OUTBACK: A Multimodal Synthetic Dataset for Rural Australian Off-road Robot Navigation

Liyana Wijayathunga
School of Engineering
Edith Cowan University
Perth, Australia
lwijayat@our.ecu.edu.au

Dulitha Dabare
School of Engineering
Edith Cowan University
Perth, Australia
hewaged@our.ecu.edu.au

Alexander Rassau
School of Engineering
Edith Cowan University
Perth, Australia
a.rassau@ecu.edu.au

Douglas Chai
School of Engineering
Edith Cowan University
Perth, Australia
d.chai@ecu.edu.au

Syed Mohammed Shamsul Islam
School of Science
Edith Cowan University
Perth, Australia
syed.islam@ecu.edu.au

***Abstract*— *One of the most important aspects of robot scene understanding is semantic segmentation of external environments. Urban environment semantic segmentation has been extensively investigated by researchers and many real-world and synthetic datasets have been utilised to develop highly accurate segmentation results. However, the number of off-road datasets available for robot navigation research remains limited. To address this, this paper introduces a novel framework to generate varied photorealistic synthetic off-road datasets capable of supporting multiple sensor modalities. To evaluate this approach, a synthetic multimodal dataset for off-road ground robot navigation in typical Western Australian outback conditions has been created. The robot simulations for synthetic dataset generation were conducted using the NVIDIA Isaac Sim robotics simulator platform and the dataset consists of camera, LiDAR, and IMU sensor data of two synthetic off-road environment models, including 18 semantic classes. Experiments on the dataset were performed using three state-of-the-art image segmentation models. These experiments examine the effectiveness of urban semantic segmentation deep learning models on rural off-road driving scenarios and additionally evaluate the Sim2Real transfer of an off-road semantic segmentation model pretrained on our synthetic dataset.*

Keywords*— *synthetic dataset, off-road, semantic segmentation, robot navigation

I. Introduction

The area of autonomous navigation in unstructured outdoor environments has gained much interest in recent years, leading to multiple advancements in works related to simulation, models and datasets in tasks such as semantic segmentation, depth perception, and object detection [1, 2]. The task of semantic segmentation for identifying navigable terrain in autonomous navigation requires datasets that accurately represent the target environments the agent is deployed in to guarantee robust and safe navigation. This is especially vital in navigating off-road unstructured environments for use in areas such as disaster recovery, agriculture, environment surveying and mining. Navigating such terrain requires the autonomous agent to identify the navigable paths by differentiating between rough, smooth or untraversable terrain such as gravel, dirt, mud, and puddles [3]. Additional complexities arise from the wide variation in off-road terrain types across difference geographical areas. For example, forests, deserts, fields and farms each present unique navigational challenges.

Development of new models to tackle these challenges require datasets that represent the varied environments faced in off-road unstructured navigation scenarios [4]. While there are multiple works [2, 5, 6] that tackle the dataset problem for urban navigation scenarios, the works covering off-road datasets are sparse and only provide data collected in forested paths and farms in North-American and European geographic areas. The use of synthetic datasets to complement real-world datasets has gained traction in the research community as synthetic datasets can help overcome certain challenges associated with collecting real-world data. First, the challenge of collecting suitably varied data with sufficient variation in weather, time-of-day and geographical combinations, which can be time consuming and expensive. Second, the annotation quality of the data can often be rife with errors. With recent advancements in computational power, ray-tracing technologies and the wide availability of 3D engines such as Unreal and Unity, the quality of detail and lighting has improved to the point that synthetic scenes and the point-cloud data obtainable from these 3D environments offer a viable alternative to gathering real-world off-road data for autonomous navigation model training and simulation, while circumventing the challenges outlined above.

To that end, we propose a novel framework for generating off-road robotic sensor data and high-fidelity stereo and monocular RGB images with corresponding annotations using synthetic 3D environments. We also present the OUTBACK synthetic multimodal dataset, comprising of annotated stereo and monocular RGB images, LiDAR point cloud and Inertial Measurement Unit (IMU) sensor data. The high-fidelity off-road 3D environments used in our proposed framework were designed using the Unreal Engine 5 software package. The off-road ground robotic simulations were performed on the NVIDIA Isaac Sim platform [7] which was used to generate synthetic sensor data and image annotations (for stereo and monocular RGB images).The compiled multimodal dataset was developed in aid of research work in off-road unstructured terrain robotic navigation. We also provide performance comparison of multiple state-of-the-art semantic segmentation models trained on our dataset and evaluated on a real-world dataset to highlight the adaptability of the OUTBACK synthetic dataset to real-world tasks. This paper's contributions are as follows:

