DiverSim: A Customizable Simulation Tool to Generate Diverse Vulnerable Road User Datasets

Jon Ander Iñiguez de Gordoa[1,2][a], Martín Hormaetxea[1][b], Marcos Nieto[1][c], Gorka Vélez[1][d] and Andoni Mujika[2][e]

[1]*Vicomtech Foundation, Basque Research and Technology Alliance (BRTA), Mikeletegi 57, 20009, Donostia-San Sebastián, Spain*

[2]*University of the Basque Country (UPV/EHU), Donostia-San Sebastián, Spain*

{jainiguez, mhormaetxea, mnieto, gvelez}@vicomtech.org, andoni.mujika@ehu.eus

Keywords: Synthetic Data, Simulation, Unreal Engine, Diversity, Mobility Aids, Fisheye, ADAS.

Abstract: This work presents DiverSim, a highly customizable simulation tool designed for the generation of diverse synthetic datasets of vulnerable road users to address key challenges in pedestrian detection for Advanced Driver Assistance Systems (ADAS). Although recent Deep Learning models have advanced pedestrian detection, their performance still depends on the diversity and inclusivity of training data. DiverSim, developed on Unreal Engine 5, allows users to control various environmental conditions and pedestrian characteristics, including age, gender, ethnicity and mobility aids. The tool features a highly customizable virtual fisheye camera and a Python API for easy configuration and automated data annotation in the ASAM OpenLABEL format. Our experiments demonstrate DiverSim's capability to evaluate pedestrian detection models across diverse user profiles, revealing potential biases in current state-of-the-art models. By making both the simulator and Python API open source, DiverSim aims to contribute to fairer and more effective AI solutions in the field of transportation safety.

1 INTRODUCTION

The accurate detection and tracking of pedestrians play a critical role in Advanced Driver Assistance Systems (ADAS) and Autonomous Driving (AD) technologies in order to prevent collisions and ensure the safety of Vulnerable Road Users (VRU). As Deep Learning has gained prominence in visual processing tasks, integrating neural networks into perception functions within autonomous vehicles has become standard practice.

Artificial Intelligence (AI) and Deep Learning-based approaches depend heavily on the datasets used to train the algorithms. However, creating diverse and representative datasets remains a challenge, as real-world data is often biased or lacks the necessary variations to train robust AI systems (Buolamwini and Gebru, 2018). Training machine learning algorithms with biased datasets can lead to algorithmic discrimination (Bolukbasi et al., 2016).

Moreover, the US National Institute of Standards and Technology (NIST) highlighted a tendency to prioritize the availability of datasets over their relevance or suitability (Schwartz et al., 2022). Consequently, the data used in AI training often diverge from real-world scenarios, leading to underrepresentation and exclusion of certain societal groups (Shahbazi et al., 2023).

In this work, we introduce DiverSim, a flexible and photorealistic simulation tool designed for the generation of diverse, synthetic pedestrian data. Built on Unreal Engine 5, DiverSim enables researchers to simulate a wide range of environmental and pedestrian conditions. It aims to support research in areas like Computer Vision, fairness, and bias mitigation by providing rich, annotated datasets that can be adapted to different use cases.

Our main contributions are:

- A highly customizable simulation environment, built on Unreal Engine 5, designed to generate synthetic and diverse pedestrian data with ground truth annotations. DiverSim enables extensive

[a] https://orcid.org/0000-0002-9008-5620
[b] https://orcid.org/0009-0000-6124-236X
[c] https://orcid.org/0000-0002-9008-5620
[d] https://orcid.org/0000-0002-8367-2413
[e] https://orcid.org/0000-0003-0998-3886

Iñiguez de Gordoa, J. A., Hormaetxea, M., Nieto, M., Vélez, G. and Mujika, A.
DiverSim: A Customizable Simulation Tool to Generate Diverse Vulnerable Road User Datasets.
DOI: 10.5220/0013201600003941

In *Proceedings of the 11th International Conference on Vehicle Technology and Intelligent Transport Systems (VEHITS 2025)*, pages 17-24
ISBN: 978-989-758-745-0; ISSN: 2184-495X

Figure 1: Fisheye camera view from the DiverSim simulator, showing pedestrians with various mobility needs.

customization options such as weather, time of day, pedestrian density and pedestrian features (e.g., physical characteristics or mobility needs). These settings can be easily adjusted through a JSON configuration file, enabling users to adapt simulations to their specific needs.

