Addressing Mode Collapse in GANs for Autonomous Vehicle Simulation

Om R Muddapur[1], Chandanagouda B H[2], Vidhi Patel[3], Swati Bhat[4], Sharada K Shiragudikar[5]

[1,2,3,4,5]School of Computer Science and Engineering, KLE Technological University, Street, Hubli, Karnataka, India.

Contributing authors: 01fe22bcs262@kletech.ac.in; 01fe22bcs287@kletech.ac.in; 01fe22bcs269@kletech.ac.in; 01fe22bcs273@kletech.ac.in; sharada.shiragudikar@kletech.ac.in;

Abstract

Generative Adversarial Networks (GANs) have demonstrated significant potential in generating realistic images for simulating autonomous driving scenarios. However, a persistent challenge known as mode collapse—where the generator produces limited or repetitive outputs—continues to hinder output diversity and model robustness. This paper proposes an enhanced GAN framework designed to address mode collapse by integrating minibatch discrimination within the discriminator and a perceptual loss component in the generator. The model is trained using a combination of real-world urban scene images and data from the CARLA simulator, incorporating architectural improvements and regularization techniques. Experimental results show marked improvements in image diversity and realism, achieving a Peak Signal-to-Noise Ratio (PSNR) of 21.72 dB and an Structural Similarity Index (SSIM) of 0.721, alongside visually improved image quality across training epochs. The proposed approach offers a stable training process and enhanced image fidelity, making it a valuable contribution to the generation of high-quality synthetic data for autonomous vehicle systems.

Keywords: Generative Adversarial Networks, Autonomous Driving, Mode Collapse, Urban Scene Generation, Minibatch Discrimination

1 Introduction

Autonomous cars are a revolutionary development in transport technology, with the potential to reduce traffic accidents, optimize transport, and enhance accessibility to travel. Autonomous systems rest on sophisticated machine learning algorithms that have to respond in real-time to dynamic, uncertain driving situations. The performance of these algorithms significantly depends on access to diverse, high-quality, and large-sized datasets with a broad range of scenarios including varying lighting conditions, road topographies, traffic patterns, and weather conditions. Such real-world data acquisition, however, is logistically cumbersome, expensive, and risky, especially for gathering atypical or hazardous driving situations.

Simulation-based training and synthetic data generation have therefore become increasingly popular to overcome such difficulties. Simulators like CARLA (Car Learning to Act) have proved to be very useful in this regard [1], with safe and effective generation of rich urban driving scenarios having configurable parameters such as time of day, weather, road topology, and traffic flow. To further enhance these simulated environments, researchers have translated the application of Generative Adversarial Networks (GANs), originally developed by Goodfellow et al. in 2014 [2]. GANs have demonstrated great capability in producing realistic images close to real-world driving scenarios [3] and are therefore particularly suited to assist simulation data and add diversity to training input for autonomous vehicle models.

Deep learning has shown remarkable success in solving complex, data-intensive tasks across domains like agriculture and healthcare. Recent applications include predicting crop resilience and diagnosing diseases [4–6], as well as medical image-based diagnosis and prognosis [7, 8]. These advancements highlight the potential of deep learning in data generation and imitation, supporting further research into generative AI for autonomous vehicles.

However, traditional GANs often suffer from mode collapse, where the generator produces limited image variations, reducing scene diversity and model robustness. Addressing this issue is crucial for improving the generalization of autonomous systems to real-world scenarios. This paper presents an enhanced GAN architecture that tackles mode collapse using *minibatch discrimination* and a *perceptual loss* module. The proposed DCGAN is trained on a combination of real urban scenes and CARLA-simulated images, with architectural and training enhancements to generate high-quality, diverse outputs.

The remaining part of the paper is structured as follows: Section II covers related work and preliminary concepts. Section III presents the proposed model with its structure, mode collapse prevention methods, and training process. Section IV details experimental results and performance evaluation tests. Finally, Section V concludes the paper by presenting main contributions and mapping out directions for future research.

2 Background Study

Generative Adversarial Networks (GANs), introduced by Goodfellow et al. in 2014 [2], have significantly advanced synthetic image generation. Their ability to create realistic

images from random noise makes them valuable for autonomous vehicle simulation, enabling perception systems to train on diverse scenarios with minimal real-world data.