979-8-3503-7903-7/24/$31.00 ©2024 IEEE
DOI 10.1109/DICTA63115.2024.00065

1. We present a framework for obtaining annotated stereo and monocular RGB images, 3D LiDAR point-clouds and IMU data from Unreal Engine 5-based 3D environments using cameras and sensors mounted on simulated wheeled robots.
2. We introduce the OUTBACK synthetic dataset, comprising 4086 pairs of pixel-wise annotated stereo and monocular RGB image sets, 3D LiDAR point-clouds and IMU data. The data corresponds to two 3D models of the Western Australian landscape in sparse and forested daylight with 18 classes that commonly occur in the given environment.
3. We compare and evaluate the performance of state-of-the-art models in both structured and unstructured outdoor environments in the task of 2D semantic segmentation and Sim2Real transfer using our synthetic dataset.

The rest of the paper is organised as follows. Section 2 summarises the related literature. Section 3 describes the methodology adopted for curating the dataset. Section 4 notes the baseline experiments conducted, and Section 5 reports the results. Finally, Section 6 highlights the features of the proposed dataset and indicates potential future research directions.

II. Related Work

The seminal work in autonomous navigation datasets is the Cityscapes dataset by Cordts et al. [2] representing 19 classes found in urban navigation scenarios. Another prominent dataset is BDD100K which comprises 10, 000 annotated images for semantic segmentation representing urban navigation scenarios with different weather conditions [5]. Both of the above datasets contain annotated image data for use in object detection and semantic segmentation tasks, while the KITTI dataset offers a multimodal dataset combining RGB camera data with LiDAR point clouds in urban scenarios [8].

Most of the datasets available for the task of semantic segmentation in autonomous navigation are focused on urban on-road navigation scenarios. However, datasets such as RUGD [9], RELLIS-3D [4] and GOOSE [10] provide autonomous navigation data for off-road environments, which can be used to train robots on tasks such as semantic segmentation of images and point clouds. The RUGD dataset offers a set of annotated images for frames taken from a wheeled mobile robot mounted camera but does not offer 3D point-cloud data or IMU data to be used in further research work. The RELLIS-3D dataset offers annotated data for both images and point-clouds, collected from the paths near the Rellis campus in Texas. The dataset represents densely vegetated areas and paths native to that region. The GOOSE dataset also provides both annotated image and 3D point-cloud data gathered in off-road scenarios present in a forested region of Germany.

The work on developing synthetic datasets for autonomous navigation tasks are important due to factors such as the cost of collecting data covering all structured and unstructured scenarios, which can hinder the robustness of the trained model due to perceptual uncertainty [11] and the errors in annotated data for 3D LiDAR point-clouds [12] and RGB images usually present in multimodal datasets. These drawbacks can be rectified by the use of synthetic data, which can simulate time-of-day and weather combinations [13], cost-effectively represent different geographical locations and produce annotated image and point cloud data automatically [14] using simulator metadata. One such dataset is the Virtual KITTI [15] dataset, which uses synthetic images to simulate different weather conditions corresponding to the scenarios present in the real-world KITTI dataset. Furthermore, researchers have introduced methods for using large simulated environments for autonomous navigation [1], which offer customisable urban environments to train and test autonomous agents in European and North-American urban scenarios. Others have presented the use of existing video-game engines such as Unreal Engine combined with procedural generation to mimic weather conditions and emergent elements an autonomous agent might face in urban environments [13, 16] and planetary exploration [17]. Synthetic data generation for off-road navigation has also seen recent research work [18] using the ROS Gazebo environment to produce a multimodal synthetic dataset for off-road navigation. However, the prior work does not perform any evaluation of the collected data based on experiments with semantic segmentation tasks.

III. Methodology

This research work introduces a framework to generate high fidelity synthetic datasets by utilising Unreal Engine 5 and the NVIDIA Isaac Sim robotics simulator.