- A virtual fisheye camera within the simulator, a feature often missing in other simulators despite its prevalence in ADAS. Users can fully customize the fisheye camera parameters in order to generate images according to their specifications and simulate the camera capture of a fisheye camera on board of a vehicle.
- A Python API (application programming interface) to control the configuration of the simulation settings, camera parameters, and the initialization, data recording and annotation of the simulations. This API simplifies interaction with the simulation environment, making it more accessible for users.
- Both the Python API and the executable of the DiverSim simulator have been published to the benefit of the research community[1]. The licenses of all the assets employed in DiverSim are compatible with Artificial Intelligence applications, which means that the data generated with this simulator can be legitimately used for training, validation and testing of AI models.

The remainder of the paper is organized as follows: Section 2 shows related work; Section 3 presents the DiverSim simulating tool; Section 4 showcases the experiments to evaluate a state-of-the-art pedestrian detection model using synthetic data from the DiverSim simulator, focusing on different mobility aids in order to find potential detection biases; finally, in Section 5 the conclusions and future work are presented.

2 RELATED WORK

2.1 Diverse Real-World Pedestrian Datasets

While there are numerous open-source datasets available for ADAS applications, most have been recorded in regions with predominantly Caucasian populations, such as Europe (Geiger et al., 2013; Cordts et al., 2015; Geyer et al., 2020; Maddern et al., 2017), and North America (Sun et al., 2020; Caesar et al., 2020; Yu et al., 2020; Wilson et al., 2021). Moreover, most pedestrians in these datasets do not present mobility needs, and those who use mobility aids (e.g., wheelchairs, walkers, or canes) are often underrepresented, which limits the overall diversity of the data.

There have been various attempts to generate datasets that show greater diversity in the representation of pedestrians, addressing the limitations of existing datasets in portraying a broader range of demographics. For instance, the Database of Human Attributes (HAT) (Sharma and Jurie, 2011) specifically included people of different ages. Other datasets have focused on achieving gender balance (Linder et al., 2015). The *MIAP* subset (Schumann et al., 2021) of the Open Images Dataset (Kuznetsova et al., 2020) introduces more inclusive annotations for people (including attributes such as perceived gender and age), with a focus on fairness analysis. Furthermore, there have been multiple attempts to generate datasets that contain people with different mobility aids such as wheelchairs or crutches (Vasquez et al., 2017; Mohr et al., 2023; Yang et al., 2022).

2.2 Synthetic Datasets and Simulators

Despite the efforts to generate more diverse pedestrian datasets, it is extremely difficult, if not unfeasible, to obtain a real-world dataset that is fully balanced across gender, age, ethnicity, and mobility needs. Moreover, Deep Learning-based detection and tracking methods require large datasets, and privacy concerns in data acquisition and the manual effort needed for accurate annotation pose significant challenges. Synthetic environments offer an effective solution enabling controlled conditions and automated annotation through scenario-based data generation (de Gordoa et al., 2023).

For instance, CARLA simulator (Dosovitskiy et al., 2017) features a diverse range of pedestrian

[1] https://github.com/Vicomtech/DiverSim

blueprints representing various skin tones and ethnic backgrounds. Synthetic pedestrian datasets created with CARLA have demonstrated success across various tasks (Fabbri et al., 2021; Calle et al., 2024). However, the pedestrians in the official CARLA releases show little diversity in terms of mobility needs, and attributes such as gender or ethnicity are not labeled. As a result, the only feasible approach to create a balanced dataset is to randomize the pedestrian blueprints and hope for a reasonably balanced outcome.

Recent initiatives try to address some of these challenges by providing an accessibility-centered development environment. For instance, The X-World simulation module (Zhang et al., 2021), integrated into CARLA, enables the generation of agents with diverse mobility aids, although its current version only supports wheelchairs and walking canes. Other studies focus on utilizing Digital Twins to introduce pedestrians with different disabilities within diverse urban scenarios (Luna-Romero et al., 2024). These innovative approaches show the potential of synthetic datasets and simulators to meet the diversity needs of autonomous systems.

