18–22 June 2018; pp. 954–960.
55. Krajewski, R.; Bock, J.; Kloeker, L.; Eckstein, L. The Highd Dataset: A Drone Dataset of Naturalistic Vehicle Trajectories on German Highways for Validation of Highly Automated Driving Systems. In Proceedings of the 2018 21st International Conference on Intelligent Transportation Systems (ITSC), Maui, HI, USA, 4–7 November 2018; pp. 2118–2125.
56. Bock, J.; Krajewski, R.; Moers, T.; Runde, S.; Vater, L.; Eckstein, L. The Ind Dataset: A Drone Dataset of Naturalistic Road User Trajectories at German Intersections. In Proceedings of the 2020 IEEE Intelligent Vehicles Symposium (IV), Las Vegas, NV, USA, 19 October–13 November 2020; pp. 1929–1934.
57. Krajewski, R.; Moers, T.; Bock, J.; Vater, L.; Eckstein, L. The Round Dataset: A Drone Dataset of Road User Trajectories at Roundabouts in Germany. In Proceedings of the 2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC), Rhodes, Greece, 20–23 September 2020; pp. 1–6.
58. Punzo, V.; Borzacchiello, M.T.; Ciuffo, B. On the Assessment of Vehicle Trajectory Data Accuracy and Application to the Next Generation SIMulation (NGSIM) Program Data. *Transp. Res. C Emerg. Technol.* **2011**, *19*, 1243–1262. [CrossRef]
59. Strigel, E.; Meissner, D.; Seeliger, F.; Wilking, B.; Dietmayer, K. The Ko-per Intersection Laserscanner and Video Dataset. In Proceedings of the 17th International IEEE Conference on Intelligent Transportation Systems (ITSC), Qingdao, China, 8–11 October 2014; pp. 1900–1901.
60. Kang, Y.; Yin, H.; Berger, C. Test Your Self-Driving Algorithm: An Overview of Publicly Available Driving Datasets and Virtual Testing Environments. *IEEE Trans. Intell. Veh.* **2019**, *4*, 171–185. [CrossRef]

61. Dingus, T.A.; Klauer, S.G.; Neale, V.L.; Petersen, A.; Lee, S.E.; Sudweeks, J.; Perez, M.A.; Hankey, J.; Ramsey, D.; Gupta, S. *The 100-Car Naturalistic Driving Study, Phase II-Results of the 100-Car Field Experiment*; Department of Transportation, National Highway Traffic Safety: Washington, DC, USA, 2006.
62. Pham, Q.-H.; Sevestre, P.; Pahwa, R.S.; Zhan, H.; Pang, C.H.; Chen, Y.; Mustafa, A.; Chandrasekhar, V.; Lin, J. A* 3D Dataset: Towards Autonomous Driving in Challenging Environments. In Proceedings of the 2020 IEEE International Conference on Robotics and Automation (ICRA), Paris, France, 31 May–31 August 2020; pp. 2267–2273.
63. Chang, M.-F.; Lambert, J.; Sangkloy, P.; Singh, J.; Bak, S.; Hartnett, A.; Wang, D.; Carr, P.; Lucey, S.; Ramanan, D. Argoverse: 3d Tracking and Forecasting with Rich Maps. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 8748–8757.
64. Yu, F.; Xian, W.; Chen, Y.; Liu, F.; Liao, M.; Madhavan, V.; Darrell, T. Bdd100k: A Diverse Driving Video Database with Scalable Annotation Tooling. *arXiv* **2018**, arXiv:1805.04687.
65. Brostow, G.J.; Fauqueur, J.; Cipolla, R. Semantic Object Classes in Video: A High-Definition Ground Truth Database. *Pattern Recognit. Lett.* **2009**, *30*, 88–97. [CrossRef]
66. Cityscapes Dataset–Semantic Understanding of Urban Street Scenes. Available online: https://www.cityscapes-dataset.com/ (accessed on 1 November 2022).
67. Zheng, O.; Abdel-Aty, M.; Yue, L.; Abdelraouf, A.; Wang, Z.; Mahmoud, N. CitySim: A Drone-Based Vehicle Trajectory Dataset for Safety Oriented Research and Digital Twins. *arXiv* **2022**, arXiv:2208.11036.
68. Zyner, A.; Worrall, S.; Nebot, E. Naturalistic Driver Intention and Path Prediction Using Recurrent Neural Networks. *IEEE Trans. Intell. Transp. Syst.* **2019**, *21*, 1584–1594. [CrossRef]
69. Patil, A.; Malla, S.; Gang, H.; Chen, Y.-T. The H3d Dataset for Full-Surround 3d Multi-Object Detection and Tracking in Crowded Urban Scenes. In Proceedings of the 2019 International Conference on Robotics and Automation (ICRA), Montreal, QC, Canada, 20–24 May 2019; pp. 9552–9557.
70. Zhan, W.; Sun, L.; Wang, D.; Shi, H.; Clausse, A.; Naumann, M.; Kummerle, J.; Konigshof, H.; Stiller, C.; de La Fortelle, A. Interaction Dataset: An International, Adversarial and Cooperative Motion Dataset in Interactive Driving Scenarios with Semantic Maps. *arXiv* **2019**, arXiv:1910.03088.