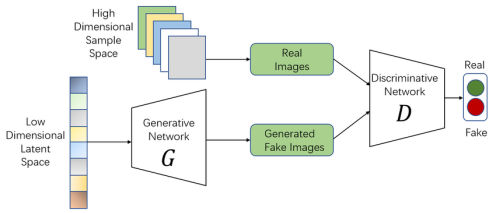

Fig. 1 Architecture of GAN

[9]

As shown in Fig. 1 [9], a typical GAN consists of a Generator that creates synthetic images from noise and a Discriminator that distinguishes real from fake images. Through adversarial training, both models improve iteratively. A popular variant, Deep Convolutional GAN (DCGAN), uses convolutional layers instead of fully connected ones, enhancing its ability to model spatial hierarchies and generate high-resolution images.

To improve data diversity under challenging conditions, researchers have developed GAN variants like CycleGAN [10], which performs unpaired image-to-image translation (e.g., converting clear to foggy scenes), and DeblurGAN [11], which enhances image sharpness under motion blur for better low-visibility perception.

Despite such progress, a fundamental challenge remains: *mode collapse*, where the generator produces limited output variations. This leads to repetitive scenes in simulation, limiting the training diversity needed for robust real-world performance.

Mode collapse often stems from unstable dynamics between the generator and discriminator. Techniques like *feature matching* and *perceptual loss* have been proposed to address this. Feature matching aligns the generator with the discriminator's internal representations, while perceptual loss compares high-level semantic features instead of raw pixels [12].

Other approaches include *mini-batch discrimination*, which assesses sample diversity within a batch, and the use of **Wasserstein GAN** (WGAN) [13], which introduces a smoother loss for better convergence. WGAN-GP further improves training stability with gradient penalty regularization.

Simulation platforms like **CARLA** [14] rely on data variety to effectively train and validate perception and decision-making systems under diverse driving conditions. The ability of GANs to produce diverse, high-quality data is therefore crucial for their utility in such platforms.

Xu et al. [15] observed that while synthetic data alone may lack reliability, combining it with real-world data boosts model robustness and adaptability. Sankar's survey [16] further highlights how architectural innovations help simulate complex environments like fog, snow, and night driving.

Similarly, Huang et al.'s "Yes, We GAN" paper [17] showed that GANs can enhance object detection and scene understanding, but stressed that mode collapse degrades performance—underscoring the need to preserve diversity.

Recent research has adopted hybrid strategies, such as generating mid-level representations (e.g., bird's eye views) before rendering full scenes, allowing better output control. GANs are also increasingly combined with domain adaptation techniques to bridge the gap between synthetic and real-world data.

In summary, GANs are powerful tools for synthetic data generation in autonomous driving, but their success depends on producing diverse, high-fidelity outputs. Addressing mode collapse through architectural, loss function, and training innovations is essential for generating more stable and varied simulation data, ultimately enabling safer autonomous systems.

Several studies have proposed solutions:

- Techniques like *feature matching*, *perceptual loss*, and *mini-batch discrimination* improve data diversity.
- Simulation tools like **CARLA** help evaluate the effectiveness of generated images in driving simulations [14].

3 Proposed Model

This section presents an enhanced Deep Convolutional Generative Adversarial Network (DCGAN) framework tailored to generate diverse and realistic urban driving scenes. The model incorporates architectural improvements and training techniques specifically designed to address the issue of mode collapse—an inherent limitation in traditional GANs that results in repetitive or low-diversity outputs.

As illustrated in Fig. 2, the proposed pipeline begins with data acquisition from real-world urban driving environments and synthetic datasets generated using the CARLA simulator. These inputs undergo standard preprocessing, including resizing and normalization, to ensure consistency for training.

The core GAN architecture consists of a Generator and a Discriminator. The Generator employs six transposed convolution layers with Batch Normalization and ReLU activation, generating images from random noise. The Discriminator is built with convolutional layers, LeakyReLU activation, and includes a Minibatch Discrimination module to evaluate diversity across samples.

To effectively mitigate mode collapse, two mechanisms are incorporated:

- **Minibatch Discrimination** compares features within a batch to encourage variety in generated outputs.
- **Perceptual Loss** (L1-based) guides the Generator to produce images that are not only visually similar but also semantically consistent with real images.