3 DIVERSIM SIMULATOR: AN OVERVIEW

DiverSim enables the generation of datasets with diverse pedestrians in different urban scenarios. By generating synthetic environments that realistically portray various pedestrian characteristics (such as age, gender, ethnicity, and mobility aids), DiverSim addresses the critical need for inclusive and representative training and evaluating data in AI models. This section provides an overview of DiverSim's key components and contributions, including the virtual environment created in Unreal Engine 5, configuration options, and a user-friendly Python API for data generation and annotation.

3.1 Virtual Environment

The simulator is designed to represent vulnerable road users in various urban scenarios, such as crossing at a zebra crossing, navigating through parking lots (as shown in Figure 2), and being picked up by vehicles, all within a city-like environment. In each simulation run, the scene is populated with pedestrians who appear and disappear at off-camera locations, creating a dynamic flow of individuals walking or crossing streets. This aims to replicate real-world pedestrian flow in urban environments.

Figure 2: Example scenario of vulnerable road users navigating the virtual parking lot.

To ensure diversity in the generated synthetic data, factors such as pedestrians' gender, race, and age are balanced by default, while varying environmental conditions, including lighting, building appearances, weather, and vehicle types, are randomized. Pedestrians are modeled with various mobility aids (such as wheelchairs, white canes, crutches, walking sticks, or no aid at all) to reflect a broad range of mobility needs and accessibility considerations within the environment.

While some assets and animations used in the simulation have been developed in-house, those sourced externally have been carefully selected to ensure that their licenses are compatible with Artificial Intelligence applications. The pedestrian models are sourced from CARLA, while animations come from a variety of sources: the crutches animation was created using Microsoft Kinect motion capture; the white cane animation was coded directly; the wheelchair animation was custom-developed in-house; and the walking and cane animations were purchased from the Unreal Engine Marketplace (Unreal Engine Marketplace, 2024b; Unreal Engine Marketplace, 2024a).

Moreover, DiverSim leverages AirSim (Shah et al., 2018) as a plugin to interface with the environment, enabling the introduction of virtual cameras and the extraction of ground truth data. The AirSim source code has been slightly adapted and modified to facilitate detailed ground truth annotation of all pedestrians and their attributes through the Python API presented in Section 3.3.

3.2 Configuration of the Simulation

DiverSim offers a degree of customization by allowing users to modify certain simulation parameters through an external configuration file. This file, written in JSON format, is processed each time the simulation is run, making it easy to edit either with the provided Python API (explained in Section 3.3) or other preferred tools.

The configuration file governs various parameters that influence the simulation environment, such as weather conditions, lighting, time of day, and vehicle density. Additionally, users can adjust the proportion of different mobility need categories within the simulation, enabling, for example, a higher representation of wheelchair users compared to white cane users, or the exclusion of a particular mobility aid from the scene entirely. It is worth noting that whether the classes should be balanced, reflect real-world proportions or prioritize minority classes depends on the specific objectives of the training or validation process. To support different approaches, we enable users to adjust the proportion of each category, allowing them to customize their simulation according to their preferred strategy.

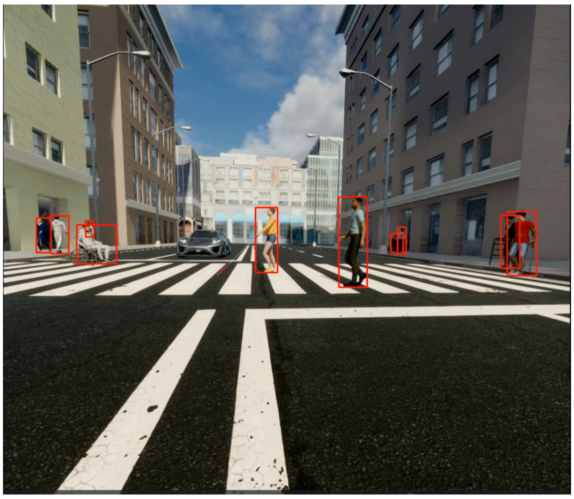

Figure 3: Example of a pinhole camera image with ground truth 2D bounding boxes for pedestrians.