71. Choi, Y.; Kim, N.; Hwang, S.; Park, K.; Yoon, J.S.; An, K.; Kweon, I.S. KAIST Multi-Spectral Day/Night Data Set for Autonomous and Assisted Driving. *IEEE Trans. Intell. Transp. Syst.* **2018**, *19*, 934–948. [CrossRef]
72. Geiger, A.; Lenz, P.; Stiller, C.; Urtasun, R. Vision Meets Robotics: The Kitti Dataset. *Int. J. Robot. Res.* **2013**, *32*, 1231–1237. [CrossRef]
73. Houston, J.; Zuidhof, G.; Bergamini, L.; Ye, Y.; Chen, L.; Jain, A.; Omari, S.; Iglovikov, V.; Ondruska, P. One Thousand and One Hours: Self-Driving Motion Prediction Dataset. In Proceedings of the Conference on Robot Learning, PMLR, London, UK, 8–11 November 2021; pp. 409–418.
74. Caesar, H.; Bankiti, V.; Lang, A.H.; Vora, S.; Liong, V.E.; Xu, Q.; Krishnan, A.; Pan, Y.; Baldan, G.; Beijbom, O. Nuscenes: A Multimodal Dataset for Autonomous Driving. In Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 11621–11631.
75. Maddern, W.; Pascoe, G.; Linegar, C.; Newman, P. 1 Year, 1000 Km: The Oxford RobotCar Dataset. *Int. J. Robot. Res.* **2017**, *36*, 3–15. [CrossRef]
76. Robicquet, A.; Sadeghian, A.; Alahi, A.; Savarese, S. Learning Social Etiquette: Human Trajectory Prediction in Crowded Scenes. In Proceedings of the European Conference on Computer Vision (ECCV), Glasgow, UK, 23–28 August 2020; Volume 2.
77. Wu, F.; Wang, D.; Hwang, M.; Hao, C.; Lu, J.; Zhang, J.; Chou, C.; Darrell, T.; Bayen, A. Decentralized Vehicle Coordination: The Berkeley DeepDrive Drone Dataset. *arXiv* **2022**, arXiv:2209.08763.
78. Chandra, R.; Bhattacharya, U.; Bera, A.; Manocha, D. Traphic: Trajectory Prediction in Dense and Heterogeneous Traffic Using Weighted Interactions. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 8483–8492.
79. Li, X.; Flohr, F.; Yang, Y.; Xiong, H.; Braun, M.; Pan, S.; Li, K.; Gavrila, D.M. A New Benchmark for Vision-Based Cyclist Detection. In Proceedings of the 2016 IEEE Intelligent Vehicles Symposium (IV), Gothenburg, Sweden, 19–22 June 2016; pp. 1028–1033.
80. Zipfl, M.; Fleck, T.; Zofka, M.R.; Zöllner, J.M. From Traffic Sensor Data to Semantic Traffic Descriptions: The Test Area Autonomous Driving Baden-Württemberg Dataset (TAF-BW Dataset). In Proceedings of the 2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC), Rhodes, Greece, 20–23 September 2020; pp. 1–7.
81. Yang, Z.; Zhang, Y.; Yu, J.; Cai, J.; Luo, J. End-to-End Multi-Modal Multi-Task Vehicle Control for Self-Driving Cars with Visual Perceptions. In Proceedings of the 2018 24th International Conference on Pattern Recognition (ICPR), Beijing, China, 20–24 August 2018; pp. 2289–2294.
82. Sun, P.; Kretzschmar, H.; Dotiwalla, X.; Chouard, A.; Patnaik, V.; Tsui, P.; Guo, J.; Zhou, Y.; Chai, Y.; Caine, B. Scalability in Perception for Autonomous Driving: Waymo Open Dataset. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 2446–2454.
83. Goss, Q.; AlRashidi, Y.; Akbaş, M.İ. Generation of Modular and Measurable Validation Scenarios for Autonomous Vehicles Using Accident Data. In Proceedings of the 2021 IEEE Intelligent Vehicles Symposium (IV), Nagoya, Japan, 11–17 July 2021; pp. 251–257.
84. Rabbani, M.B.A.; Musarat, M.A.; Alaloul, W.S.; Ayub, S.; Bukhari, H.; Altaf, M. Road Accident Data Collection Systems in Developing and Developed Countries: A Review. *Int. J. Integr. Eng.* **2022**, *14*, 336–352. [CrossRef]

85. So, J.J.; Park, I.; Wee, J.; Park, S.; Yun, I. Generating Traffic Safety Test Scenarios for Automated Vehicles Using a Big Data Technique. *KSCE J. Civ. Eng.* **2019**, *23*, 2702–2712. [CrossRef]
86. Xinxin, Z.; Fei, L.; Xiangbin, W. CSG: Critical Scenario Generation from Real Traffic Accidents. In Proceedings of the 2020 IEEE Intelligent Vehicles Symposium (IV), Las Vegas, NV, USA, 19 October–13 November 2020; pp. 1330–1336.
87. Ahmed, A.; Sadullah, A.F.M.; Yahya, A.S. Errors in Accident Data, Its Types, Causes and Methods of Rectification-Analysis of the Literature. *Accid. Anal. Prev.* **2019**, *130*, 3–21. [CrossRef]
88. Hu, L.; Bao, X.; Lin, M.; Yu, C.; Wang, F. Research on Risky Driving Behavior Evaluation Model Based on CIDAS Real Data. *Proc. Inst. Mech. Eng. D J. Automob. Eng.* **2021**, *235*, 2176–2187. [CrossRef]
89. Tan, Z.; Che, Y.; Xiao, L.; Hu, W.; Li, P.; Xu, J. Research of Fatal Car-to-Pedestrian Precrash Scenarios for the Testing of the Active Safety System in China. *Accid. Anal. Prev.* **2021**, *150*, 105857. [CrossRef] [PubMed]
90. Otte, D.; Jänsch, M.; Haasper, C. Injury Protection and Accident Causation Parameters for Vulnerable Road Users Based on German In-Depth Accident Study GIDAS. *Accid. Anal. Prev.* **2012**, *44*, 149–153. [CrossRef] [PubMed]
91. Kaplan, S.; Prato, C.G. Risk Factors Associated with Bus Accident Severity in the United States: A Generalized Ordered Logit Model. *J. Saf. Res.* **2012**, *43*, 171–180. [CrossRef] [PubMed]
92. Lemmen, P.; Fagerlind, H.; Unselt, T.; Rodarius, C.; Infantes, E.; van der Zweep, C. Assessment of Integrated Vehicle Safety Systems for Improved Vehicle Safety. *Procedia-Soc. Behav. Sci.* **2012**, *48*, 1632–1641. [CrossRef]
93. Richards, D.C.; Cookson, R.E.; Cuerden, R.W. *Linking Accidents in National Statistics to In-Depth Accident Data*; IHS: Bracknell, UK, 2010; ISBN 1846088909.
