# AnoVox: A Benchmark for Multimodal Anomaly Detection in Autonomous Driving

Daniel Bogdoll[1,2]*, Iramm Hamdard[1,2]*, Lukas Namgyu Rößler[2]*, Felix Geisler[2], Muhammed Bayram[2], Felix Wang[2], Jan Imhof[1,2], Miguel de Campos[3], Anushervon Tabarov[1,2], Yitian Yang[1], Martin Gontscharow[1], Hanno Gottschalk[3], and J. Marius Zöllner[1,2]

[1] FZI Research Center for Information Technology, 76131 Karlsruhe, Germany
bogdoll@fzi.de
[2] Karlsruhe Institute of Technology, 76131 Karlsruhe, Germany
[3] Technical University of Berlin, 10623 Berlin, Germany

**Abstract.** The scale-up of autonomous vehicles depends heavily on their ability to deal with rare objects on the road. In order to handle such situations, it is necessary to detect anomalies in the first place. Anomaly detection has made great progress in the past years but suffers from poorly designed benchmarks with a strong focus on camera data. In this work, we present AnoVox, the largest benchmark for ANOmaly detection in autonomous driving to date. AnoVox incorporates multimodal sensor data and spatial VOXel ground truth, allowing for the comparison of methods independent of their used sensor. We propose a formal definition of normality and provide a compliant training dataset. AnoVox is the first benchmark to contain both *content* and *temporal* anomalies.

**Keywords:** Anomaly Detection · Autonomous Driving · Benchmark

## 1 Introduction

In the past years, we have seen autonomous vehicles scale from small Operational Design Domains (ODD) to city-wide areas, reaching geographic areas covering multiple neighboring cities in the near future [73–75]. However, with a growing fleet in an expanding ODD, scenarios from the long tail of rare events [71] occur more frequently as a consequence of increased exposure. Expert perspectives, focusing on content and temporal anomalies, are commonly used to judge "long tail" data points [10, 11, 17, 18, 40, 61, 66], as replaying training data during inference is not possible. Such expert views are often used for the creation of anomaly detection benchmarks but are of no help for anomaly detection. Few publications take a data-oriented perspective on the definition of anomalies [25, 39, 82], also known as out-of-distribution (OOD), corner cases, or outliers. Recent research has focused on anomaly detection methods [42] and its ramifications, such as OOD segmentation [58], object detection [31], instance segmentation [55], and

[*] These authors contributed equally

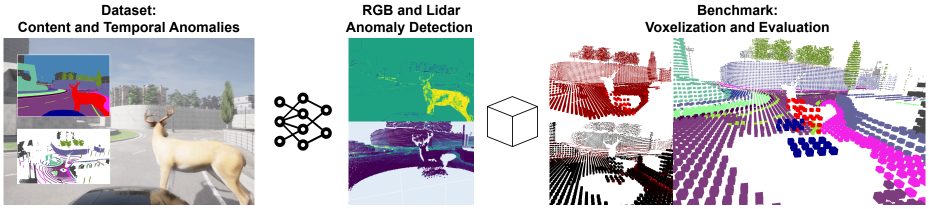

**Fig. 1: Overview of AnoVox.** AnoVox includes both *content* and *temporal* anomalies. Here, we see an example of a *content* anomaly. Ground truth is available for both camera and lidar, enabling the application of anomaly detection methods for both modalities. We additionally provide ground truth in a voxelized form. By mapping the anomaly scores to this voxelized space, we can compare methods from any sensor.

video tracking [54]. These methods typically do not learn a representation of normality from training data alone, but require a semantic segmentation stage and fine-tuning on auxiliary data containing (artificial) anomalies [9,23]. While there exists a considerable number of datasets in autonomous driving [14,49,53], only around 5 - 10% of datasets are designed for anomaly detection [13,15]. These existing datasets, however, have significant limitations. First, the majority are camera-only, despite autonomous driving relying on multi-modal sensor setups. Second, temporal information is often overlooked, limiting anomaly detection to single frames. Additionally, *content* anomalies are rarely found in challenging traffic situations, which allows for the assumption that anything on the road might be an anomaly [54]. Most anomalies are human-defined, such as dogs on the street [9], making detection challenging as these classes are typically included in training datasets but not annotated [27,31]. As a result, anomaly detection methods can miss anomalies they don't perceive as atypical due to their training.

With AnoVox, we offer a challenging and adaptable benchmark for ANOmaly detection, leveraging the CARLA simulation engine [30] to provide ground truth in all modalities and a spatial VOXel representation. Given the surge in foundation models [21,60], world models [45,80], and large-scale pre-training approaches [43,78], we see potential for the application of self- and unsupervised training paradigms in the field of anomaly detection as well. Hence, we prioritize support for label-free training paradigms and provide an extensive synthetic dataset to enable research in OOD segmentation, addressing the limitations of current datasets. All code is available on GitHub. The dataset can be found on Zenodo. Our contributions include:

1. Formal definition of normality and provision of a compliant training dataset for a fair comparison of anomaly detection methods
2. Large and challenging benchmark with a high scene complexity and temporal scenarios that include many frames without anomalies
3. First benchmark to provide content and temporal anomalies, adaptable multimodal sensor data, ego-vehicle state information, and ground truth in all modalities, as well as a spatial voxel representation

**Table 1:** Vision-based anomaly detection benchmarks. In the *Normality* column, $^\dagger$ denotes a domain shift between normal data and the proposed dataset, and $^{\circledast}$ denotes that the data with anomalies is based on a subset of the normal data. Adapted from [15].