3.3 Python API

A Python API has been developed in order to facilitate the data generation and annotation process to users. This API leverages the AirSim and VCD libraries (Nieto et al., 2021) to retrieve information from the synthetic Unreal Engine environment and annotate it in ASAM OpenLABEL format, respectively.

This API allows users to select several recording parameters such as its length, frames per second, camera or vehicle trajectory during the simulation, and the path in which the dataset will be saved. It also allows to update and modify several fields in the simulation settings file presented in Section 3.2 (such as the weather, light condition or urban scenario). The simulation can be initialized and stopped using this API, therefore, a large and diverse dataset generation can be planned by randomizing settings and initializing, recording and stopping the simulation iteratively.

Many simulators, such as AirSim or CARLA, provide pinhole camera sensor models only, and offer limited options to add distortion. In contrast, fisheye images, such as the one shown in Figure 1, are also generated in DiverSim using this Python API. In this simulator, six pinhole cameras are introduced by default to the vehicle setup. For each frame, these six cameras generate a cubemap image. Users can specify the fisheye camera parameters with the API, which then automatically maps the pixels of the cubemap image to the fisheye image. This process produces the image that would be captured by the described fisheye camera model. By default, images from both the user-defined fisheye camera and the front-facing pinhole camera of the cubemap are saved in the dataset.

3.4 Automated Data Annotation

The Python API presented in Section 3.3 annotates the generated data in a JSON file using the ASAM OpenLABEL standard. Each simulation or recording produces a unique JSON file that include details about the simulation context (weather and time of the day), streams (camera intrinsic parameters) and coordinate systems (e.g., camera extrinsics). Most importantly, it provides detailed annotations of the pedestrians within the generated images:

- Pedestrians are tracked as distinct objects across frames, rather than being generated as independent objects per frame.
- For each frame, 2D bounding boxes are annotated for the pedestrians. Each bounding box contains the name of its corresponding camera (fisheye or pinhole). If a pedestrian appears in multiple cameras in the same frame, a separate bounding box is annotated for each camera. These annotations are based on the ground truth instance segmentation of the images. Although there is no minimum visibility or bounding box size threshold in the default ground truth annotations, the simulator's open-source API can be easily modified to introduce such thresholds or conditions as needed.
- Gender, ethnicity, age and mobility needs are annotated as attributes to the generic *pedestrian* class. Although these attributes may not be essential for certain applications, they can prove valuable for assessing fairness or addressing bias concerns.
- Pedestrian models were labeled as either male or female, based on their human-perceived gender

presentation. We opted not to include a non-binary label to the pedestrian gender attribute, as gender identity should be determined by individuals themselves (Scheuerman et al., 2020). The same approach was used for determining attributes such as ethnicity and age group of the pedestrian models.

- The mobility needs attribute is determined by the mobility aid *object* (e.g., wheelchair, walking cane, crutches, or none) carried by the pedestrian.

4 EXPERIMENTS

To demonstrate the potential of DiverSim in assessing the generalization capabilities of AI models across people with different characteristics, we tested a state-of-the-art pedestrian detection model using synthetic data generated by this simulator. The goal was to evaluate the performance of the model in terms of inclusivity for diverse pedestrian traits, particularly focusing on individuals using different mobility aids. Although there is an essential domain gap between the training data and the synthetic data employed for testing, the results presented in Section 4.2 suggest that the analysed model might have some negative detection bias against pedestrians using specific mobility aids.

4.1 Experiment Setup

For this experiment, we tested the YOLOv5 architecture (Jocher et al., 2020), specifically the YOLOv5s model pre-trained on the COCO dataset (Lin et al., 2014).

The synthetic dataset used for testing was generated with the DiverSim simulating tool. The scenes took place at a crosswalk from the perspective of the frontal camera of a stationary vehicle, capturing several pedestrians as they passed. We ensured a balanced representation of pedestrians across different ethnicities, mobility aids and genders. The atmospheric conditions were set to daylight and sunny weather. Overall, 1,000 images were generated for testing.

For this evaluation, we chose pinhole images instead of fisheye images due to the specific challenges posed by the radial distortion of fisheye lenses for bounding box detection in this setup (Rashed et al., 2021). However, fisheye images remain valuable in other experimental contexts within this work, as they provide a wider field of view and capture more complex spatial information.