94. Bauder, M.; Lecheler, K.; Wech, L.; Böhm, K.; Paula, D.; Schweiger, H.-G. Determination of Accident Scenarios via Freely Available Accident Databases. *Open Eng.* **2022**, *12*, 453–467. [CrossRef]
95. Hu, L.; Hu, X.; Wang, J.; Kuang, A.; Hao, W.; Lin, M. Casualty Risk of E-Bike Rider Struck by Passenger Vehicle Using China in-Depth Accident Data. *Traffic Inj. Prev.* **2020**, *21*, 283–287. [CrossRef]
96. Johannsen, H. *Unfallmechanik und Unfallrekonstruktion*; Springer: Berlin/Heidelberg, Germany, 2013.
97. Moosavi, S.; Samavatian, M.H.; Parthasarathy, S.; Ramnath, R. A Countrywide Traffic Accident Dataset. *arXiv* **2019**, arXiv:1906.05409.
98. Chen, Q.; Chen, Y.; Bostrom, O.; Ma, Y.; Liu, E. *A Comparison Study of Car-to-Pedestrian and Car-to-E-Bike Accidents: Data Source: The China in-Depth Accident Study (CIDAS)*; SAE Technical Paper; SAE International: Warrendale, PA, USA, 2014.
99. El Tayeb, A.A.; Pareek, V.; Araar, A. Applying Association Rules Mining Algorithms for Traffic Accidents in Dubai. *Int. J. Soft Comput. Eng.* **2015**, *5*, 1–12.
100. Home | GIDAS. Available online: https://www.gidas.org/start-en.html (accessed on 1 November 2022).
101. Kim, S.; Lee, J.; Youn, Y. A Study on the Construction of the Database Structure for the Korea In-Depth Accident Study. *Trans. Korean Soc. Automot. Eng.* **2014**, *22*, 29–36. [CrossRef]
102. Chung, Y. Development of an Accident Duration Prediction Model on the Korean Freeway Systems. *Accid. Anal. Prev.* **2010**, *42*, 282–289. [CrossRef]
103. Mooney, P.; Corcoran, P. Characteristics of Heavily Edited Objects in OpenStreetMap. *Future Internet* **2011**, *4*, 285. [CrossRef]
104. Deng, B.; Wang, H.; Chen, J.; Wang, X.; Chen, X. Traffic Accidents in Shanghai—General Statistics and in-Depth Analysis. In Proceedings of the 23rd International Technical Conference on the Enhanced Safety of Vehicles, Seoul, Republic of Korea, 27–30 May 2013.
105. Tay, R.; Rifaat, S.M. Factors Contributing to the Severity of Intersection Crashes. *J. Adv. Transp.* **2007**, *41*, 245–265. [CrossRef]
106. Clarke, D.D.; Ward, P.; Truman, W. *In-Depth Accident Causation Study of Young Drivers*; TRL Report 542; TRL Limited: Crowthorne, Berkshire, UK, 2002.
107. Shah, A.P.; Lamare, J.-B.; Nguyen-Anh, T.; Hauptmann, A. CADP: A Novel Dataset for CCTV Traffic Camera Based Accident Analysis. In Proceedings of the 2018 15th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS), Auckland, New Zealand, 27–30 November 2018; pp. 1–9.
108. Krajzewicz, D. Traffic Simulation with SUMO–Simulation of Urban Mobility. In *Fundamentals of Traffic Simulation*; Springer: Berlin/Heidelberg, Germany, 2010; pp. 269–293.
109. Li, X.; Wang, Y.; Yan, L.; Wang, K.; Deng, F.; Wang, F.-Y. ParallelEye-CS: A New Dataset of Synthetic Images for Testing the Visual Intelligence of Intelligent Vehicles. *IEEE Trans. Veh. Technol.* **2019**, *68*, 9619–9631. [CrossRef]
110. Liu, C.; Liu, Z.; Chai, Y.; Liu, T. Review of Virtual Traffic Simulation and Its Applications. *J. Adv. Transp.* **2020**, *2020*, 8237649. [CrossRef]
111. Vrbanić, F.; Čakija, D.; Kušić, K.; Ivanjko, E. Traffic Flow Simulators with Connected and Autonomous Vehicles: A Short Review. *Transform. Transp.* **2021**, 15–30. [CrossRef]
112. Li, A.; Chen, S.; Sun, L.; Zheng, N.; Tomizuka, M.; Zhan, W. SceGene: Bio-Inspired Traffic Scenario Generation for Autonomous Driving Testing. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 14859–14874. [CrossRef]
113. Zofka, M.R.; Ulbrich, S.; Karl, D.; Fleck, T.; Kohlhaas, R.; Rönnau, A.; Dillmann, R.; Zöllner, J.M. Traffic Participants in the Loop: A Mixed Reality-Based Interaction Testbed for the Verification and Validation of Autonomous Vehicles. In Proceedings of the 2018 21st International Conference on Intelligent Transportation Systems (ITSC), Maui, HI, USA, 4–7 November 2018; pp. 3583–3590.