| **Dataset** | **Data** | **Size** | **Ano.** | **Ano.** | **#Ano.** | **Ground** | **Temp.** | **Ego** | **Reg.** | **Normality** |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | **Source** | **Type** | **Classes** | **truth** | **Data** | **Act.** | **Tasks** | |
| **Fishyscapes** [8,9] | | | | | | | | | | |
| FS Lost and Found | Cam. | 375 | Recording | Content | 1 | Sem. mask (2D) | — | — | — | — |
| FS Static | Cam. | 1,030 | Data Augmentation | Content | 1 | Sem. mask (2D) | — | — | — | Cityscapes$^{\circledast}$ |
| **Crash to Not Crash** [46] | | | | | | | | | | |
| YouTubeCrash | Cam. | 2,400 | Web Sourcing | Temporal | 1 | Bbox (2D) | ✓ | — | ✓ | YouTubeCrash |
| GTACrash | Cam. | 154,400 | Simulation | Temporal | 1 | Bbox (2D) | ✓ | — | ✓ | **GTA V** |
| **CAOS** [41] | | | | | | | | | | |
| StreetHazards | Cam. | 1,500 | Simulation | Content | 1 | Sem. mask (2D) | ✓ | — | ✓ | CARLA$^\dagger$ |
| BDD-Anomaly | Cam. | 810 | Class Exclusion | Content | 3 | Sem. mask (2D) | — | — | ✓ | BDD100K$^{\circledast}$ |
| **SegmentMeIfYouCan** [23] | | | | | | | | | | |
| RoadAnomaly21 | Cam. | 110 | Web Sourcing | Content | 1 | Sem. mask (2D) | — | — | — | Cityscapes$^\dagger$ |
| RoadObstacle21 | Cam. | 412 | Recording | Content | 1 | Sem. mask (2D) | ✓ | — | — | Cityscapes$^\dagger$ |
| **Rare Road Objects** [19] | | | | | | | | | | |
| Synthetic Fire Hydrants | Cam. | 30,000 | Simulation | Content | 1 | Bbox (2D) | — | — | — | CARLA$^\dagger$ |
| Synthetic Crosswalks | Cam. | 20,000 | Simulation | Content | 1 | Bbox (2D) | — | — | — | CARLA$^\dagger$ |
| **CODA** [51] | | | | | | | | | | |
| CODA-KITTI | Cam., Lidar | 309 | Void Classes | Content | 6 | Bbox (2D) | — | — | — | KITTI$^{\circledast}$ |
| CODA-nuScenes | Cam., Lidar | 134 | Void Classes | Content | 17 | Bbox (2D) | — | — | — | nuScenes$^{\circledast}$ |
| CODA-ONCE | Cam., Lidar | 1,057 | OOD Detection | Content | 32 | Bbox (2D) | — | — | — | ONCE$^{\circledast}$ |
| CODA2022-ONCE | Cam., Lidar | 1,057 | OOD Detection | Content | 29 | Bbox (2D) | — | — | ✓ | ONCE$^{\circledast}$ |
| CODA2022-SODA10M | Cam. | 8,711 | OOD Detection | Content | 29 | Bbox (2D) | — | — | ✓ | SODA10M$^{\circledast}$ |
| **W-OOD Tracking** [54] | | | | | | | | | | |
| Street Obstacle Sequences | Cam., Depth | 1,129 | Recording | Content | 13 | Inst. mask (2D) | ✓ | — | — | Cityscapes$^\dagger$ |
| CARLA-WildLife | Cam., Depth | 1,210 | Simulation | Content | 18 | Inst. mask (2D) | ✓ | — | ✓ | **CARLA** |
| **Misc** | | | | | | | | | | |
| Lost and Found [62] | Stereo Cam. | 2,104 | Recording | Content | 42 | Sem. mask (2D) | ✓ | — | — | — |
| WD-Pascal [6] | Cam. | 70 | Data Augmentation | Content | 1 | Sem. mask (2D) | — | — | — | WildDash$^{\circledast}$ |
| Vistas-NP [36] | Cam. | 11,167 | Class Exclusion | Content | 4 | Sem. mask (2D) | — | — | ✓ | Mapillary Vistas$^{\circledast}$ |
| MUAD [32] | Cam., Depth | 4,641 | Sim., Class Exclusion | Content | 9 | Sem. mask (2D) | — | — | ✓ | MUAD |
| DeepAccident [72] | Cam., Lidar | 57,000 | Simulation | Temporal | 9 | Bbox (3D) | ✓ | — | ✓ | **CARLA** |
| **AnoVox (ours)** | **Cam., Lidar**<br>**Depth** | **245,600** | Simulation | **Content**<br>**Temporal** | **178** | **Inst. mask (2D,3D)**<br>**Voxel (3D)** | ✓ | ✓ | ✓ | **CARLA** |

## 2 Related Work

In autonomous driving, anomaly detection has been classically evaluated on a small set of benchmarks. For our analysis of related works, we have included works with published open-access perception datasets from an ego perspective that provide pixel- or point-wise ground truth. Thus, we neglected frameworks that do not provide explicit data [38, 65, 68, 76, 77], such with missing or incomplete data [34, 48, 52], and any that provide only frame-wise annotations [2, 67, 70, 79]. For our overview of related works as shown in Table 1, we updated and extended the survey of Bogdoll et al. [15].

It can be observed that most benchmarks are small and designed for camera-based *content* anomaly detection, providing ground truth in the form of semantic masks. Among the datasets including *content* anomalies, just the CODA [51] family includes lidar data but only provides ground truth in the form of 2D bounding boxes in the camera space. The DeepAccident [72] benchmark is the only one to provide *temporal* 3D labels for lidar point clouds.

There are different categories of how anomalies were introduced [15] in the datasets. **Recording** and **Simulation** are similar in the way that selected anomalies were directly introduced into the data. This way, the anomalies are truly part of the environment [19, 54, 62]. The definition of what counts as an anomaly can vary, though. **Data Augmentation** typically follows a copy-and-

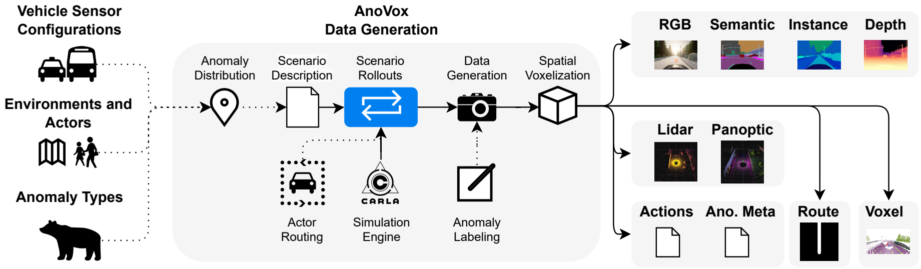

**Fig. 2: Overview:** With Anovox, highly configurable scenarios can be created that either represent *normality* or include *content* or *temporal* anomalies. Generated datasets include rich labels and ground truth in 2D, 3D, and a spatial voxel space.

paste pattern, where images of anomalies are pasted onto a scene from another dataset. This way, a distribution shift between the anomaly and the underlying data is introduced [6, 9]. **Web Sourcing** describes the process of manually curating images from the web that are deemed anomalous [23, 46]. **Class Exclusion** is based on existing datasets and removes selected classes from the training data, thus treating them as anomalous, while the classes themselves remain rather normal from a human point of view [35, 41]. **Void Classes** utilizes void or misc classes from existing datasets and labels them as anomalous. This can be done with additional labeling guidelines [51]. Finally, **OOD Detection** uses an anomaly detection method to derive anomaly proposals from a dataset which can then be labeled, typically after a human quality inspection [51].

While many datasets include temporal data in the form of sequences and provide labels for regular tasks, such as object detection or semantic segmentation, none includes state information about the ego vehicle. However, during deployment, such information is generally available and can be leveraged.

Most benchmarks do not provide a definition of normality [9, 23, 41]. This makes it especially hard for un- and self-supervised methods to precisely detect anomalies if the semantic training distribution is not fully known. Especially desirable is a well-defined normality that allows for the generation of compliant training data. This is only possible in simulation, where full control over both the training and evaluation data is available. Sometimes, even unlabeled anomalies occur in evaluation data [72].

AnoVox is a challenging benchmark that addresses anomaly detection from an embodied AI perspective rather than from a pure computer vision perspective. We provide data in a temporal context with state information about the ego vehicle for typical multimodal sensor setups. For a clear definition of what anomalies are, we provide a formalized definition of normality and compliant training data. This allows for the usage of un- and self-supervised anomaly detection methods, which often rely on large amounts of unlabeled data.