Training is carried out using the Adam optimizer with a batch size of 64 images. The effectiveness of the model is evaluated using PSNR and SSIM metrics, along with visual analysis. The results demonstrate that the proposed architecture outperforms baseline GAN and DCGAN models by producing images with higher fidelity and

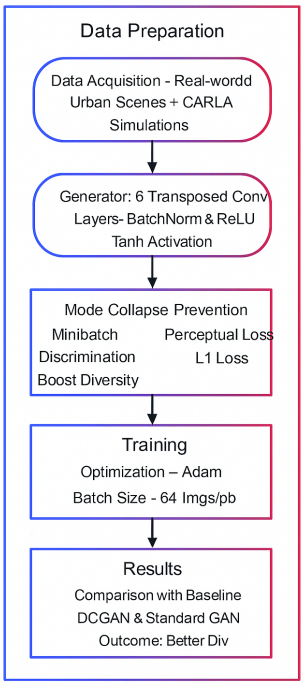

Fig. 2 Simplified pipeline of the proposed DCGAN framework showing key components such as data preparation, generator architecture, mode collapse prevention strategies, training setup, and final outcomes.

greater structural diversity, making it well-suited for autonomous vehicle simulation tasks.

3.1 Overview of DCGAN Architecture

The suggested architecture is made up of two main neural networks: the **Generator** and the **Discriminator**, both of which are developed based on deep convolutional layers in PyTorch.

The **Generator** maps a randomly drawn 100-dimensional vector from a normal distribution to a 128×128 color image. Six transposed convolutional layers are applied in succession, each followed by batch normalization and ReLU activation. A Tanh activation is applied to the final layer to scale output values between $[-1, 1]$, which corresponds to normalized input data.

The **Discriminator** takes in 128×128 RGB images through five convolution layers, each making use of LeakyReLU activations as well as batch normalization (apart from the first one). A minibatch discrimination layer is added specifically to enhance the network's capability to detect lack of diversity. Lastly, a fully connected Sigmoid-activated layer produces a single probability representing the possibility of the input being real or fake.

3.2 Enhancements to Mitigate Mode Collapse

To reduce the risk of the Generator producing repetitive outputs (mode collapse), the model integrates the following strategies:

- **Minibatch Discrimination:** Incorporated in the Discriminator, this technique measures feature differences between samples in the same batch, encouraging output variety from the Generator. The associated loss term is formulated as:

$$\mathcal{L}_{\text{MB}} = \frac{1}{B^2} \sum_{m=1}^{B} \sum_{n=1}^{B} \left\| \mathbf{f}_m - \mathbf{f}_n \right\|_1, \tag{1}$$

  where B is the batch size and $\mathbf{f}_m$ is the extracted feature vector of the m-th sample. This formulation penalizes homogeneity among samples within a batch.
- **Perceptual Loss:** To enhance visual quality, the Generator is guided not only by adversarial loss but also by a pixel-level similarity measure. This L1-based loss compares generated and real images:

$$\mathcal{L}_{\text{perc}} = \frac{1}{H\,W} \sum_{u=1}^{H} \sum_{v=1}^{W} \left| \hat{I}_{u,v} - I_{u,v} \right|, \tag{2}$$

  where $\hat{I}$ is the generated image, I is a corresponding real image, and H, W represent image height and width.

3.3 Dataset and Preprocessing

Training data includes urban driving scenes derived from real vehicle camera captures and synthetic images generated from the CARLA simulator. Ten subdirectories of data representative of diverse angles and drive sessions are processed. All the images are resized to 128×128 and normalized to range $[-1, 1]$ with mean and standard deviation set at 0.5 for every channel. Utilities in PyTorch such as ImageFolder, ConcatDataset, and DataLoader are utilized for efficient loading and batching.

3.4 Training Details

The model is trained for more than 100 epochs at a batch size of 64. Generator and Discriminator are optimized alternately with the Adam optimizer. Learning rates are 0.0002 for the Generator and 0.0001 for the Discriminator, both having momentum parameters $\beta_1 = 0.5$ and $\beta_2 = 0.999$. To favor convergence, learning rates are decayed by half every 30 epochs with the use of the
textttStepLR scheduler.

The Discriminator is updated to distinguish between fake and real images, and the Generator is also updated to produce realistic images that can trick the Discriminator. This alternating approach enables the two networks to co-evolve. Losses employed at training time are outlined below:

- **Discriminator Loss:** A binary cross-entropy loss is used to train the Discriminator. It is defined as:

$$\mathcal{L}_D = -\mathbb{E}_{\mathbf{x}\sim p_{\text{data}}}\big[\ln D(\mathbf{x})\big] - \mathbb{E}_{\mathbf{z}\sim p_z}\big[\ln\big(1 - D\big(G(\mathbf{z})\big)\big)\big], \tag{3}$$

where $\mathbf{x}$ is a real image, and $G(\mathbf{z})$ is a generated image from noise vector $\mathbf{z}$.