114. Feng, S.; Feng, Y.; Yan, X.; Shen, S.; Xu, S.; Liu, H.X. Safety Assessment of Highly Automated Driving Systems in Test Tracks: A New Framework. *Accid. Anal. Prev.* **2020**, *144*, 105664. [CrossRef]

115. Di-Castro, S.; di Castro, D.; Mannor, S. Sim and Real: Better Together. *Adv. Neural Inf. Process. Syst.* **2021**, *34*, 6868–6880.
116. Hauer, F.; Gerostathopoulos, I.; Schmidt, T.; Pretschner, A. Clustering Traffic Scenarios Using Mental Models as Little as Possible. In Proceedings of the 2020 IEEE Intelligent Vehicles Symposium (IV), Las Vegas, NV, USA, 19 October–13 November 2020; pp. 1007–1012.
117. Jacobo, A.-M.; Nobuyuki, U.; Kunio, Y.; Koichiro, O.; Eiichi, K.; Satoshi, T. Development of a Safety Assurance Process for Autonomous Vehicles in Japan. In Proceedings of the ESV Conference 2019, Eindhoven, The Netherlands, 10–13 June 2019.
118. Yu, X. Improvement and Validation of a Highway Traffic Complexity Metric for Test Scenarios of Automated Vehicles. Master's Thesis, University of Munich, Munich, Germany, 2020.
119. Wang, C.; Winner, H. Overcoming Challenges of Validation Automated Driving and Identification of Critical Scenarios. In Proceedings of the 2019 IEEE Intelligent Transportation Systems Conference (ITSC), Auckland, New Zealand, 27–30 October 2019; pp. 2639–2644.
120. Hasan, B.M.S.; Abdulazeez, A.M. A Review of Principal Component Analysis Algorithm for Dimensionality Reduction. *J. Soft Comput. Data Min.* **2021**, *2*, 20–30.
121. Jia, W.; Sun, M.; Lian, J.; Hou, S. Feature Dimensionality Reduction: A Review. *Complex Intell. Syst.* **2022**, *8*, 2663–2693. [CrossRef]
122. Oskolkov, N. Dimensionality Reduction. *Appl. Data Sci. Tour.* **2022**, 151–167. [CrossRef]
123. Paleyes, A.; Urma, R.-G.; Lawrence, N.D. Challenges in Deploying Machine Learning: A Survey of Case Studies. *ACM Comput. Surv. (CSUR)* **2021**. [CrossRef]
124. Lu, Y.; Cohen, I.; Zhou, X.S.; Tian, Q. Feature Selection Using Principal Feature Analysis. In Proceedings of the 15th ACM International Conference on Multimedia, Augsburg, Germany, 25–29 September 2007; pp. 301–304.
125. Wagner, S.; Knoll, A.; Groh, K.; Kühbeck, T.; Watzenig, D.; Eckstein, L. Virtual Assessment of Automated Driving: Methodology, Challenges, and Lessons Learned. *SAE Int. J. Connect. Autom. Veh.* **2019**, *2*. [CrossRef]
126. de Gelder, E.; Hof, J.; Cator, E.; Paardekooper, J.-P.; den Camp, O.O.; Ploeg, J.; de Schutter, B. Scenario Parameter Generation Method and Scenario Representativeness Metric for Scenario-Based Assessment of Automated Vehicles. *IEEE Trans. Intell. Transp. Syst.* **2022**, 1–14. [CrossRef]
127. Anowar, F.; Sadaoui, S.; Selim, B. Conceptual and Empirical Comparison of Dimensionality Reduction Algorithms (Pca, Kpca, Lda, Mds, Svd, Lle, Isomap, Le, Ica, t-Sne). *Comput. Sci. Rev.* **2021**, *40*, 100378. [CrossRef]
128. Hoseini, F.S.; Rahrovani, S.; Chehreghani, M.H. A Generic Framework for Clustering Vehicle Motion Trajectories. *arXiv* **2020**, arXiv:2009.12443.
129. Demetriou, A.; Alfsvåg, H.; Rahrovani, S.; Chehreghani, M.H. A Deep Learning Framework for Generation and Analysis of Driving Scenario Trajectories. *arXiv* **2020**, arXiv:2007.14524.
130. Schubert, E.; Sander, J.; Ester, M.; Kriegel, H.P.; Xu, X. DBSCAN Revisited, Revisited: Why and How You Should (Still) Use DBSCAN. *ACM Trans. Database Syst. (TODS)* **2017**, *42*, 19. [CrossRef]
131. Carreira-Perpinán, M.A. A Review of Dimension Reduction Techniques. *Dep. Comput. Sci. Univ. Sheffield Tech. Rep.* **1997**, *9*, 1–69.
132. Karunakaran, D.; Berrio, J.S.; Worrall, S.; Nebot, E. Automatic Lane Change Scenario Extraction and Generation of Scenarios in OpenX Format from Real-World Data. *arXiv* **2022**, arXiv:2203.07521.
133. Elrofai, H.; Worm, D.; Op den Camp, O. Scenario Identification for Validation of Automated Driving Functions. In *Advanced Microsystems for Automotive Applications 2016*; Springer: Berlin/Heidelberg, Germany, 2016; pp. 153–163.