**Table 2:** A formal definition of normality and how the training data generated and provided by AnoVox aligns with the definition.

| Category | Description | AnoVox |
|---|---|---|
| **Ego** | | |
| Ego vehicle | Recording vehicle | Lincoln MKZ 2020 |
| Sensor config. | Sensor types, placements | Configurations Mono, Stereo, Multi, Surround |
| Ego behavior | Driving characteristics | Behavior Agent (actions) |
| **Domain** | | |
| Area | Geographical area | Towns 01,02,03,04,05,06,07,10HD |
| Environment | Weather, time of day | ClearNoon, CloudyNoon, WetNoon, WetCloudyNoon, HardRainNoon, SoftRainNoon, ClearSunset, CloudySunset, WetSunset, WetCloudySunset, MidRainSunset, HardRainSunset, SoftRainSunset |
| **Physical entities** | | |
| Traffic participants | Vehicles, VRUs | Vehicles, Walkers ∈ Blueprint library |
| Vehicle behavior | Driving characteristics | Traffic Manager Autopilot |
| Pedestrian behavior | Movement characteristics | AI Walker |

# 3 AnoVox Dataset

AnoVox provides both data that represents *normality* and data including *content* and *temporal* anomalies. While we do provide a large-scale dataset, AnoVox is primarily a scalable framework that can be used with arbitrary vehicle setups and a wide selection of parameters to create further data. This does not only allow for the detection of anomalies in known environments but also for the detection of anomalies under domain shifts. In Sec. 3.1, we provide a formalized definition of normality and demonstrate how the training data provided by AnoVox follows this definition of normality. In Sec. 3.2, we show what types of scenarios can be generated with AnoVox. Finally, Sec. 3.3 provides an overview of our dataset.

## 3.1 Definition of Normality

In the past, the definition of normality was often not extensively discussed when anomaly detection benchmarks were presented. A typical solution is to define normality as all semantic classes from the Cityscapes dataset [9, 23]. However, this is not necessarily related to the training data. And if it is, it requires labeled training data as normality to be aware of the classes.

With AnoVox, we provide full controllability about both normality and anomalies in synthetic environments. This ensures that anomalies included in the benchmark are true anomalies and are not included in an unlabeled training dataset by chance. For a fair benchmark of anomaly detection methods, it is important that they share the same definition of normality. We argue that this normality needs to be primarily defined by the training data rather than expert-defined concepts. This, however, requires the option to generate large amounts of training data following a definition of normality, which is infeasible in the real world, as anomalies would certainly occur in fleet-sized, unlabeled datasets. However, in the field of autonomous driving as a subfield of embodied AI, there is more to the training data than just frames: There is a recording entity that

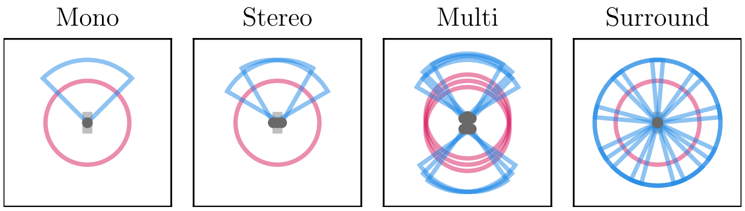

**Fig. 3:** Preconfigured sensor configurations in the AnoVox suite. Blue wedges visualize RGB cameras, while red circles visualize lidar sensors.

performs actions, and there is temporal context. Therefore, we provide a formal definition of normality based on three categories:

**Ego:** Domain shifts in data cannot only be induced by novel environmental conditions but also by different capturing methods. In autonomous driving, this especially refers to sensor types and configurations. In addition, temporal changes in the environment are heavily influenced by our own actions. Thus, also the behavior of the ego agent counts towards normality.

**Domain:** With the domain, we describe the static environment around the vehicle. This includes the geographical areas the vehicle has traversed, but also seen weather types and time of day specifications.

**Physical Entities:** These are the dynamic actors in the scene, most typically other vehicles, cyclists, and pedestrians. However, also categories such as animals or potentially moving objects can be included here.

Based on these concepts, we provide a definition of normality for the CARLA simulation engine in Table 2 that allows for the generation of compliant datasets.

### 3.2 Scenario Generation

AnoVox is designed to create configurable, large-scale datasets that either define normality or include anomalies. As shown in Fig. 2, first, the vehicle sensor configuration needs to be set. AnoVox currently supports an arbitrary number of camera, lidar, and depth sensors, which can be positioned freely on the ego vehicle of choice. This allows for the replication of existing sensor setups, the alignment with other datasets, or the testing of new configurations.

We provide four pre-designed multimodal vehicle sensor configurations. As shown in Fig. 3, the **mono** configuration consists of a lidar and a camera which are centered on top of the vehicle, centered forward. Our **stereo** setup consists of two cameras at the front edge and both a camera and lidar on top. The **multi** setup adds rear-facing cameras and additional lidar sensors at the front and rear, positioned at a lower level compared to the roof-mounted lidars. Finally, the **surround** setup provides a full 360° camera view next to a top-mounted lidar. Every RGB camera automatically comes with a depth camera.

Second, the environment and actors need to be set. AnoVox currently supports eight different regions and 14 weather and time of day presets. Pedestrians, cyclists, as well as multiple types of vehicles can be spawned in the scene.

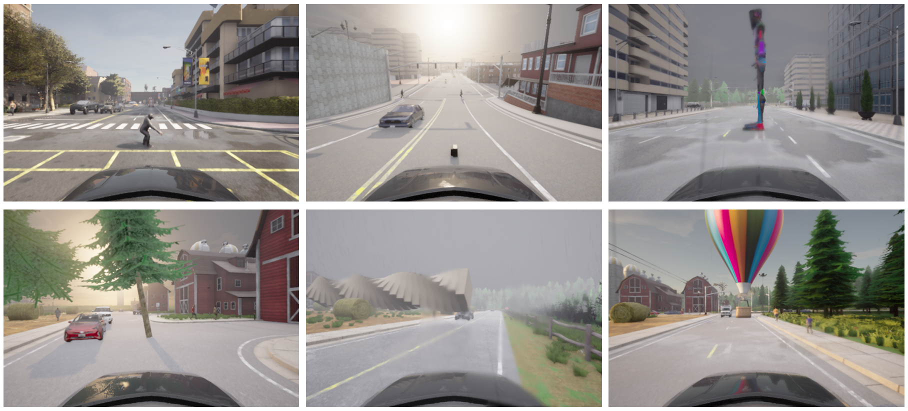

**Fig. 4: Content anomalies:**. Examples from our six categories: An ape as an animal anomaly, an old tv as a home anomaly, a statue as a special anomaly, a tree as a nature anomaly, a pillar as a falling anomaly, and a hot air balloon as an airplane anomaly.

Third, the type of anomaly needs to be defined. Currently, AnoVox supports to create *normality* without anomalies, the placement of *content* anomalies, and the activation of *temporal* anomalies. By removing domains from the training dataset, AnoVox also supports the detection of anomalies under domain shifts.

Given these configurations, AnoVox pre-computes the scenario flows and stores them as scenario descriptions. Thus, meta-data describing all scenarios is available and allows for effortless dataset analysis. Scenario descriptions can be shared among peers by sharing the scenario description files and locally generating the scenarios without the need to move hundreds of gigabytes of data.

Based on the given configuration, AnoVox executes the driving scenarios in simulation. We provide a custom-built CARLA simulation engine [30] including all *content* anomalies that we have manually collected and processed. Each scenario has a length of 20 seconds and is recorded at 10 Hz, resulting in 200 frames. When the scenario starts, the ego vehicle is spawned in the world and follows a given route to its target goal. At some point along the way, a *content* or *temporal* anomaly will appear. To guarantee that the anomaly is reached in time, a green wave is activated along the route of the ego vehicle. Since physics computations remain active in the simulation, the ego vehicle will make contact with the anomalies, which leads to realistic collisions.