A. NVIDIA Isaac Sim platform

The Isaac Sim platform was selected for the dataset preparation due to its tools for robot designing, development and navigation. The platform is also capable of simulating legged, wheeled, articulated and flying robots for localisation, mapping, motion planning and control, while also simulating robotic sensors. Additionally, it is part of the NVIDIA Omniverse modular software package, which includes integrations for robot machine learning, 3D modelling, industrial robot simulation, and digital twin development. Importing 3D environments from Unreal Engine 5 to Isaac Sim is done via an extension provided by the Omniverse software platform. The synthetic dataset generation through Isaac Sim is, therefore, modular, streamlined, and expandable for additional research requirements.

B. The synthetic 3D environment generation

Our synthetic dataset generation process comprises of two main steps,

1. 3D synthetic environment modelling in Unreal Engine 5.
2. Robot simulation and sensor data collection within the synthetic 3D environments using the Isaac Sim platform.

The 3D environment generation process on Unreal Engine 5 was facilitated by utilising an available open source Unreal Engine 5 asset pack [19], which includes high quality photogrammetry-based captured collection of a rural Australian landscape including logs, flora, roads and fences. The two synthetic environment scenarios were designed in the Unreal Engine by organising these assets along with some additional assets imported from the Unreal Engine marketplace and the Quixel Megascans library. The first environment consists of scenes in a rural environment with two fences, an asphalt road and road signs, surrounded by bushland, while the second environment is modelled after a natural forest scenario. Together, these two 3D environments cover 18 semantic classes. After the generation of these environments using Unreal Engine 5, the necessary files including universal scene description (USD), material description and other asset related files were imported to the Isaac Sim simulator environment using the Omniverse Unreal Engine Connector extension and then saved in the Omniverse Nucleus local server.

Isaac Sim simulator platform additionally supports the Unity 3D engine. Therefore, the above steps we have followed for integrating an 3D environment to Isaac Sim can be replicated for the Unity 3D engine as well, enabling the development of 3D environments using Unity as an alternative to the Unreal Engine.

C. Robot simulation and sensor data capturing

In this research, Isaac Sim platform was utilised to conduct the robot simulation and sensor data capturing process within the imported 3D environments. The simulation procedure was developed using the Isaac Sim graphical user interface along with the Isaac Sim Python programming interface, which was used to define physics properties such as ground plane, rigid body, collider properties and robot motion planning. A differential wheeled robot was selected for the simulation from the Isaac Sim-provided wheeled robot library, which includes the Carter, Clearpath Jackal, Jetbot, iw.hub, and Create 3 robots. For our simulation, we use the NVIDIA Carter v1 robot as of the base platform incorporates camera, LiDAR and IMU sensors. The physical properties and sensor details of NVIDIA's first Carter robot version, Carter v1 are included in Table 1.

1) The Carter v1 robot and its sensor configuration: The Carter v1 robot has two front differential wheels and one caster wheel at the back. This robot uses the Jetson AGX Xavier developer kit to process the robot data. The selected robot NVIDIA Carter v1 USD file and relevant asset files have been imported into the simulation from the Omniverse local server. The robot wheel radius and wheelbase dimensions were defined in the wheeled robot controller while the robot LiDAR sensor and IMU sensor configurations were included in the Python script of the simulation. The physics simulation process was run at a frequency of 60 Hz and the robot navigation through the 3D environment was achieved by following a pre-determined trajectory. The monocular RGB and stereo images with respective semantic annotations were captured for the dataset at a 5Hz frequency. The IMU and LiDAR depth data were saved with the relevant physics step and time stamp details. The information of sensor position and orientation is included in Table 2. The counterclockwise rotation about X, Y and Z axes in Table 2 is considered as the positive rotation of each sensor.

Table 1. Carter v1 physical properties and sensor payload.

Depth camera	Intel RealSense D435
Stereo camera	ZED
3D LiDAR	Velodyne VLP-16
IMU	Bosch BMI160
Wheelbase	0.54 m
Wheel radius	0.24 m

Table 2. Carter v1 sensor pose details for robot simulations.

Sensor	X (m)	Y (m)	Z (m)	Roll (deg)	Pitch (deg)	Yaw (deg)
RGB	0.089	0	0.327	90	270	0
Stereo left	0.103	0.06	0.310	90	-90	0
Stereo right	0.103	-0.06	0.310	90	-90	0
3D LiDAR	-0.06	0	0.38	0	0	0
IMU	0	0	0	0	0	0

The X, Y, and Z (0,0,0) is defined at the midpoint of the front differential wheel axle of the Carter v1 robot as shown in Figure 1. The monocular RGB camera captures images at a resolution of 1920×1080 pixels with a horizontal field of view (FoV) of 121.5 degrees and a vertical FoV of 73.5 degrees.