- **Generator Loss:** The Generator is optimized using a combination of adversarial and perceptual losses:

$$\mathcal{L}_G = -\mathbb{E}_{\mathbf{z}\sim p_z}\big[\ln D\big(G(\mathbf{z})\big)\big] + \alpha\,\mathcal{L}_{\text{perc}}, \tag{4}$$

where $\alpha = 0.1$ balances the perceptual term with the adversarial objective.

The generated images are stored at every 10th epoch to visually inspect progress and diversity. The capability of the model to mimic realistic driving conditions is constantly inspected throughout training.

3.5 Evaluation Metrics

The generated image quality is measured in terms of two widely used metrics:

- **PSNR** Measures pixel-level similarity. Increased PSNR indicates greater resemblance to actual images.
- **SSIM** Measures structural consistency and perceived similarity between generated and actual images.

Both of these metrics are calculated across a batch of 25 generated samples against the corresponding ground-truth images from the data set. This makes sure that the model is tested on both quantitative and perceptual grounds.

4 Results and Analysis

The performance of the suggested model is measured in terms of generated image quality, diversity, and the quality of strategies implemented to reduce mode collapse. Evaluation metrics employed are **PSNR** and **SSIM**. Visual results are also presented to show the quality improvements in images with respect to training epochs.

4.1 Quantitative Evaluation

In order to analyze the quality of the resulting images, the below-specified metrics were calculated for both real and generated images:

- **PSNR:** This measures the pixel-wise similarity between the real image and generated image. The higher the value of PSNR, the better the quality.
- **SSIM:** This measure determines the structural similarity of real and generated images, looking at perceptual content. The greater the SSIM score, the closer the generated image is to the real image in structure and visual content.

Throughout training, the model consistently demonstrated a rise in both PSNR and SSIM, showing that the Generator was indeed generating more realistic and varied images as epochs progressed. After 100 epochs, the final test metrics were as follows:

- **PSNR:** 21.72 dB

- **SSIM:** 0.721

These values reflect a notable increase in the image quality over baseline models. The substantial SSIM value, especially, reflects that the perceptual organization of the generated images closely resembles real urban road scenes. These values not only confirm improved image quality, but the consistent rise in both metrics across epochs indicates stable convergence and a reduction in generator artifacts. The PSNR value of 21.72 dB reflects low pixel-level noise, while the SSIM score of 0.721 shows that the generated images preserve structural consistency and perceptual similarity. These improvements can be directly attributed to the perceptual loss component, which enhances semantic alignment with real images, and the use of minibatch discrimination, which increases output diversity.

4.2 Qualitative Evaluation

Figures 3 and 4 show visual results of the generated images over multiple epochs. It shows the continuous improvement of realism and diversity in the generated road scenes as training continues. The early-stage generated figures display conspicuous artifacts and diversity deficits, whereas the later-stage figures show more sophisticated and realistic features, including diversified lighting, road features, and traffic components.

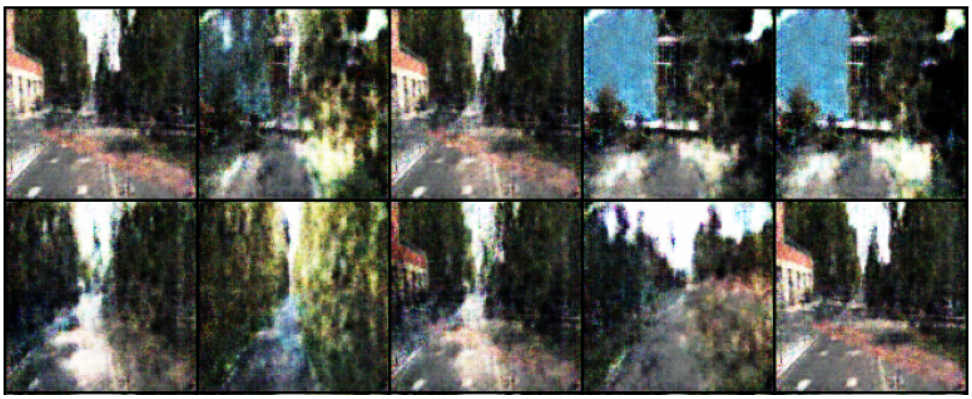

Fig. 3 Generated Images at initial Epochs: Early-stage images with artifacts and limited diversity.