**Actor Routing.** Strange objects on the road do the same thing in simulation as they would do in real life - they cause traffic jams. This makes it often unfeasible to reach the anomaly for the ego vehicle. Thus, we deploy a filter and rerouting algorithm for all other vehicles in our scenario. First, we filter all actors close to our spawn point and on our direct path toward the anomaly. Then, we continuously monitor all planned paths and reroute vehicles whenever they were to enter a lane with an anomaly on it, as they would get stuck. This rerouting technically

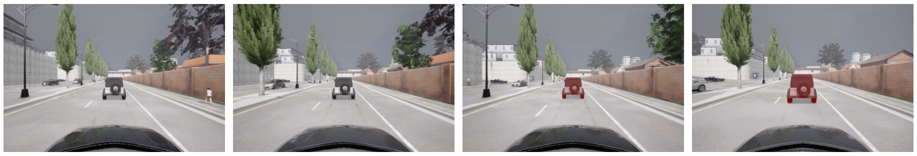

**Fig. 5: Temporal anomalies:** This scenario shows our implemented type of *temporal* anomalies. The first two images show the regular vehicle following mode. The third and fourth images show the active braking maneuver with overlayed ground truth.

changes their driving behavior. As shown in Table 2, in the training data, only vehicles are present, which show an *autopilot* driving behavior. Rerouting makes them switch into a *behavior agent*. We deal with this by providing novel labels, as further explained in Sec. 3.2. This way, false positives, which might occur due to a violation of the alignment with normality, can be filtered out for evaluation.

**Content Anomalies.** We provide 178 different *content* anomalies in five different size classes *tiny, small, medium, big*, and *huge*. Semantically, they hierarchically belong to six different super-classes. Every anomaly has an individual label next to its super-class and size label. The class **animal** includes 33 animals of different sizes. The category **home** includes 53 typical household items such as furniture, tables, backpacks, or cardboard boxes. The category **special** includes 67 objects of rather atypical types and such that fall into a misc category, such as Pokemon, some of which could appear in the real world in the form of costumes or cuddly toys. The class **nature** includes 12 outdoor objects, such as rocks or wood. In the category **falling**, we provide 9 large objects, such as novel trees, that were spawned in an unstable position, which made them fall over. Finally, the **airplane** class consists of four types of large flying objects.

As especially large anomalies sometimes require manual positioning, we use the classes *home, special,* and *animal* for our automated scenario generation. Here, we place all anomalies in critical positions along the way distributed around lane centers. For the classes *nature, falling,* and *airplane*, we provide a manually curated dataset in a geographic area with large free-space areas.

**Temporal Anomalies.** There are two ways to generate *temporal* anomalies: Optimization-based or knowledge-based [72]. While both are of value, for our benchmark, it is important to comply with the defined normality, which is more challenging with optimization-based methods. Thus, we provide knowledge-based *content* anomalies. As prior work has shown that such expert-defined temporal anomalies require high engineering efforts [2,11,12], we provide only a single type of *temporal* anomalies. We implement **sudden braking** scenarios of a lead vehicle, which is a very typical scenario in everyday traffic [46]. While we set a planned route for the ego vehicle in our *content* anomaly scenarios, we use the same route and follow it with a lead vehicle. Then, our ego vehicle follows that lead vehicle. Along the route, the lead vehicle will perform a sudden brake

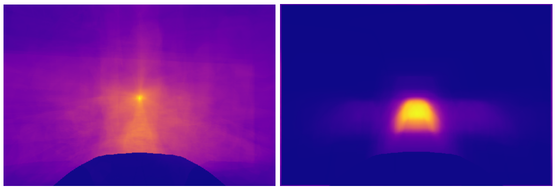

**Fig. 6:** Distribution of our *content* anomalies (left) and distribution of the *temporal* anomalies (right), based on the 2D semantic labels in the dataset.

with negative velocities which cannot be seen during training. While braking, we label the lead vehicle in the same way as we label *content* anomalies.

**Data Generation.** As shown in Fig. 2, we provide sensor data for all positioned RGB cameras, depth cameras, and lidar sensors. For regular perception tasks, we provide panoptic masks for both camera and lidar. For each frame, also information about the state of the ego vehicle is collected, such as its actions *throttle, street, and brake.* For approaches that require additional information about the planned route [3, 16, 44, 81], we provide a standard-format Bird's-eye view (BEV) representation of the planned route.

For the anomalous instances, we provide meta-data, such as their positions and their size. Ground truth is embedded into the semantic masks for both camera and lidar. As we designed the benchmark to compare methods that use different modalities for the detection of anomalies, we represent all anomalies also in a 3D voxel grid with customizable grid size. Based on our depth maps and lidar point clouds, we fuse all visible points from all sensors in 3D and quantize them into the voxel grid. In addition, the task of detecting anomalies in 3D is especially helpful for downstream tasks, such as motion planning.

By using the CARLA simulation engine, we support the largest simulation ecosystem in autonomous driving. However, it has inherent flaws, such as imperfect behavior agents [47]. Thus, we perform minor data cleaning to remove scenarios that contain collisions with pedestrians in the evaluation data and scenarios that show high deceleration values for other vehicles in the training data. This ensures that we avoid unlabeled anomalies in the evaluation data and that our temporal anomalies are distinct from all behaviors seen during training.

**Labeling.** AnoVox provides 40 label classes, as shown in Fig. 7. We provide labels for standard tasks as used by Cityscapes and CARLA [27,30]. Additionally, we label the ego vehicle if visible. As described in Sec. 3.2, some vehicles in the scene might switch from autopilot to a behavior agent. As this driving behavior is not present in training data, we provide additional labels for all vehicle classes while controlled by a behavior agent. Finally, we provide labels for all *content* and *temporal* anomalies. Labels for the super-classes can be found in the semantic labels, while fine-granular, individual anomaly labels are provided in additional

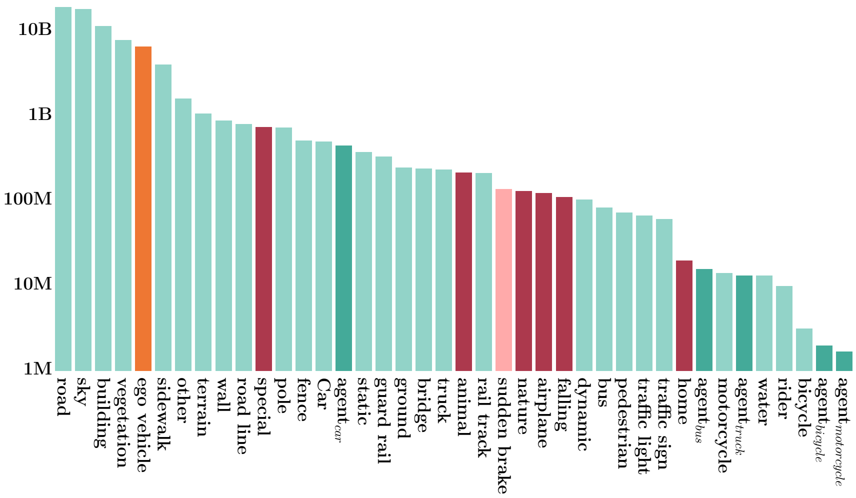

**Fig. 7: Number of pixels per class:** Bright green represents standard classes, and dark green ones additional classes for vehicles that switched into an agent behavior mode. The orange class shows our ego vehicle, and all red classes represent anomalies.

metadata. Finally, voxels are provided with an anomaly label when the closest point to their center is anomalous.