The stereo RGB camera captures images at the same resolution as the monocular camera with a horizontal FoV of 110 degrees and a vertical FoV of 70 degrees. The Velodyne VLP-16 3D LiDAR has a range of 0.05-120 m and an operating frequency of 5-20 Hz at respective resolutions of 0.09-0.36 degrees. In the simulation, the sensor captures point cloud data at a frequency of 20 Hz. The IMU sensor (Bosch BMI160), which was attached to the Carter v1 chassis, has a data acquisition rate of 25/32 Hz~1600 Hz. However, in the simulation it operates at a frequency of 60 Hz.

2) Robot sensor data capturing scenarios: In the first simulation, the robot operates within a rural Western Australian scenario that consists of 17 semantic classes including two fences and an asphalt road. The second robot simulation captures a scenario with higher vegetation density in comparison to the first scenario and covers 17 semantic classes including logs, fallen leaves, and small rocks. In each of the synthetic environment scenarios, the robot follows the set trajectory for approximately 6 minutes and 48 seconds. Figures 2 and 3 show two examples from the first and second simulation scenarios, respectively.

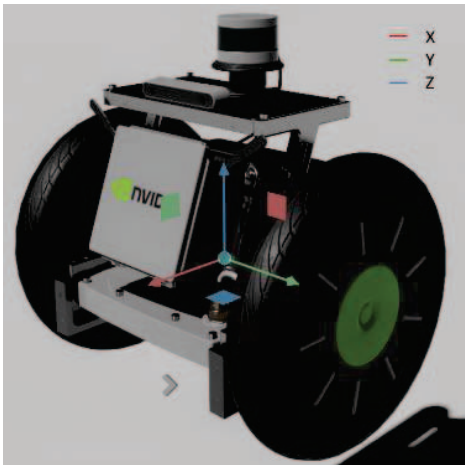

Figure 1. NVIDIA Carter v1 robot in Isaac Sim.

Figure 2. RGB (left) and annotated (right) images from Scenario 1.

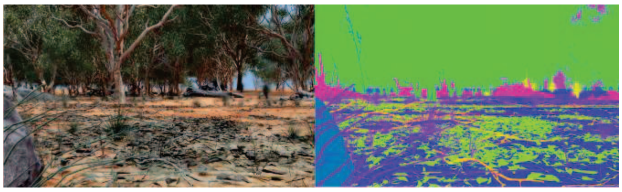

Figure 3. RGB (left) and annotated (right) images from Scenario 2.

IV. Experiments

In order to investigate the need for developing models dedicated to off-road navigation scenarios, the conducted experiments are as follows. First, we analyse the performance of state-of-the-art semantic segmentation models for urban navigation tasks on the semantic annotated monocular RGB images of OUTBACK dataset. Furthermore, we evaluate the Sim2Real potential of our synthetic dataset using an off-road centric semantic segmentation model and a real-world off-road dataset. This is accomplished by comparing the performance of the off-road model pre-trained on our dataset and then tested using the test set of the RELLIS-3D real-world dataset.

A. Evaluation metrics

In the experiments, we evaluated the results using metrics commonly applied in semantic segmentation research, including intersection over union (IoU), mean intersection over union (mIoU), and mean pixel accuracy (mPA). These metrics are defined as

$$IoU = \frac{TP_c}{TP_c+FP_c+FN_c}, \quad (1)$$

$$mIoU = \frac{1}{n}\sum_{c=1}^{c=n}\frac{TP_c}{TP_c+FP_c+FN_c}, \quad (2)$$

$$mPA = \frac{1}{n}\sum_{c=1}^{c=n}\frac{TP_c}{TP_c+FP_c}, \quad (3)$$

where TP_c, FP_c and FN_c are the number of predicted true positive, false positive and false negative pixels, respectively, in class *c*, and the total number of semantic classes is given by *n*.

B. Experiment-1: Performance evaluation of the state-of-the-art semantic segmentation models on the synthetic dataset

The semantic segmentation models for the first experiment were chosen while taking into consideration the requirements of the off-road navigation task. First, we chose a state-of-the-art model with high mIoU and mPA metrics on urban datasets, which is, therefore, anticipated to be suitable for the high accuracy requirements in off-road navigation for cases where model complexity is not constrained.