4.2 Results

Table 1 shows the performance metrics obtained by the selected pedestrian detector model on the generated synthetic dataset. It includes the precision, recall and mean average precision (mAP) calculated at the intersection over union (IoU) thresholds of 0.50 and 0.75, and the mAP averaged over IoU threshold from 0.50 to 0.95 ($\text{mAP}_{50\text{-}95}$). These metrics are widely used in the evaluation of object detection models (Padilla et al., 2020).

Table 2 analyses the $\text{mAP}_{50\text{-}95}$ metric across varying sizes of ground truth bounding boxes. The ground truth bounding boxes are annotated using semantic segmentation cameras, allowing even mostly occluded pedestrians to be annotated in the dataset. The smaller bounding boxes (likely representing pedestrians that are very distant or just entering the camera's field of view) would probably not be annotated in a manual labeling process. As shown in the table, the pedestrian detector exhibits much lower performance metrics with the smaller bounding boxes.

We also aimed to assess the performance of the evaluated model across the mobility aids used by pedestrians. Since the evaluated YOLOv5 model classifies all pedestrians under the same category, we calculated the recall for each subcategory by considering the true positives (correctly detected pedestrians) and false positives (undetected pedestrians) for each mobility aid:

$$Recall(aid) = \frac{TP_{aid}}{TP_{aid} + FN_{aid}}, \tag{1}$$

where TP_{aid} represents the true positives or correctly detected pedestrians with a specific mobility aid, and FN_{aid} represents the false negatives or undetected pedestrians with the corresponding mobility aid.

Considering the results obtained in Table 2, and in order to ensure more representative results, we decided to exclude the smaller ground truth bounding boxes from the calculation of these recall metrics, as these might not accurately reflect the detection capabilities for pedestrians with mobility aids.

Table 3 presents the recall metrics obtained for each mobility aid category. This table shows that the model obtains a similar recall across most mobility aids, except for wheelchair users, where the recall value drops to 0.513. This decline suggests that the analysed YOLOv5 model may have biases in its detection capabilities, probably due to limited training data representing wheelchair users, as well as the differences in height and posture between wheelchair users and walking pedestrians, which could make detection more challenging.

Table 1: Performance metrics of the analysed model in the synthetically generated pedestrian dataset.

Model	**Dataset**	**Precision**	**Recall**	**mAP$_{50}$**	**mAP$_{75}$**	**mAP$_{50\text{-}95}$**
YOLOv5	COCO	0.405	0.564	0.405	0.294	0.269

Table 2: mAP$_{50\text{-}95}$ metric for different ground truth bounding box sizes.

Bbox size	**Area (pixels)**	**mAP$_{50\text{-}95}$**
Small	$< 32 \times 32$	0.058
Medium	$[32 \times 32, 92 \times 92]$	0.254
Large	$> 92 \times 92$	0.309
Overall		**0.269**

Table 3: Recall performance of the pedestrian detection model across different pedestrian mobility aids, evaluated at an IoU threshold of 0.5.

Mobility Aid	**Recall**
None	0.709
White cane	0.742
Walking stick	0.688
Crutches	0.725
Wheelchair	0.513
Overall	**0.655**

5 CONCLUSIONS

We presented a highly configurable simulation tool that allows to generate diverse pedestrian datasets that represent people with different characteristics in terms of mobility aids, ethnicity or gender. Section 4 demonstrated the ability of the DiverSim simulator to assess potential biases of AI pedestrian detection models. However, the applications of this simulator can extend beyond evaluation, as the generated synthetic data can also be used to train or fine-tune new models.

As future work, we plan to enhance DiverSim by incorporating 3D information to the generated data, including the generation of point clouds from LIDAR sensors and the annotation of ground truth 3D bounding boxes. Additionally, we aim to increase the variety of pedestrian assets in the simulation to achieve greater diversity (by including a range of clothing styles and representations of different cultures, such as veiled women), and introduce more urban scenarios beyond the existing crosswalk and parking environments in the current release.

Furthermore, we plan to explore how DiverSim can be employed to train pedestrian detection models, enhancing their robustness and promoting fairer detection performance across various user profiles, including individuals with mobility aids and from diverse demographic groups.

ACKNOWLEDGEMENTS

This work has received funding from the European Union's Horizon Europe research and innovation programme under grant agreement number 101076868 (AWARE2ALL project).