134. De Gelder, E.; Manders, J.; Grappiolo, C.; Paardekooper, J.-P.; den Camp, O.O.; de Schutter, B. Real-World Scenario Mining for the Assessment of Automated Vehicles. In Proceedings of the 2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC), Rhodes, Greece, 20–23 September 2020; pp. 1–8.
135. Riedmaier, S.; Schneider, D.; Watzenig, D.; Diermeyer, F.; Schick, B. Model Validation and Scenario Selection for Virtual-Based Homologation of Automated Vehicles. *Appl. Sci.* **2020**, *11*, 35. [CrossRef]
136. Montanari, F.; Stadler, C.; Sichermann, J.; German, R.; Djanatliev, A. Maneuver-Based Resimulation of Driving Scenarios Based on Real Driving Data. In Proceedings of the 2021 IEEE Intelligent Vehicles Symposium (IV), Nagoya, Japan, 11–17 July 2021; pp. 1124–1131.
137. Paardekooper, J.-P.; Montfort, S.; Manders, J.; Goos, J.; de Gelder, E.; den Camp, O.O.; Bracquemond, A.; Thiolon, G. Automatic Identification of Critical Scenarios in a Public Dataset of 6000 Km of Public-Road Driving. In Proceedings of the 26th International Technical Conference on the Enhanced Safety of Vehicles (ESV), Eindhoven, The Netherlands, 10–13 June 2019; Mira Smart. pp. 1–8.
138. Hartjen, L.; Philipp, R.; Schuldt, F.; Friedrich, B.; Howar, F. Classification of Driving Maneuvers in Urban Traffic for Parametrization of Test Scenarios. In Proceedings of the 9 Tagung Automatisiertes Fahren, Munich, Germany, 21–22 November 2019.
139. Saini, P.; Kaur, J.; Lamba, S. A Review on Pattern Recognition Using Machine Learning. *Adv. Mech. Eng.* **2021**, 619–627. [CrossRef]
140. Kruber, F.; Wurst, J.; Botsch, M. An Unsupervised Random Forest Clustering Technique for Automatic Traffic Scenario Categorization. In Proceedings of the 2018 21st International Conference on Intelligent Transportation Systems (ITSC), Maui, HI, USA, 4–7 November 2018; pp. 2811–2818.
141. Langner, J.; Grolig, H.; Otten, S.; Holzäpfel, M.; Sax, E. Logical Scenario Derivation by Clustering Dynamic-Length-Segments Extracted from Real-World-Driving-Data. In Proceedings of the VEHITS 2019, Crete, Greece, 3–5 May 2019; pp. 458–467.

142. Deng, N.; Jiang, K.; Cao, Z.; Zhou, W.; Yang, D. Decision-Oriented Driving Scenario Recognition Based on Unsupervised Learning. In Proceedings of the 21st COTA International Conference of Transportation Professionals 2021, Xi'an, China, 16–19 December 2021; pp. 564–573.
143. Li, S.; Wang, W.; Mo, Z.; Zhao, D. Clustering of Naturalistic Driving Encounters Using Unsupervised Learning. *arXiv* **2018**, arXiv:1802.10214.
144. Weber, N.; Thiem, C.; Konigorski, U. UnScenE: Toward Unsupervised Scenario Extraction for Automated Driving Systems from Urban Naturalistic Road Traffic Data. *arXiv* **2022**, arXiv:2202.06608.
145. Vrbanić, F.; Miletić, M.; Ivanjko, E.; Majstorović, Ž. Creating Representative Urban Motorway Traffic Scenarios: Initial Observations. In Proceedings of the 2021 International Symposium ELMAR, Zadar, Croatia, 13–15 September 2021; pp. 183–188.
146. Montanari, F.; Ren, H.; Djanatliev, A. Scenario Detection in Unlabeled Real Driving Data with a Rule-Based State Machine Supported by a Recurrent Neural Network. In Proceedings of the 2021 IEEE 93rd Vehicular Technology Conference (VTC2021-Spring), Helsinki, Finland, 25–28 April 2021; pp. 1–5.
147. De Gelder, E.; Paardekooper, J.-P. Assessment of Automated Driving Systems Using Real-Life Scenarios. In Proceedings of the 2017 IEEE Intelligent Vehicles Symposium (IV), Los Angeles, CA, USA, 11–14 June 2017; pp. 589–594.
148. Huang, Z.; Guo, Y.; Arief, M.; Lam, H.; Zhao, D. A Versatile Approach to Evaluating and Testing Automated Vehicles Based on Kernel Methods. In Proceedings of the 2018 Annual American Control Conference (ACC), Milwaukee, WI, USA, 27–29 June 2018; pp. 4796–4802.
149. Kuhn, D.R.; Bryce, R.; Duan, F.; Ghandehari, L.S.; Lei, Y.; Kacker, R.N. Combinatorial Testing: Theory and Practice. *Adv. Comput.* **2015**, *99*, 1–66.
150. Gao, F.; Duan, J.; He, Y.; Wang, Z. A Test Scenario Automatic Generation Strategy for Intelligent Driving Systems. *Math. Probl. Eng.* **2019**, *2019*, 3737486. [CrossRef]
151. Kuhn, D.R.; Reilly, M.J. An Investigation of the Applicability of Design of Experiments to Software Testing. In Proceedings of the 27th Annual NASA Goddard/IEEE Software Engineering Workshop, Greenbelt, MD, USA, 5–6 December 2002; pp. 91–95.