### 3.3 Dataset and Statistics

Next to all code to generate further datasets, we provide over 1.3 TB of generated data. We have split the data into chunks smaller than 50 GB for easier processing, which we call instantiations. We provide 1,117 scenarios based on the *mono* sensor configuration, 76 for the *stereo* configuration, and 35 for the *multi* configuration. While the *stereo* and *multi* configurations are more of a symbolic nature, our dataset for the *mono* sensor configuration is extensive. We provide data for eight different areas, and for each, we record *normality* training data as well as evaluation data with *content* and *temporal* anomalies, resulting in 24 different types of scenarios. Additionally, for each of those 24, we create many scenarios where settings such as the weather, time of day, or spawned anomaly vary. For *content* anomalies, 14.8% of all frames contain visible anomalies in camera data and 74.8% in lidar data. For *temporal* anomalies, 15.5% of all frames include anomalies equally visible in both sensor modalities. As shown in Fig. 6 anomalies are well distributed over the visible space but with a focus on critical areas in front of the ego vehicle.

Finally, Fig. 7 gives an overview of the number of pixels per class label in our dataset. We support all basic classes and provide additional classes for our ego vehicle and other traffic participants that have switched from autopilot to

**Table 3:** A comparison of current state-of-the-art methods for anomaly detection.

| | **Year** | **Sensor** | **Aux** |
|---|---|---|---|
| 3DUIS [59] | 2022 | LiDAR | ✗ |
| APF [50] | 2023 | LiDAR | ✗ |
| cDNP [33] | 2023 | Camera | ✗ |
| EAM [37] | 2023 | Camera | ✓ |
| ElC-OIS [28] | 2023 | LiDAR | ✗ |
| LS-VOS [63] | 2023 | LiDAR | ✗ |
| Mask2Anomaly [64] | 2023 | Camera | ✓ |
| Maskomaly [1] | 2023 | Camera | ✗ |
| Maximized Entropy [24] | 2021 | Camera | ✓ |
| Najibi et al. [56] | 2022 | LiDAR | ✗ |
| ObsNet [5] | 2021 | Camera | ✗ |
| RbA [57] | 2023 | Camera | Both |
| ReaL [22] | 2022 | LiDAR | Both |
| SynBoost [29] | 2021 | Camera | ✓ |

an agent behavior, so they can be excluded from evaluation if needed, as they behave slightly differently than the ones available in the training data.

# 4 AnoVox Benchmark Suite

We provide a full evaluation suite for camera and lidar-based anomaly detection methods for content anomalies. Based on pixel- or pointwise anomaly scores, we voxelize the results and compare them against the voxelized anomaly ground truth of AnoVox. We evaluate on voxel grid of size $100m \times 100m \times 64m$, where each voxel has a length of 0.5 m. We have performed ablation studies on smaller voxel sizes, but have not found a significant impact on the results.

## 4.1 Anomaly Detection

To get first insights into how current state-of-the-art methods for *content* anomalies will perform on our benchmark, we have implemented two candidate methods, one based on camera data and one based on lidar data. For their selection, we have analyzed existing surveys and benchmark results [9,13,15,23]. However, as there is no existing benchmark for lidar-based detection method, this was more challenging. We performed an extensive search, also focusing on open-set segmentation, as this is a similar but more active field. Our final candidates can be found in Table 3. To emphasize the definition of normality based on training data alone, we then neglected all methods that require auxiliary data with anomalies during training. While we are also interested in methods purely trained on raw data, we allowed the usage of semantic labels during training, as the state-of-the-art heavily relies on those. We then picked the best-performing method that provided open-source code for implementations. For camera data,

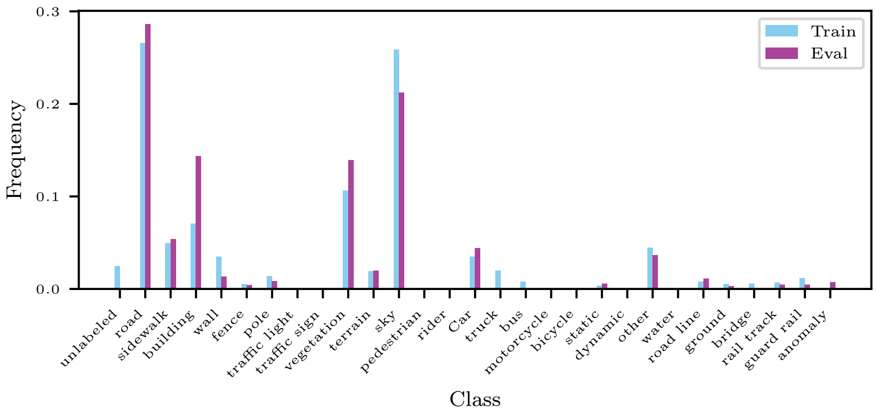

**Fig. 8:** Class distributions for the train and evaluation datasets.

we selected the Rejected By All (RbA) method from Nayal et al. [57], and for lidar data we selected the Redundancy classifier (REAL) framework by Cen et al. [22]. While both methods allow for the usage of auxiliary data during training, we did not fine-tune on anomalies. For RbA this means we did not perform "Outlier Data Exposure" [57], and for REAL this means we only used the "predictive distribution calibration" without "unknown object synthesis" [22].

The **Rejected by All** model proposed by Nayal et al. uses camera data and is based on the Mask2Former model [26]. The authors propose that object queries are specialized on single classes, such that outliers can be detected when the input is rejected by all queries. RbA was trained on Cityscapes and evaluated on SegmentMeIfYouCan anomaly and obstacle tracks, as well as on the Fishyscapes Lost&Found track, where they perform well with $\text{FPR}_{95}$ values below 12.

The **Redundancy classifier** approach from Cen et al. uses lidar data and is based on the Cylinder3D [83] framework. To learn high anomaly scores, the authors introduce a new loss, where the second-highest predictions per point are assigned to an unknown class, which they use for uncertainty prediction during inference while also performing close-set semantic segmentation. RbA was trained and evaluated on both SemanticKITTI [4] and Nuscenes [20]. The approach works reasonably well with an AUPR of 10.0 and an AUROC of 77.5. However, due to the domain gap, these values cannot easily be compared with camera-based methods.

**Training** For a fair comparison, we have trained both methods on the same training dataset. For this, we have created a small *normality* instantiation in the size of the Cityscapes dataset that consists of 2,975 frames with temporal scenarios. Contrary to the setting of REAL, where unknown objects are included in the training set but ignored for the loss computation, our training dataset does

**Table 4:** Evaluation of REAL and RbA on AnoVox.

| | **AUPR** ↑ | **AUROC**↑ | **F1**↑ | **PPV**↑ | $\mathbf{FPR}_{95}$↓ |
|---|---|---|---|---|---|
| REAL | 0.14 | 43.30 | 0.0 | 0.0 | 100 |
| $\text{REAL}_{+norm}$ | 0.04 | 43.53 | 0.0 | 0.0 | 100 |
| $\text{REAL}_{big}$ | 0.17 | 44.7 | 0.0 | 0.0 | 100 |
| $\text{REAL}_{medium}$ | 0.11 | 42.8 | 0.0 | 0.0 | 100 |
| $\text{REAL}_{small}$ | 0.21 | 55.1 | 0.0 | 0.0 | 100 |
| RbA | 0.7 | 57.3 | 2.6 | 1.4 | 100 |
| $\text{RbA}_{+norm}$ | 0.2 | 57.6 | 2.6 | 0.4 | 100 |
| $\text{RbA}_{big}$ | **2.3** | 54.9 | **3.7** | **2.7** | 100 |
| $\text{RbA}_{medium}$ | 0.6 | **60.5** | 0.0 | 0.0 | 100 |
| $\text{RbA}_{small}$ | 0.01 | 53.2 | 0.0 | 0.0 | 100 |

not include any anomalies. We followed the standard training procedures and parameters as provided by the authors.