The second model was chosen for its lightweight architecture, making it more suitable for deployment on a mobile robot. A brief description of the models used in the first experiment are given below:

1. OneFormer [20]: The first model, OneFormer, is a state-of-the-art encoder-decoder transformer model holding a mIoU of 84.6 on the semantic segmentation task in the Cityscapes validation dataset.
2. DDRNet [21]: The second model, DDRNet, is a lightweight two-branch architecture-based model introduced in and is designed for real-time semantic segmentation. The implementation used for the experiments are from the MMSegmentation [22] library. It reports an mIoU of 82.4 on Cityscapes test data for semantic segmentation.

The DDRNet model and the OneFormer model are trained on the OUTBACK dataset with 18 semantic classes and then evaluated using a segregated test set from the same synthetic dataset.

C. Experiment-2: Sim2Real performance evaluation of the GA-Nav model trained on the synthetic dataset

For the second experiment, we chose GA-Nav, a model with the current best state-of-the-art performance for off-road navigation tasks. The GA-Nav model is an encoder-decoder model and it achieves mIoU of 74.44 and 89.08 on the RELLIS-3D and RUGD real-world off-road navigation datasets respectively.

The GA-Nav model was trained on the synthetic OUTBACK dataset and then evaluated using the test set from the RELLS-3D dataset. The base GA-Nav model is then trained from its initial state on the RELLIS-3D train and validation dataset and then evaluated using the RELLIS-3D test set. The GA-Nav model uses six semantic classes defined based on the navigability levels of the observed terrain. These navigability levels are defined as Background (0), L1-Smooth (1), L2-Rough (2), L3-Bumpy (3), Non-navigable (4) and Obstacle (5).

Table 3. Class label mapping from OUTBACK dataset to GA-Nav navigability levels.

Class	**Background**	**Grass tree**	**Pole**	**Tree**	**Leaves**	**Fence net**	**Log**	**Grass**	**Road sign**
Navigability level	0	5	5	5	1	5	5	1	5
Class	**Small branch**	**Gravel**	**Ground**	**Horizon**	**Roots**	**Sky**	**Delineator**	**Rock**	**Road**
Navigability level	5	1	1	0	5	0	5	5	1

Table 4. Class label mapping from RELLIS-3D dataset to GA-Nav navigability levels.

Class	Void	Dirt	Grass	Tree	Pole	Water	Sky	Vehicle	Object	Puddle
Navigability level	0	1	4	5	4	4	0	4	4	4
Class	**Asphalt**	**Building**	**Log**	**Person**	**Fence**	**Bush**	**Concrete**	**Barrier**	**Rubble**	**Mud**
Navigability level	1	4	5	4	5	5	1	4	1	4

Due to the difference between the environments and semantic classes present in the OUTBACK and RELLIS-3D datasets, when defining the GA-Nav terrain navigability levels for the RELLIS-3D dataset, we selected the common classes available in both datasets and relegated the extraneous classes of the RELLIS-3D dataset to the Non-navigable (4) navigability level. Tables 3 and 4 show the mapping of the class labels present in the respective OUTBACK dataset and the RELLIS-3D dataset to the relevant navigability levels. Furthermore, due to the extraneous classes present in the RELLIS-3D dataset, we define a test set derived from the RELLIS-3D dataset only consisting of the intersection of semantic classes present in both datasets.

D. Implementation

Our experimental setup comprised two NVIDIA RTX A6000 GPUs with 48 GB GDDR6 memory capacity. Each model was trained with a batch size of 2 for 40k iterations.

V. RESULTS

We present the results obtained in the experiments conducted using the DDRNet and OneFormer models as well as the results of the Sim2Real performance conducted using the GA-Nav semantic segmentation model.

A. Results of experiment-1: Performance of state-of-the-art semantic segmentation models on the synthetic dataset

In Table 5, we present the mIoU and mPA metrics obtained from training the DDRNet and OneFormer models on the OUTBACK synthetic dataset and testing on the segregated OUTBACK test set for the semantic segmentation task. The lightweight DDRNet model has shown lower mIoU and mPa values when compared with the more capable OneFormer model. This is in-line with the expectations that the OneFormer model with its more complex architecture will outperform the simpler DDRNet model. Both models' mIoU metrics are lower for the OUTBACK dataset compared to their reported results on the on Cityscapes dataset. This can be attributed to the lower iteration count selected for the experiments combined with the unstructured nature of the environments simulated in the OUTBACK dataset.