152. Hartjen, L.; Schuldt, F.; Friedrich, B. Semantic Classification of Pedestrian Traffic Scenarios for the Validation of Automated Driving. In Proceedings of the 2019 IEEE Intelligent Transportation Systems Conference (ITSC), Auckland, New Zealand, 27–30 October 2019; pp. 3696–3701.
153. Xia, Q.; Duan, J.; Gao, F.; Chen, T.; Yang, C. *Automatic Generation Method of Test Scenario for Adas Based on Complexity*; SAE Technical Paper; SAE International: Warrendale, PA, USA, 2017.
154. Tao, J.; Li, Y.; Wotawa, F.; Felbinger, H.; Nica, M. On the Industrial Application of Combinatorial Testing for Autonomous Driving Functions. In Proceedings of the 2019 IEEE International Conference on Software Testing, Verification and Validation Workshops (ICSTW), Xi'an, China, 22–23 April 2019; pp. 234–240.
155. Birkemeyer, L.; Pett, T.; Vogelsang, A.; Seidl, C.; Schaefer, I. Feature-Interaction Sampling for Scenario-Based Testing of Advanced Driver Assistance Systems. In Proceedings of the 16th International Working Conference on Variability Modelling of Software-Intensive Systems, Florence, Italy, 23–25 February 2022; pp. 1–10.
156. Laurent, T.; Arcaini, P.; Ishikawa, F.; Ventresque, A. A Mutation-Based Approach for Assessing Weight Coverage of a Path Planner. In Proceedings of the 2019 26th Asia-Pacific Software Engineering Conference (APSEC), Putrajaya, Malaysia, 2–5 December 2019; pp. 94–101.
157. Huang, Z.; Lam, H.; LeBlanc, D.J.; Zhao, D. Accelerated Evaluation of Automated Vehicles Using Piecewise Mixture Models. *IEEE Trans. Intell. Transp. Syst.* **2017**, *19*, 2845–2855. [CrossRef]
158. De Gelder, E.; Elrofai, H.; Saberi, A.K.; Paardekooper, J.-P.; den Camp, O.O.; de Schutter, B. Risk Quantification for Automated Driving Systems in Real-World Driving Scenarios. *IEEE Access* **2021**, *9*, 168953–168970. [CrossRef]
159. Zhao, D.; Lam, H.; Peng, H.; Bao, S.; LeBlanc, D.J.; Nobukawa, K.; Pan, C.S. Accelerated Evaluation of Automated Vehicles Safety in Lane-Change Scenarios Based on Importance Sampling Techniques. *IEEE Trans. Intell. Transp. Syst.* **2016**, *18*, 595–607. [CrossRef]
160. Xu, Y.; Zou, Y.; Sun, J. Accelerated Testing for Automated Vehicles Safety Evaluation in Cut-in Scenarios Based on Importance Sampling, Genetic Algorithm and Simulation Applications. *J. Intell. Connect. Veh.* **2018**, *1*, 28–38. [CrossRef]
161. Zhang, S.; Peng, H.; Zhao, D.; Tseng, H.E. Accelerated Evaluation of Autonomous Vehicles in the Lane Change Scenario Based on Subset Simulation Technique. In Proceedings of the 2018 21st International Conference on Intelligent Transportation Systems (ITSC), Maui, HI, USA, 4–7 November 2018; pp. 3935–3940.
162. Akagi, Y.; Kato, R.; Kitajima, S.; Antona-Makoshi, J.; Uchida, N. A Risk-Index Based Sampling Method to Generate Scenarios for the Evaluation of Automated Driving Vehicle Safety. In Proceedings of the 2019 IEEE Intelligent Transportation Systems Conference (ITSC), Auckland, New Zealand, 27–30 October 2019; pp. 667–672.
163. Li, G.; Li, Y.; Jha, S.; Tsai, T.; Sullivan, M.; Hari, S.K.S.; Kalbarczyk, Z.; Iyer, R. AV-FUZZER: Finding Safety Violations in Autonomous Driving Systems. In Proceedings of the 2020 IEEE 31st International Symposium on Software Reliability Engineering (ISSRE), Coimbra, Portugal, 12–15 October 2020; pp. 25–36.
164. Klischat, M.; Liu, E.I.; Holtke, F.; Althoff, M. Scenario Factory: Creating Safety-Critical Traffic Scenarios for Automated Vehicles. In Proceedings of the 2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC), Rhodes, Greece, 20–23 September 2020; pp. 1–7.

165. Zhu, B.; Zhang, P.; Zhao, J.; Deng, W. Hazardous Scenario Enhanced Generation for Automated Vehicle Testing Based on Optimization Searching Method. *IEEE Trans. Intell. Transp. Syst.* **2021**, *23*, 7321–7331. [CrossRef]
166. Feng, S.; Feng, Y.; Sun, H.; Zhang, Y.; Liu, H.X. Testing Scenario Library Generation for Connected and Automated Vehicles: An Adaptive Framework. *IEEE Trans. Intell. Transp. Syst.* **2020**, *23*, 1213–1222. [CrossRef]
167. Tuncali, C.E.; Fainekos, G. Rapidly-Exploring Random Trees-Based Test Generation for Autonomous Vehicles. *arXiv* **2019**, arXiv:1903.10629.
168. Bussler, A.; Hartjen, L.; Philipp, R.; Schuldt, F. Application of Evolutionary Algorithms and Criticality Metrics for the Verification and Validation of Automated Driving Systems at Urban Intersections. In Proceedings of the 2020 IEEE Intelligent Vehicles Symposium (IV), Las Vegas, NV, USA, 19 October–13 November 2020; pp. 128–135.