In early epochs, the images often suffered from mode collapse, showing repetitive textures and unnatural object outlines. As training progressed, lighting conditions, road geometries, and vehicle shapes became more realistic. The model began to synthesize scenes with greater variation in environmental features such as lane curvature, shading, and roadside textures. This visual diversity aligns well with the quantitative trends seen in PSNR and SSIM.

Fig. 4 Generated Images at Epoch 100: Improved image quality with diverse and realistic features.

4.3 Impact of Architectural and Training Modifications

The improvements in image quality and diversity can be attributed to several architectural and training modifications:

- **Minibatch Discrimination:** The implementation of minibatch discrimination helped the Generator avoid producing repetitive images, thereby enhancing the diversity of the generated scenes.
- **Perceptual Loss:** The addition of perceptual loss (L1) guided the Generator to produce images that are perceptually closer to real-world urban road scenes, improving both image quality and realism.

As a whole, these adjustments ensured the training process was stable and enabled the Generator to generate high-quality and diverse urban road scenes for autonomous vehicle simulation.

4.4 Comparison with Baseline Models

In order to evaluate the performance of the suggested architecture of GAN, we compared it with baseline models such as a traditional GAN and a DCGAN trained without minibatch discrimination and perceptual loss. The baseline models are widely used in image synthesis but tend to experience mode collapse and poor visual diversity.

The comparison relies upon two popular quantitative metrics:

- **PSNR:** Measures the relation between the maximum achievable power of a signal and the corrupting noise power. The greater PSNR, the higher the quality of the image.
- **SSIM:** Assesses perceived image quality in terms of luminance, contrast, and structure. Increasing value of SSIM indicates increased structural similarity to actual images.

Table 1 Comparison of Proposed Model with Baseline and GAN Literature Models

Model	PSNR (dB)	SSIM
Standard GAN [2]	16.84	0.612
DCGAN [18]	18.35	0.654
Proposed Model (w/ Minibatch + Perceptual Loss)	**21.72**	**0.721**

As shown in Table 1, the proposed model beats both baseline approaches by delivering the highest SSIM and PSNR. Qualitative assessments also endorse these findings—baseline models rendered artifacts, blurry textures, and duplicated patterns, while the proposed model came up with better and more varied images. These gains indicate that adding both perceptual loss and minibatch discrimination makes a substantial difference in quality and diversity of scenes generated, improving the model for use in autonomous driving simulation tasks.

4.4.1 Comparison with Recent GAN Frameworks

To position the proposed approach within the broader GAN landscape, it is informative to compare it with more advanced models. WGAN-GP [13] addresses mode collapse using a gradient penalty, providing stable convergence but at increased computational cost. CycleGAN [10] is highly effective for domain translation tasks but is not directly suitable for image generation from noise. StyleGAN2 [19], while capable of generating highly detailed and diverse images, is often over-parameterized for simulation-specific tasks.

In contrast, our proposed DCGAN-based model, enhanced with perceptual loss and minibatch discrimination, achieves a strong trade-off between output quality, training stability, and computational efficiency, making it more suited to autonomous vehicle simulation scenarios that require high realism, variation, and tractability.

5 Conclusion and Future Work

The proposed GAN model effectively addresses mode collapse and limited diversity in image synthesis by incorporating *minibatch discrimination* and a *perceptual loss* module. These architectural and training enhancements enable the Generator to produce a broader range of city driving scenarios with improved structural consistency, reduced artifacts, and enhanced realism. Both quantitative metrics such as PSNR and SSIM and qualitative visual comparisons demonstrate the superiority of the proposed model over baseline DCGAN configurations. This capability to generate diverse, high-quality synthetic images holds significant potential for augmenting datasets used in autonomous vehicle simulation and training. Future work will explore *conditional generation* to control scene properties and extend the model to synthesize *temporally coherent video sequences*. Further improvements in output quality and realism will involve evaluating the model using metrics such as FID and investigating *progressive growing* strategies. These advancements aim to bring the model closer to real-world

deployment for training perception modules in autonomous systems and enhancing synthetic datasets for robust urban scene understanding.