B. Results of experiment-2: Sim2Real performance of the GA-Nav model trained on the synthetic dataset

The results obtained from the second experiment are presented in Table 6. As discussed in the previous section, the mIoU is calculated considering only the Background (0), L1-Smooth (1), and Obstacle (5) navigability levels. As expected, the mIoU and mPa metrics of the model trained and tested on the RELLIS-3D dataset are higher than the same metrics obtained using the model trained on the synthetic OUTBACK dataset and tested on the RELLIS-3D dataset's test set.

However, the results in the Background navigability level are comparable in both experiments, which can be explained by the fact that both datasets have visually similar and frequently occurring classes present in the Background navigability level. The IoU value achieved for the L1 navigability level for the experiments, while relatively low, does show promise towards the potential to achieve improved Sim2Real results when tested with a real-world dataset with which it will share more common environmental features. However, the environments present in the RELLIS-3D dataset for the L1, and Obstacle navigability classes are highly divergent from the environment presented in the synthetic dataset, with the latter dataset representing a harsh desert landscape as opposed to the verdant forested landscape present in the former. We have presented Figure 4 with images from the test set of the RELLIS-3D dataset and an example from the OUTBACK training set, with the corresponding ground truth annotations to highlight this observation. In the figure, the RELLIS-3D dataset's classes, trees and bushes, are shown in light green and pink respectively and are mapped to the Obstacle navigable level. Similarly, on the right we have presented the examples from the OUTBACK synthetic dataset consisting of classes tree, log and grass tree, which are mapped to the same navigable level and shown in green, pink and yellow respectively. Therefore, we can see that the images used to train the GA-Nav model on the synthetic dataset differ vastly from the images the model is then evaluated on, with disparities visible in the shapes, sizes and annotation granularity of the ground truth images used to train and test the model.

Table 5. Semantic segmentation models' results by training and testing on the OUTBACK dataset.

Model	Classes	mIoU	mPA
DDRNet	18	54.28	64.56
OneFormer	18	64.08	74.65

Table 6. GA-Nav model's performance on the RELLIS-3D test set.

Metrics	Navigability level	Model trained on OUTBACK dataset and tested with RELLIS-3D dataset	Model trained and tested with RELLIS-3D dataset
IoU	**Background**	66.62	94.73
	L1	21.8	43.19
	Non-navigable	-	-
	Obstacle	20.37	82.15
mIoU	**All levels**	36.26	73.36
mPA	**All levels**	45.92	83.14

Figure 4. Comparison of RGB and annotated images of RELLIS-3D test set (left) and OUTBACK training set (right).

These distinct contrasts between the images of the two datasets can account for the difference in the results achieved in the Obstacle navigable level, by the GA-Nav model.

VI. Conclusions

In this paper, we have presented a novel methodology for generating synthetic datasets for off-road semantic segmentation tasks using NVIDIA's Isaac Sim platform in conjunction with Unreal Engine 5 and high-quality photogrammetry assets. We also publish an off-road dataset based on the rural Western Australian environment consisting of stereo and monocular RGB images, LiDAR point clouds and IMU sensor data. Our experiments using this synthetic dataset have shown that promising results can be achieved in Sim2Real transfer of semantic segmentation tasks after training with this kind of high-quality synthetic data. The disparity between the metrics achieved between the GA-Nav models trained with our synthetic dataset and real-world dataset can be attributed to the substantial difference in the environments depicted in the two datasets, further demonstrating the importance of being able to produce datasets representing the diversity of real-world terrains. In future work, we aim to further refine the synthetic dataset and conduct experiments on Sim2Real transfer with real-world Western Australian off-road data.

References

[1] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun, "CARLA: An open urban driving simulator," in Conference on robot learning, 2017: PMLR, pp. 1-16.

[2] M. Cordts et al., "The cityscapes dataset for semantic urban scene understanding," in Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 3213-3223.

[3] T. Guan, D. Kothandaraman, R. Chandra, A. J. Sathyamoorthy, K. Weerakoon, and D. Manocha, "Ga-nav: Efficient terrain segmentation for robot navigation in unstructured outdoor environments," IEEE Robotics and Automation Letters, vol. 7, no. 3, pp. 8138-8145, 2022.