REFERENCES

Bolukbasi, T., Chang, K.-W., Zou, J. Y., Saligrama, V., and Kalai, A. T. (2016). Man is to computer programmer as woman is to homemaker? debiasing word embeddings. *Advances in neural information processing systems*, 29.

Buolamwini, J. and Gebru, T. (2018). Gender shades: Intersectional accuracy disparities in commercial gender classification. In Friedler, S. A. and Wilson, C., editors, *Proceedings of the 1st Conference on Fairness, Accountability and Transparency*, volume 81 of *Proceedings of Machine Learning Research*, pages 77–91. PMLR.

Caesar, H., Bankiti, V., Lang, A. H., Vora, S., Liong, V. E., Xu, Q., Krishnan, A., Pan, Y., Baldan, G., and Beijbom, O. (2020). nuscenes: A multimodal dataset for autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11621–11631.

Calle, J., Unzueta, L., Leskovsky, P., and García, J. (2024). Learning domain-invariant spatio-temporal visual cues for video-based crowd panic detection. In *Paradigms on Technology Development for Security Practitioners*, pages 297–310. Springer.

Cordts, M., Omran, M., Ramos, S., Scharwächter, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., and Schiele, B. (2015). The cityscapes dataset. In *CVPR Workshop on the Future of Datasets in Vision*, volume 2, page 1.

de Gordoa, J. A. I., García, S., Urbieta, I., Aranjuelo, N., Nieto, M., de Eribe, D. O., et al. (2023). Scenario-based validation of automated train systems using a 3d virtual railway environment. In *2023 IEEE 26th International Conference on Intelligent Transportation Systems (ITSC)*, pages 5072–5077. IEEE.

Dosovitskiy, A., Ros, G., Codevilla, F., Lopez, A., and Koltun, V. (2017). Carla: An open urban driving simulator. In *Conference on robot learning*, pages 1–16. PMLR.

Fabbri, M., Brasó, G., Maugeri, G., Cetintas, O., Gasparini, R., Ošep, A., Calderara, S., Leal-Taixé, L., and Cucchiara, R. (2021). Motsynth: How can synthetic data help pedestrian detection and tracking? In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10849–10859.

Geiger, A., Lenz, P., Stiller, C., and Urtasun, R. (2013). Vision meets robotics: The kitti dataset. *The International Journal of Robotics Research*, 32(11):1231–1237.

Geyer, J., Kassahun, Y., Mahmudi, M., Ricou, X., Durgesh, R., Chung, A. S., Hauswald, L., Pham, V. H., Mühlegg, M., Dorn, S., et al. (2020). A2d2: Audi autonomous driving dataset. *arXiv preprint arXiv:2004.06320*.

Jocher, G., Stoken, A., Borovec, J., Changyu, L., Hogan, A., Diaconu, L., Ingham, F., Poznanski, J., Fang, J., Yu, L., et al. (2020). ultralytics/yolov5: v3. 1-bug fixes and performance improvements. *Zenodo*.

Kuznetsova, A., Rom, H., Alldrin, N., Uijlings, J., Krasin, I., Pont-Tuset, J., Kamali, S., Popov, S., Malloci, M., Kolesnikov, A., et al. (2020). The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale. *International journal of computer vision*, 128(7):1956–1981.

Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., and Zitnick, C. L. (2014). Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, pages 740–755. Springer.

Linder, T., Wehner, S., and Arras, K. O. (2015). Real-time full-body human gender recognition in (rgb)-d data. *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3039–3045.

Luna-Romero, S. F., Stempniak, C. R., de Souza, M. A., and Reynoso-Meza, G. (2024). Urban digital twins for synthetic data of individuals with mobility aids in curitiba, brazil, to drive highly accurate ai models for inclusivity. In Salgado-Guerrero, J. P., Vega-Carrillo, H. R., García-Fernández, G., and Robles-Bykbaev, V., editors, *Systems, Smart Technologies and Innovation for Society*, pages 116–125, Cham. Springer Nature Switzerland.

Maddern, W., Pascoe, G., Linegar, C., and Newman, P. (2017). 1 year, 1000 km: The oxford robotcar dataset. *The International Journal of Robotics Research*, 36(1):3–15.