**Evaluation** For the evaluation, we created another small instantiation with *content* anomalies. The class distribution between the training and evaluation datasets can be found in Figure 8. To evaluate both methods, their anomaly scores need to be mapped to voxel space first. While this is straightforward for lidar data, we lifted the anomaly scores from RbA into 3D using ground truth depth data. Due to this evaluation in voxel space, it must be noted that the class imbalance between normal data and anomalies is much larger than in the sensor space due to quantization effects. Thus, results cannot be compared to reported values from the state of the Art. However, we have also performed evaluations on the sensor data directly, confirming higher scores.

As shown in Table 4, both methods had significant issues detecting anomalies. While we observed stable training performance and saw improved performance when trained on larger datasets, the results were rather surprising. RbA was able to detect some anomalies, as shown in Fig. 1, but struggled with most. REAL on the other hand, while generating well-performing closed-set predictions, was unable to generate any meaningful uncertainties in the absence of unknown objects in the training data, as exemplarily shown in Fig. 9.

This shows that the task of anomaly detection becomes much harder on a challenging benchmark such as AnoVox, where anomalies are not defined by not belonging to Cityscapes classes, but are defined by being absent from the training set, as most state-of-the-art methods currently utilize deviations based on semantic segmentations to derive anomalies. In addition, there is no domain shift for the anomaly classes, as they are rendered in the same simulation engine from which the training data stems from. Finally, lots of frames without visible anomalies and particularly small anomalies raise the bar additionally. This setting, combined with the induced class imbalance due to our quantization loss during voxelization, makes it much harder to perform well on AnoVox compared to existing benchmarks, which we hope will benefit the community greatly.

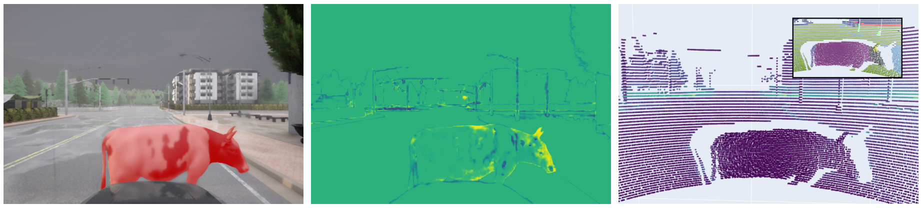

**Fig. 9:** AnoVox scene with a cow as an animal *content* anomaly (left). The RbA (middle) anomaly detection method is only able to detect some border parts as anomalous, while REAL (right) assigns the same uncertainty to the cow as it does to the ground. In the accompanying closed-set detection, it is mostly classified as a car.

## 5 Conclusion

We presented the AnoVox benchmark for **ANO**maly detection in autonomous driving. AnoVox provides ground truth in the form of semantic masks for RGB images and lidar point clouds as well as spatial **VOX**els.

We analyzed the shortcomings of existing benchmarks, most notably no clear definition of normality and a strong focus on RGB data. AnoVox is the first and largest benchmark that provides a concise formalization of normality, accompanied by compliant training data. We provided first insights on how well current SotA models perform on this new paradigm of anomaly detection benchmarking. We see a great need for anomaly detection methods to learn normality based on training data instead of detecting deviations from known Cityscape classes.

**Limitations.** At the moment, each voxel contains only a single class. This can lead to cases where small anomalies are not represented in single frames of voxelized data. For our evaluation, we map camera-based anomaly detections into 3D by using ground truth depth. We have performed experiments using ZoeDepth [7], a state-of-the-art metric depth estimation model, to map these predictions into 3D, but the overlap between the resulting voxels and the ground truth voxels was too large. Here, more sophisticated methods are necessary. We limited ourselves to only a single temporal anomaly, more types could be implemented. Finally, while simulation is necessary to have full control over both training and evaluation, this leads to a Sim-2-Real gap.

**Outlook.** We plan to implement multi-class voxel in a future release to reduce losses during quantization. The Sim-2-Real gap could be tackled by style transfers, as proposed by Tian et al. [69]. We would like to include radar data in a future release, but the current implementation in CARLA is too naive to be helpful. Finally, we plan to release more data with our *surround* sensor setup.

## Acknowledgment

This work results from the just better DATA project supported by the German Federal Ministry for Economic Affairs and Climate Action (BMWK), grant number 19A23003H.

## References

1. Ackermann, J., Sakaridis, C., Yu, F.: Maskomaly: Zero-Shot Mask Anomaly Segmentation. In: BMVC (2023)
2. Aliakbarian, M.S., Saleh, F.S., Salzmann, M., Fernando, B., Petersson, L., Andersson, L.: Viena2: A driving anticipation dataset. In: ACCV (2018)
3. Bansal, M., Krizhevsky, A., Ogale, A.: Chauffeurnet: Learning to drive by imitating the best and synthesizing the worst. In: RSS (2019)
4. Behley, J., Garbade, M., Milioto, A., Quenzel, J., Behnke, S., Stachniss, C., Gall, J.: SemanticKITTI: A dataset for semantic scene understanding of LiDAR sequences. In: ICCV (2019)
5. Besnier, V., Bursuc, A., Picard, D., Briot, A.: Triggering failures: Out-of-distribution detection by learning from local adversarial attacks in semantic segmentation. In: ICCV (2021)
6. Bevandić, P., Krešo, I., Oršić, M., Šegvić, S.: Simultaneous Semantic Segmentation and Outlier Detection in Presence of Domain Shift. In: GCPR (2019)
7. Bhat, S.F., Birkl, R., Wofk, D., Wonka, P., Müller, M.: ZoeDepth: Zero-shot transfer by combining relative and metric depth. arXiv:2302.12288 (2023)
8. Blum, H., Sarlin, P.E., Nieto, J., Siegwart, R., Cadena, C.: Fishyscapes: A Benchmark for Safe Semantic Segmentation in Autonomous Driving. In: ICCVW (2019)
9. Blum, H., Sarlin, P.E., Nieto, J., Siegwart, R., Cadena, C.: The Fishyscapes Benchmark: Measuring Blind Spots in Semantic Segmentation. IJCV **129** (2021)
10. Bogdoll, D., Breitenstein, J., Heidecker, F., Bieshaar, M., Sick, B., Fingscheidt, T., Zöllner, J.M.: Description of Corner Cases in Automated Driving: Goals and Challenges. In: ICCVW (2021)
11. Bogdoll, D., Guneshka, S., Zöllner, J.M.: One Ontology to Rule Them All: Corner Case Scenarios for Autonomous Driving. In: ECCVW (2022)
12. Bogdoll, D., Nekolla, M., Joseph, T., Zöllner, J.M.: Quantification of Actual Road User Behavior on the Basis of Given Traffic Rules. In: IEEE IV (2022)
13. Bogdoll, D., Nitsche, M., Zöllner, J.M.: Anomaly Detection in Autonomous Driving: A Survey. In: CVPRW (2022)
14. Bogdoll, D., Schreyer, F., Zöllner, J.M.: AD-Datasets: A Meta-Collection of Data Sets for Autonomous Driving. In: VEHITS (2022)
15. Bogdoll, D., Uhlemeyer, S., Kowol, K., Zöllner, J.M.: Perception Datasets for Anomaly Detection in Autonomous Driving: A Survey. In: IEEE IV (2023)
16. Bogdoll, D., Yang, Y., Zöllner, J.M.: MUVO: A Multimodal Generative World Model for Autonomous Driving with Geometric Representations. arXiv:2311.11762 (2023)
17. Breitenstein, J., Termohlen, J.A., Lipinski, D., Fingscheidt, T.: Corner Cases for Visual Perception in Automated Driving: Some Guidance on Detection Approaches. arXiv:2102.05897 (2021)
18. Breitenstein, J., Termöhlen, J.A., Lipinski, D., Fingscheidt, T.: Systematization of Corner Cases for Visual Perception in Automated Driving. In: IEEE IV (2020)
19. Bu, T., Zhang, X., Mertz, C., Dolan, J.M.: CARLA Simulated Data for Rare Road Object Detection. In: IEEE ITSC (2021)
20. 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: CVPR (2020)
21. Caron, M., Touvron, H., Misra, I., Jégou, H., Mairal, J., Bojanowski, P., Joulin, A.: Emerging properties in self-supervised vision transformers. In: ICCV (2021)