References

[1] Malik, S., Khan, M.A., El-Sayed, H.: Carla: Car learning to act — an inside out. Procedia Computer Science **198**, 742–749 (2022) https://doi.org/10.1016/j.procs.2021.12.316 . 12th International Conference on Emerging Ubiquitous Systems and Pervasive Networks / 11th International Conference on Current and Future Trends of Information and Communication Technologies in Healthcare

[2] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. In: Advances in Neural Information Processing Systems (2014)

[3] Meshram, K., Jadhav, H., Narsale, N., Raut, R., Devkar, A.: Image generation from random noise using generative adversarial networks. In: 2023 International Conference on Computational Intelligence and Sustainable Engineering Solutions (CISES), pp. 381–385 (2023). https://doi.org/10.1109/CISES58720.2023.10183425

[4] Shiragudikar, S.K., Bharamagoudar, G., Manohara, K.K., *et al.*: Insight analysis of deep learning and a conventional standardized evaluation system for assessing rice crop's susceptibility to salt stress during the seedling stage. SN Computer Science **4**, 262 (2023)

[5] Shiragudikar, S.K., Bharamagoudar, G., K., M.K., Y., M.S., Totad, G.S.: Predicting salinity resistance of rice at the seedling stage: An evaluation of transfer learning methods. In: Intelligent Systems in Computing and Communication (ISCComm 2023), CCIS, Vol. 2231. Springer, ??? (2025)

[6] Shiragudikar, S.K., Bharamagoudar, G.: Enhancing rice crop resilience: Leveraging image processing techniques in deep learning models to predict salinity stress of rice during the seedling stage. International Journal of Intelligent Systems and Applications in Engineering **12**(14s), 116–124 (2024)

[7] Malathi, S.Y., Bharamagoudar, G.R., Shiragudikar, S.K.: Diagnosing and grading knee osteoarthritis from x-ray images using deep neural angular extreme learning machine. Proceedings of the Indian National Science Academy **91**, 95–108 (2025)

[8] Malathi, S., Bharamagoudar, G., Shiragudikar, S.K., Totad, G.S.: Predictive models for the early diagnosis and prognosis of knee osteoarthritis using deep learning techniques. In: Intelligent Systems in Computing and Communication (ISCComm 2023), CCIS, Vol. 2231. Springer, ??? (2025)

[9] Yang, K.: Deep Learning - DCGAN (Deep Convolutional Generative Adversarial Network). https://medium.com/@kyang3200/

deep-learning-dcgan-deep-convolutional-generative-adversarial-network-882624fdefe3. Accessed: 2025-05-01 (2020). https://medium.com/@kyang3200/deep-learning-dcgan-deep-convolutional-generative-adversarial-network-882624fdefe3

[10] Zhu, J.-Y., Park, T., Isola, P., Efros, A.A.: Unpaired image-to-image translation using cycle-consistent adversarial networks. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 2223–2232 (2017)

[11] Kupyn, O., Budzan, V., Mykhailych, M., Mishkin, D., Matas, J.: Deblurgan: Blind motion deblurring using conditional adversarial networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 8183–8192 (2018)

[12] Salimans, T., Goodfellow, I., Zaremba, W., Cheung, V., Radford, A., Chen, X.: Improved techniques for training gans. In: Advances in Neural Information Processing Systems (NeurIPS), vol. 29, pp. 2234–2242 (2016)

[13] Arjovsky, M., Chintala, S., Bottou, L.: Wasserstein gan. arXiv preprint arXiv:1701.07875 (2017)

[14] Dosovitskiy, A., Ros, G., Codevilla, F., Lopez, A., Koltun, V.: Carla: An open urban driving simulator. In: Conference on Robot Learning, pp. 1–16 (2017)

[15] Xu, W., Souly, N., Brahma, S.P.: Reliability of gan generated data to train and validate perception systems for autonomous vehicles. In: Proceedings of the IEEE Winter Conference on Applications of Computer Vision Workshops (WACVW) (2021)

[16] Sankar, V.: A Survey of GAN Applications for Enhancing Autonomous Vehicle Perception in Adverse Conditions. Available at https://www.researchgate.net/publication/385736095 (2022)

[17] Huang, J., Zhou, X., Lu, B.: Yes, We GAN: Applying Adversarial Techniques for Autonomous Driving. Available at https://www.researchgate.net/publication/336117190 (2019)

[18] Radford, A., Metz, L., Chintala, S.: Unsupervised representation learning with deep convolutional generative adversarial networks. arXiv preprint arXiv:1511.06434 (2015)

[19] Karras, T., Laine, S., Aila, T.: Analyzing and improving the image quality of stylegan. In: Proc. CVPR, pp. 8110–8119 (2020)