Mohr, L., Kirillova, N., Possegger, H., and Bischof, H. (2023). A comprehensive crossroad camera dataset of mobility aid users. In *34th British Machine Vision Conference: BMVC 2023*. The British Machine Vision Association.

Nieto, M., Senderos, O., and Otaegui, O. (2021). Boosting ai applications: Labeling format for complex datasets. *SoftwareX*, 13:100653.

Padilla, R., Netto, S. L., and Da Silva, E. A. (2020). A survey on performance metrics for object-detection algorithms. In *2020 international conference on systems, signals and image processing (IWSSIP)*, pages 237–242. IEEE.

Rashed, H., Mohamed, E., Sistu, G., Kumar, V. R., Eising, C., El-Sallab, A., and Yogamani, S. (2021). Generalized object detection on fisheye cameras for autonomous driving: Dataset, representations and baseline. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2272–2280.

Scheuerman, M. K., Spiel, K., Haimson, O. L., Hamidi, F., and Branham, S. M. (2020). Hci guidelines for gender equity and inclusivity.

Schumann, C., Ricco, S., Prabhu, U., Ferrari, V., and Pantofaru, C. (2021). A step toward more inclusive people annotations for fairness. In *Proceedings of the 2021 AAAI/ACM Conference on AI, Ethics, and Society*, pages 916–925.

Schwartz, R., Schwartz, R., Vassilev, A., Greene, K., Perine, L., Burt, A., and Hall, P. (2022). *Towards a standard for identifying and managing bias in artificial intelligence*, volume 3. US Department of Commerce, National Institute of Standards and Technology.

Shah, S., Dey, D., Lovett, C., and Kapoor, A. (2018). Airsim: High-fidelity visual and physical simulation for autonomous vehicles. In *Field and Service Robotics: Results of the 11th International Conference*, pages 621–635. Springer.

Shahbazi, N., Lin, Y., Asudeh, A., and Jagadish, H. (2023). Representation bias in data: A survey on identification and resolution techniques. *ACM Computing Surveys*, 55(13s):1–39.

Sharma, G. and Jurie, F. (2011). Learning discriminative spatial representation for image classification. In *BMVC 2011-British Machine Vision Conference*, pages 1–11. BMVA Press.

Sun, P., Kretzschmar, H., Dotiwalla, X., Chouard, A., Patnaik, V., Tsui, P., Guo, J., Zhou, Y., Chai, Y., Caine, B., et al. (2020). Scalability in perception for autonomous driving: Waymo open dataset. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2446–2454.

Unreal Engine Marketplace (2024a). Old Man Animset. https://www.fab.com/listings/8fcce9be-d727-44f1-9261-56cfa8ef41e4. Accessed: 2024-06-01.

Unreal Engine Marketplace (2024b). Run and Walk. https://www.fab.com/es-es/listings/6f5351b5-b6c9-4e00-a248-8158e6a7c067. Accessed: 2025-01-07.

Vasquez, A., Kollmitz, M., Eitel, A., and Burgard, W. (2017). Deep detection of people and their mobility aids for a hospital robot. In *Proc. of the IEEE Eur. Conf. on Mobile Robotics (ECMR)*.

Wilson, B., Qi, W., Agarwal, T., Lambert, J., Singh, J., Khandelwal, S., Pan, B., Kumar, R., Hartnett, A., Pontes, J. K., Ramanan, D., Carr, P., and Hays, J. (2021). Argoverse 2: Next generation datasets for self-driving perception and forecasting. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks (NeurIPS Datasets and Benchmarks 2021)*.

Yang, H. F., Ling, Y., Kopca, C., Ricord, S., and Wang, Y. (2022). Cooperative traffic signal assistance system for non-motorized users and disabilities empowered by computer vision and edge artificial intelligence. *Transportation Research Part C: Emerging Technologies*, 145:103896.

Yu, F., Chen, H., Wang, X., Xian, W., Chen, Y., Liu, F., Madhavan, V., and Darrell, T. (2020). Bdd100k: A diverse driving dataset for heterogeneous multitask learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Zhang, J., Zheng, M., Boyd, M., and Ohn-Bar, E. (2021). X-world: Accessibility, vision, and autonomy meet. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9762–9771.