169. Klischat, M.; Althoff, M. Generating Critical Test Scenarios for Automated Vehicles with Evolutionary Algorithms. In Proceedings of the 2019 IEEE Intelligent Vehicles Symposium (IV), Paris, France, 9–12 June 2019; pp. 2352–2358.
170. Almanee, S.; Wu, X.; Huai, Y.; Chen, Q.A.; Garcia, J. ScenoRITA: Generating Less-Redundant, Safety-Critical and Motion Sickness-Inducing Scenarios for Autonomous Vehicles. *arXiv* **2021**, arXiv:2112.09725.
171. Hauer, F.; Pretschner, A.; Holzmüller, B. Fitness Functions for Testing Automated and Autonomous Driving Systems. In Proceedings of the International Conference on Computer Safety, Reliability, and Security, Turku, Finland, 10–13 September 2019; Springer: Berlin/Heidelberg, Germany, 2019; pp. 69–84.
172. Hejase, M.; Ozguner, U.; Barbier, M.; Ibanez-Guzman, J. A Methodology for Model-Based Validation of Autonomous Vehicle Systems. In Proceedings of the 2020 IEEE Intelligent Vehicles Symposium (IV), Las Vegas, NV, USA, 19 October–13 November 2020; pp. 2097–2103.
173. Calò, A.; Arcaini, P.; Ali, S.; Hauer, F.; Ishikawa, F. Simultaneously Searching and Solving Multiple Avoidable Collisions for Testing Autonomous Driving Systems. In Proceedings of the 2020 Genetic and Evolutionary Computation Conference, Cancún, Mexico, 8–12 July 2020; pp. 1055–1063.
174. Ben Abdessalem, R.; Nejati, S.; Briand, L.C.; Stifter, T. Testing Vision-Based Control Systems Using Learnable Evolutionary Algorithms. In Proceedings of the 2018 IEEE/ACM 40th International Conference on Software Engineering (ICSE), Gothenburg, Sweden, 27 May–3 June 2018; pp. 1016–1026.
175. Koschi, M.; Pek, C.; Maierhofer, S.; Althoff, M. Computationally Efficient Safety Falsification of Adaptive Cruise Control Systems. In Proceedings of the 2019 IEEE Intelligent Transportation Systems Conference (ITSC), Auckland, New Zealand, 27–30 October 2019; pp. 2879–2886.
176. Song, Y.; Chitturi, M.V.; Noyce, D.A. Automated Vehicle Crash Sequences: Patterns and Potential Uses in Safety Testing. *Accid. Anal. Prev.* **2021**, *153*, 106017. [CrossRef]
177. Karunakaran, D.; Worrall, S.; Nebot, E. Efficient Falsification Approach for Autonomous Vehicle Validation Using a Parameter Optimisation Technique Based on Reinforcement Learning. *arXiv* **2020**, arXiv:2011.07699.
178. Baumann, D.; Pfeffer, R.; Sax, E. Automatic Generation of Critical Test Cases for the Development of Highly Automated Driving Functions. In Proceedings of the 2021 IEEE 93rd Vehicular Technology Conference (VTC2021-Spring), Helsinki, Finland, 25–28 April 2021; pp. 1–5.
179. Xu, L.; Zhang, C.; Liu, Y.; Wang, L.; Li, L. Worst Perception Scenario Search for Autonomous Driving. In Proceedings of the 2020 IEEE Intelligent Vehicles Symposium (IV), Las Vegas, NV, USA, 19 October–13 November 2020; pp. 1702–1707.
180. Koren, M.; Alsaif, S.; Lee, R.; Kochenderfer, M.J. Adaptive Stress Testing for Autonomous Vehicles. In Proceedings of the 2018 IEEE Intelligent Vehicles Symposium (IV), Changshu, China, 26–30 June 2018; pp. 1–7.
181. Koren, M.; Nassar, A.; Kochenderfer, M.J. Finding Failures in High-Fidelity Simulation Using Adaptive Stress Testing and the Backward Algorithm. In Proceedings of the 2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Prague, Czech Republic, 27 September–1 October 2021; pp. 5944–5949.
182. Capito, L.; Weng, B.; Ozguner, U.; Redmill, K. A Modeled Approach for Online Adversarial Test of Operational Vehicle Safety. In Proceedings of the 2021 American Control Conference (ACC), New Orleans, LA, USA, 25–28 May 2021; pp. 398–404.
183. Zhang, H.; Zhou, H.; Sun, J.; Tian, Y. Risk Assessment of Highly Automated Vehicles with Naturalistic Driving Data: A Surrogate-Based Optimization Method. In Proceedings of the 2022 IEEE Intelligent Vehicles Symposium (IV), Aachen, Germany, 5–9 June 2022; pp. 580–585.
184. Feng, S.; Feng, Y.; Yu, C.; Zhang, Y.; Liu, H.X. Testing Scenario Library Generation for Connected and Automated Vehicles, Part I: Methodology. *IEEE Trans. Intell. Transp. Syst.* **2020**, *22*, 1573–1582. [CrossRef]
185. Feng, S.; Feng, Y.; Sun, H.; Bao, S.; Zhang, Y.; Liu, H.X. Testing Scenario Library Generation for Connected and Automated Vehicles, Part II: Case Studies. *IEEE Trans. Intell. Transp. Syst.* **2020**, *22*, 5635–5647. [CrossRef]
186. Batsch, F.; Daneshkhah, A.; Cheah, M.; Kanarachos, S.; Baxendale, A. Performance Boundary Identification for the Evaluation of Automated Vehicles Using Gaussian Process Classification. In Proceedings of the 2019 IEEE Intelligent Transportation Systems Conference (ITSC), Auckland, New Zealand, 27–30 October 2019; pp. 419–424.