[4] P. Jiang, P. Osteen, M. Wigness, and S. Saripalli, "Rellis-3d dataset: Data, benchmarks and analysis," in 2021 IEEE international conference on robotics and automation (ICRA), 2021: IEEE, pp. 1110-1116.

[5] F. Yu et al., "Bdd100k: A diverse driving dataset for heterogeneous multitask learning," in Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2020, pp. 2636-2645.

[6] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The kitti dataset," The International Journal of Robotics Research, vol. 32, no. 11, pp. 1231-1237, 2013.

[7] "NVIDIA Isaac Sim." https://developer.nvidia.com/isaac/sim (accessed 2024).

[8] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in 2012 IEEE conference on computer vision and pattern recognition, 2012: IEEE, pp. 3354-3361.

[9] M. Wigness, S. Eum, J. G. Rogers, D. Han, and H. Kwon, "A rugd dataset for autonomous navigation and visual perception in unstructured outdoor environments," in 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2019: IEEE, pp. 5000-5007.

[10] P. Mortimer, R. Hagmanns, M. Granero, T. Luettel, J. Petereit, and H.-J. Wuensche, "The GOOSE Dataset for Perception in Unstructured Environments," arXiv preprint arXiv:2310.16788, 2023.

[11] K. Czarnecki and R. Salay, "Towards a framework to manage perceptual uncertainty for safe automated driving," in Computer Safety, Reliability, and Security: SAFECOMP 2018 Workshops, ASSURE, DECSoS, SASSUR, STRIVE, and WAISE, Västerås, Sweden, September 18, 2018, Proceedings 37, 2018: Springer, pp. 439-445.

[12] M. Sánchez, J. L. Martínez, J. Morales, A. Robles, and M. Morán, "Automatic generation of labeled 3D point clouds of natural environments with Gazebo," in 2019 IEEE International Conference on Mechatronics (ICM), 2019, vol. 1: IEEE, pp. 161-166.

[13] S. Khan, B. Phan, R. Salay, and K. Czarnecki, "ProcSy: Procedural Synthetic Dataset Generation Towards Influence Factor Studies Of Semantic Segmentation Networks," in CVPR workshops, 2019, vol. 3, p. 4.

[14] J. L. Martinez, M. Moran, J. Morales, A. Robles, and M. Sanchez, "Supervised learning of natural-terrain traversability with synthetic 3D laser scans," Applied Sciences, vol. 10, no. 3, p. 1140, 2020.

[15] A. Gaidon, Q. Wang, Y. Cabon, and E. Vig, "Virtual worlds as proxy for multi-object tracking analysis," in Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 4340-4349.

[16] M. D. Moniruzzaman, A. Rassau, D. Chai, and S. M. S. Islam, "Long future frame prediction using optical flow-informed deep neural networks for enhancement of robotic teleoperation in high latency environments," *Journal of Field Robotics,* vol. 40, no. 2, pp. 393-425, 2023, doi: https://doi.org/10.1002/rob.22135.

[17] R. O. Chavez-Garcia, J. Guzzi, L. M. Gambardella, and A. Giusti, "Learning ground traversability from simulations," IEEE Robotics and Automation letters, vol. 3, no. 3, pp. 1695-1702, 2018.

[18] M. Sánchez, J. Morales, J. L. Martínez, J. J. Fernández-Lozano, and A. García-Cerezo, "Automatically annotated dataset of a ground mobile robot in natural environments via gazebo simulations," Sensors, vol. 22, no. 15, p. 5599, 2022.

[19] A. S. Hamilton. Rural Australia [Online] Available: https://www.unrealengine.com/marketplace/en-US/product/rural-australia

[20] J. Jain, J. Li, M. T. Chiu, A. Hassani, N. Orlov, and H. Shi, "Oneformer: One transformer to rule universal image segmentation," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 2989-2998.

[21] Y. Hong, H. Pan, W. Sun, and Y. Jia, "Deep dual-resolution networks for real-time and accurate semantic segmentation of road scenes," arXiv preprint arXiv:2101.06085, 2021.

[22] MMSegmentation contributors, "MMSegmentation: OpenMMLab Semantic Segmentation Toolbox and Benchmark," 2020 2020. [Online]. Available: https://github.com/open-mmlab/mmsegmentation.