22. Cen, J., Yun, P., Zhang, S., Cai, J., Luan, D., Wang, M.Y., Liu, M., Tang, M.: Open-world semantic segmentation for LIDAR point clouds. In: ECCV (2022)
23. Chan, R., Lis, K., Uhlemeyer, S., Blum, H., Honari, S., Siegwart, R., Fua, P., Salzmann, M., Rottmann, M.: SegmentMeIfYouCan: A Benchmark for Anomaly Segmentation. In: NeurIPS (2021)
24. Chan, R., Rottmann, M., Gottschalk, H.: Entropy maximization and meta classification for out-of-distribution detection in semantic segmentation. In: ICCV (2021)
25. Chan, R., Uhlemeyer, S., Rottmann, M., Gottschalk, H.: Deep Neural Networks and Data for Automated Driving, chap. Detecting and Learning the Unknown in Semantic Segmentation. Springer (2022)
26. Cheng, B., Misra, I., Schwing, A.G., Kirillov, A., Girdhar, R.: Masked-attention mask transformer for universal image segmentation. In: CVPR (2022)
27. Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., Schiele, B.: The cityscapes dataset for semantic urban scene understanding. In: CVPR (2016)
28. Deng, W., Huang, K., Yu, Q., Lu, H., Zheng, Z., Chen, X.: ElC-OIS: Ellipsoidal Clustering for Open-World Instance Segmentation on LiDAR Data. In: IROS (2023)
29. Di Biase, G., Blum, H., Siegwart, R., Cadena, C.: Pixel-wise anomaly detection in complex driving scenes. In: CVPR (2021)
30. Dosovitskiy, A., Ros, G., Codevilla, F., López, A.M., Koltun, V.: Carla: An open urban driving simulator. In: CoRL (2017)
31. Du, X., Wang, X., Gozum, G., Li, Y.: Unknown-Aware Object Detection: Learning What You Don't Know from Videos in the Wild. In: CVPR (2022)
32. Franchi, G., Yu, X., Bursuc, A., Tena, A., Kazmierczak, R., Dubuisson, S., Aldea, E., Filliat, D.: Muad: Multiple uncertainties for autonomous driving benchmark for multiple uncertainty types and tasks. In: BMVC (2022)
33. Galesso, S., Argus, M., Brox, T.: Far away in the deep space: Dense nearest-neighbor-based out-of-distribution detection. In: ICCVW (2023)
34. Gong, L., Zhang, Y., Xia, Y., Zhang, Y., Ji, J.: SDAC: A Multimodal Synthetic Dataset for Anomaly and Corner Case Detection in Autonomous Driving. AAAI **38**(3) (2024)
35. Grcić, M., Bevandić, P., Kalafatić, Z., Šegvić, S.: Dense out-of-distribution detection by robust learning on synthetic negative data. Sensors **24** (2024)
36. Grcić., M., Bevandić., P., Segvić., S.: Dense Open-set Recognition with Synthetic Outliers Generated by Real NVP. In: Int. Conf. Comp. Vis., Imag. Comp. Graph. Theo. Appl. (2021)
37. Grcić, M., Šarić, J., Šegvić, S.: On advantages of mask-level recognition for outlier-aware segmentation. In: CVPRW (2023)
38. Hanselmann, N., Renz, K., Chitta, K., Bhattacharyya, A., Geiger, A.: KING: Generating Safety-Critical Driving Scenarios for Robust Imitation via Kinematics Gradients. In: ECCV (2022)
39. Heidecker, F., Bieshaar, M., Sick, B.: Corner cases in machine learning processes. AI Perspectives & Advances **6**(1) (2024)
40. Heidecker, F., Breitenstein, J., Rösch, K., Löhdefink, J., Bieshaar, M., Stiller, C., Fingscheidt, T., Sick, B.: An Application-Driven Conceptualization of Corner Cases for Perception in Highly Automated Driving. In: IEEE IV (2021)
41. Hendrycks, D., Basart, S., Mazeika, M., Zou, A., Kwon, J., Mostajabi, M., Steinhardt, J., Song, D.: Scaling Out-of-Distribution Detection for Real-World Settings. In: ICML (2022)

42. Hendrycks, D., Gimpel, K.: A Baseline for Detecting Misclassified and Out-of-Distribution Examples in Neural Networks. In: ICLR (2016)
43. Hess, G., Jaxing, J., Svensson, E., Hagerman, D., Petersson, C., Svensson, L.: Masked autoencoder for self-supervised pre-training on lidar point clouds. In: IEEE Winter Conf. on Appl. of Comp. Vis. (2023)
44. Hu, A., Corrado, G., Griffiths, N., Murez, Z., Gurau, C., Yeo, H., Kendall, A., Cipolla, R., Shotton, J.: Model-based imitation learning for urban driving. In: NeurIPS (2022)
45. Hu, A., Russell, L., Yeo, H., Murez, Z., Fedoseev, G., Kendall, A., Shotton, J., Corrado, G.: GAIA-1: A Generative World Model for Autonomous Driving. arXiv:2309.17080 (2023)
46. Kim, H., Lee, K., Hwang, G., Suh, C.: Crash to Not Crash: Learn to Identify Dangerous Vehicles Using a Simulator. AAAI **33** (2019)
47. Kim, S.: Multiple issues in the behavior agent. `https://github.com/carla-simulator/carla/issues/5398` (2022), accessed: 2024-03-07
48. Kösel, M., Schreiber, M., Ulrich, M., Gläser, C., Dietmayer, K.: Revisiting Out-of-Distribution Detection in LiDAR-based 3D Object Detection. arXiv:2404.15879 (2024)
49. Li, H., Li, Y., Wang, H., Zeng, J., Xu, H., Cai, P., Chen, L., Yan, J., Xu, F., Xiong, L., Wang, J., Zhu, F., Yan, K., Xu, C., Wang, T., Xia, F., Mu, B., Peng, Z., Lin, D., Qiao, Y.: Open-sourced data ecosystem in autonomous driving: the present and future. arXiv:2312.03408 (2024)
50. Li, J., Dong, Q.: Open-Set Semantic Segmentation for Point Clouds via Adversarial Prototype Framework. In: CVPR (2023)
51. Li, K., Chen, K., Wang, H., Hong, L., Ye, C., Han, J., Chen, Y., Zhang, W., Xu, C., Yeung, D.Y., Liang, X., Li, Z., Xu, H.: CODA: A Real-World Road Corner Case Dataset for Object Detection in Autonomous Driving. In: ECCV (2022)
52. Li, Y., Zhang, W., Chen, K., Liu, Y., Li, P., Gao, R., Hong, L., Tian, M., Zhao, X., Li, Z., Yeung, D.Y., Lu, H., Jia, X.: Automated Evaluation of Large Vision-Language Models on Self-driving Corner Cases. arXiv:2404.10595 (2024)
53. Liu, M., Yurtsever, E., Zhou, X., Fossaert, J., Cui, Y., Zagar, B.L., Knoll, A.C.: A survey on autonomous driving datasets: Data statistic, annotation, and outlook. arXiv:2401.01454 (2024)
54. Maag, K., Chan, R., Uhlemeyer, S., Kowol, K., Gottschalk, H.: Two Video Data Sets for Tracking and Retrieval of Out of Distribution Objects. In: ACCV (2023)
55. Mohan, R., Kumaraswamy, K., Hurtado, J.V., Petek, K., Valada, A.: Panoptic out-of-distribution segmentation. arXiv:2310.11797 (2023)
56. Najibi, M., Ji, J., Zhou, Y., Qi, C.R., Yan, X., Ettinger, S., Anguelov, D.: Motion inspired unsupervised perception and prediction in autonomous driving. In: ECCV (2022)
57. Nayal, N., Yavuz, M., Henriques, J.F., Güney, F.: RbA: Segmenting unknown regions rejected by all. In: ICCV (2023)
58. Nunes, I., Laranjeira, C., Oliveira, H., dos Santos, J.A.: A systematic review on open-set segmentation. Computers & Graphics **115** (2023)
59. Nunes, L., Chen, X., Marcuzzi, R., Osep, A., Leal-Taixé, L., Stachniss, C., Behley, J.: Unsupervised Class-Agnostic Instance Segmentation of 3D LiDAR Data for Autonomous Vehicles. IEEE Rob. and Auto. Letters **7**(4) (2022)
60. Oquab, M., Darcet, T., Moutakanni, T., Vo, H.V., Szafraniec, M., Khalidov, V., Fernandez, P., Haziza, D., Massa, F., El-Nouby, A., Howes, R., Huang, P.Y., Xu, H., Sharma, V., Li, S.W., Galuba, W., Rabbat, M., Assran, M., Ballas, N., Synnaeve, G., Misra, I., Jegou, H., Mairal, J., Labatut, P., Joulin, A., Bojanowski, P.:

DINOv2: Learning Robust Visual Features without Supervision. Trans. on Mach. Learn. Reas. (2023)

61. Pfeil, J., Wieland, J., Michalke, T., Theissler, A.: On Why the System Makes the Corner Case: AI-based Holistic Anomaly Detection for Autonomous Driving. In: IEEE IV (2022)
62. Pinggera, P., Ramos, S., Gehrig, S., Franke, U., Rother, C., Mester, R.: Lost and Found: Detecting Small Road Hazards for Self-Driving Vehicles. In: IROS (2016)
63. Piroli, A., Dallabetta, V., Kopp, J., Walessa, M., Meissner, D., Dietmayer, K.: LS-VOS: Identifying outliers in 3d object detections using latent space virtual outlier synthesis. IEEE ITSC (2023)
64. Rai, S.N., Cermelli, F., Fontanel, D., Masone, C., Caputo, B.: Unmasking anomalies in road-scene segmentation. In: ICCV (2023)
65. Rempe, D., Philion, J., Guibas, L.J., Fidler, S., Litany, O.: Generating Useful Accident-Prone Driving Scenarios via a Learned Traffic Prior. In: CVPR (2022)
66. Rösch, K., Heidecker, F., Truetsch, J., Kowol, K., Schicktanz, C., Bieshaar, M., Sick, B., Stiller, C.: Space, Time, and Interaction: A Taxonomy of Corner Cases in Trajectory Datasets for Automated Driving. In: Symp. Series Comp. Intell. (2022)
67. Sakaridis, C., Dai, D., Van Gool, L.: ACDC: The Adverse Conditions Dataset with Correspondences for Semantic Driving Scene Understanding. In: ICCV (2021)
68. Sena Ferreira, R., Guérin, J., Guiochet, J., Waeselynck, H.: SiMOOD: Evolutionary Testing Simulation with Out-Of-Distribution Images. In: Pac. Rim Int. Symp Dep. Comp. (2022)
69. Tian, B., ang Gao, H., Cui, L., Zheng, Y., Luo, L., Wang, B., Zhi, R., Zhou, G., Zhao, H.: Latency-aware Road Anomaly Segmentation in Videos: A Photorealistic Dataset and New Metrics. arXiv:2401.04942 (2024)
70. Tremblay, M., Halder, S.S., de Charette, R., Lalonde, J.F.: Rain Rendering for Evaluating and Improving Robustness to Bad Weather. IJCV **129** (2021)
71. Urtasun, R.: Interpretable neural motion planner. https://youtube.com/watch?v=PSZ2Px9PrHg (2021), accessed: 2024-03-01
72. Wang, T., Kim, S., Ji, W., Xie, E., Ge, C., Chen, J., Li, Z., Luo, P.: DeepAccident: A Motion and Accident Prediction Benchmark for V2X Autonomous Driving. In: AAAI (2024)
73. Waymo: Waymo One: The next step on our self-driving journey. https://waymo.com/blog/2018/10/where-next-10-million-miles-will-take-us/ (2018), accessed: 2024-02-22
74. Waymo: Advice Letter 0002 (Tier 2). https://cpuc.ca.gov/-/media/cpuc-website/divisions/consumer-protection-and-enforcement-division/documents/tlab/av-programs/waymo-llc-cpuc-advice-letter-0002-tier-2--january-2024-passenger-safety-plan-update-january-192024.pdf (2024), accessed: 2024-02-22
75. Waymo: Waymo One. https://waymo.com/waymo-one/ (2024), accessed: 2024-02-22
76. Wiederer, J., Schmidt, J., Kressel, U., Dietmayer, K., Belagiannis, V.: A benchmark for unsupervised anomaly detection in multi-agent trajectories. In: IEEE ITSC (2022)
77. Xu, S., Gilpin, L.H.: DANGER: A Framework of Danger-Aware Novel Dataset Generator Extension for Robustness Test of Machine Learning. In: BayLearn Mach. Learn. Symp. (2022)
78. Yang, H., Zhang, S., Huang, D., Wu, X., Zhu, H., He, T., Tang, S., Zhao, H., Qiu, Q., Lin, B., He, X., Ouyang, W.: Unipad: A universal pre-training paradigm for autonomous driving. In: CVPR (2024)

79. Zendel, O., Honauer, K., Murschitz, M., Steininger, D., Dominguez, G.F.: WildDash - Creating Hazard-Aware Benchmarks. In: ECCV (2018)
80. Zhang, L., Xiong, Y., Yang, Z., Casas, S., Hu, R., Urtasun, R.: Learning Unsupervised World Models for Autonomous Driving via Discrete Diffusion. In: ICLR (2024)
81. Zhang, Z., Liniger, A., Dai, D., Yu, F., Van Gool, L.: End-to-end urban driving by imitating a reinforcement learning coach. In: ICCV (2021)
82. Zhou, J., Beyerer, J.: Corner cases in data-driven automated driving: Definitions, properties and solutions. In: IEEE IV (2023)
83. Zhu, X., Zhou, H., Wang, T., Hong, F., Ma, Y., Li, W., Li, H., Lin, D.: Cylindrical and asymmetrical 3d convolution networks for LiDAR segmentation. In: CVPR (2021)