187. Batsch, F.; Daneshkhah, A.; Palade, V.; Cheah, M. Scenario Optimisation and Sensitivity Analysis for Safe Automated Driving Using Gaussian Processes. *Appl. Sci.* **2021**, *11*, 775. [CrossRef]
188. Wang, Y.; Yu, R.; Qiu, S.; Sun, J.; Farah, H. Safety Performance Boundary Identification of Highly Automated Vehicles: A Surrogate Model-Based Gradient Descent Searching Approach. *IEEE Trans. Intell. Transp. Syst.* **2022**, 1–12. [CrossRef]

189. Westhofen, L.; Neurohr, C.; Koopmann, T.; Butz, M.; Schütt, B.; Utesch, F.; Neurohr, B.; Gutenkunst, C.; Böde, E. Criticality Metrics for Automated Driving: A Review and Suitability Analysis of the State of the Art. *Arch. Comput. Methods Eng.* **2022**, 1–35. [CrossRef]
190. Jansson, J. *Collision Avoidance Theory: With Application to Automotive Collision Mitigation*; Linköping University Electronic Press: Linköping, Sweden, 2005.
191. Wachenfeld, W.; Junietz, P.; Wenzel, R.; Winner, H. The Worst-Time-to-Collision Metric for Situation Identification. In Proceedings of the 2016 IEEE Intelligent Vehicles Symposium (IV), Gothenburg, Sweden, 19–22 June 2016; pp. 729–734.
192. Eggert, J. Predictive Risk Estimation for Intelligent ADAS Functions. In Proceedings of the 17th International IEEE Conference on Intelligent Transportation Systems (ITSC), Qingdao, China, 8–11 October 2014; pp. 711–718.
193. Minderhoud, M.M.; Bovy, P.H.L. Extended Time-to-Collision Measures for Road Traffic Safety Assessment. *Accid. Anal. Prev.* **2001**, *33*, 89–97. [CrossRef]
194. Varhelyi, A. Drivers' Speed Behaviour at a Zebra Crossing: A Case Study. *Accid. Anal. Prev.* **1998**, *30*, 731–743. [CrossRef]
195. Allen, B.L.; Shin, B.T.; Cooper, P.J. *Analysis of Traffic Conflicts and Collisions*; Transportation Research Record Department of Civil Engineering, McMaster University: Hamilton, ON, Canada, 1978.
196. Hupfer, C. Deceleration to Safety Time (DST)-a Useful Figure to Evaluate Traffic Safety. In Proceedings of the ICTCT Conference Proceedings of Seminar, Lund, Sweden, 5–7 November 1997; Volume 3, pp. 5–7.
197. Ghodsi, Z.; Hari, S.K.S.; Frosio, I.; Tsai, T.; Troccoli, A.; Keckler, S.W.; Garg, S.; Anandkumar, A. Generating and Characterizing Scenarios for Safety Testing of Autonomous Vehicles. In Proceedings of the 2021 IEEE Intelligent Vehicles Symposium (IV), Nagoya, Japan, 11–17 July 2021; pp. 157–164.
198. Jeong, H.; Kim, I.; Han, K.; Kim, J. Comprehensive Analysis of Traffic Accidents in Seoul: Major Factors and Types Affecting Injury Severity. *Appl. Sci.* **2022**, *12*, 1790. [CrossRef]
199. Cafiso, S.; Garcia, A.G.; Cavarra, R.; Rojas, M.A.R. Crosswalk Safety Evaluation Using a Pedestrian Risk Index as Traffic Conflict Measure. In Proceedings of the 3rd International Conference on Road safety and Simulation 2011, Indianapolis, IN, USA, 14–16 September 2011; pp. 1–15.
200. Cunto, F.; Saccomanno, F.F. Calibration and Validation of Simulated Vehicle Safety Performance at Signalized Intersections. *Accid. Anal. Prev.* **2008**, *40*, 1171–1179. [CrossRef]
201. Kolekar, S.; Gite, S.; Pradhan, B.; Kotecha, K. Behavior Prediction of Traffic Actors for Intelligent Vehicle Using Artificial Intelligence Techniques: A Review. *IEEE Access* **2021**, *9*, 135034–135058. [CrossRef]
202. Bolte, J.-A.; Bar, A.; Lipinski, D.; Fingscheidt, T. Towards Corner Case Detection for Autonomous Driving. In Proceedings of the 2019 IEEE Intelligent Vehicles Symposium (IV), Paris, France, 9–12 June 2019; pp. 438–445.
203. Schreier, M.; Willert, V.; Adamy, J. An Integrated Approach to Maneuver-Based Trajectory Prediction and Criticality Assessment in Arbitrary Road Environments. *IEEE Trans. Intell. Transp. Syst.* **2016**, *17*, 2751–2766. [CrossRef]
204. Huber, B.; Herzog, S.; Sippl, C.; German, R.; Djanatliev, A. Evaluation of Virtual Traffic Situations for Testing Automated Driving Functions Based on Multidimensional Criticality Analysis. In Proceedings of the 2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC), Rhodes, Greece, 20–23 September 2020; pp. 1–7.
205. de Gelder, E.; Saberi, A.K.; Elrofai, H. A Method for Scenario Risk Quantification for Automated Driving Systems. In Proceedings of the 26th International Technical Conference on the Enhanced Safety of Vehicles (ESV), Eindhoven, The Netherlands, 10–13 June 2019. Mira Smart.
206. Liu, H.X.; Feng, S. "Curse of Rarity" for Autonomous Vehicles. *arXiv* **2022**, arXiv:2207.02749.