Synthetic data for deep learning in object detection tasks

Kostiantyn Dergachov[1,*,†], Oleksandr Aizatskyi[1,†]

[1]National Aerospace University "Kharkiv Aviation Institute", Vadym Manko Str., 17, Kharkiv, 61070, Ukraine

Abstract

The development of robust computer vision models is frequently constrained by the scarcity of precisely annotated datasets, high labeling costs, and strict regulatory limitations on data collection. Real-world datasets often lack diversity regarding critical edge cases and night scenarios, thereby limiting model reliability. To address these challenges, this paper proposes an iterative approach to generating synthetic datasets using the Unity engine, specifically tailored for vehicle detection tasks. We developed and evaluated three generations of synthetic datasets, systematically reducing the Domain Gaps. The performance of YOLOv8 models trained on these datasets was evaluated against the VisDrone real-world baseline using Synthetic-only and Fine-tuning strategies. Experimental results demonstrate that while pure synthetic data yields lower mAP compared to large-scale real datasets, it achieves superior F1-scores at high confidence thresholds. Crucially, fine-tuning experiments reveal that improving synthetic data quality significantly enhances data efficiency. A model pre-trained on the best iteration synthetic dataset and fine-tuned on just 50 real images achieved accuracy comparable to a model trained on earlier synthetic iterations requiring 400 real images. These findings suggest that high-quality synthetic data serves as a critical force multiplier in data-scarce environments, enabling the rapid and cost-effective deployment of detection systems.

Keywords

Object detection, synthetic data generation, real-time processing, unity engine, yolo, domain gap, fine-tuning, data efficiency, visdrone, deep learning

1. Introduction

The application of computer vision tasks is experiencing a substantial surge in popularity. However, developing robust models is often hindered by a lack of available and precisely annotated datasets. The process of data collection and manual labeling is not only expensive and labor-intensive, but also frequently leads to annotation inaccuracies [1]. For example, the manual creation of masks for semantic segmentation can require up to 90 minutes of work per frame [2]. An important challenge in this field is the problem of edge cases. Rare but essential scenarios, such as unexpected road obstacles, which are difficult to capture in real-world conditions. Models that are not trained on such edge cases remain unreliable [3, 4]. Furthermore, creating aviation datasets is an extremely complex task, influenced by numerous factors and constraints. These include regulatory restrictions, such as no-fly zones and safety limitations associated with aerial imaging [5]. Additionally, privacy laws, such as the GDPR, strictly limit the collection and use of personal data [6]. The cost of data collection can also increase significantly, depending on the complexity of the target task.

In this context, the importance of synthetic data cannot be overstated. It facilitates the scaling of datasets, provides precise control over rare and critical scenarios, and resolves privacy concerns. Using synthetic data considerably reduces costs and accelerates the development process thanks to automatic labeling and rapid iteration [2]. Moreover, it enables the generation of complex situations that are impossible to reproduce under real-world conditions [3].

WDA'26: International Workshop on Data Analytics, January 26, 2026, Kyiv, Ukraine

*Corresponding author.

†These authors contributed equally.

k.dergachov@khai.edu (K. Dergachov); o.m.aizatskyi@khai.edu (O. Aizatskyi)

0000-0002-6939-3100 (K. Dergachov); 0009-0001-1284-2796 (O. Aizatskyi)

2. Related work

The rapid development of Unmanned Aerial Vehicles (UAVs) has expanded their applications from simple monitoring to complex autonomous operations. Key tasks, including automatic landing systems [7], visual positioning in environments without external navigation signals [8, 9, 10], require highly reliable computer vision algorithms. The breakthrough in deep convolution neural networks, driven by the success of ImageNet-based models, has revolutionized object detection [11, 12]. Real-time algorithms, particularly the YOLO family [13, 14], have become the UAV applications standard due to their speed and efficiency. To address data scarcity, computer vision specialists are increasingly turning to synthetic data. Large-scale synthetic datasets such as SYNTHIA and Synscapes have proven effective for semantic segmentation in autonomous driving systems [15, 16, 17, 18]. Nevertheless, training on synthetic data leads to a Reality Gap, a discrepancy between simulated and real-world distributions [19, 12]. To overcome this gap, methods such as Domain Randomization and Structured Domain Randomization have been developed by varying environmental parameters during training [12, 14, 20].

Recent research in aerial imaging has demonstrated the potential of synthetic data for a wide range of tasks [21, 22]. These applications range from drone detection using purely synthetic datasets and semantic segmentation of the environment from a UAV perspective to the creation of complex multi-modal datasets for urban environments [1, 2, 23]. In addition, statistical analysis of the training data composition and reliability benchmarks confirm that using synthetic data greatly improves the performance of object detection models [19, 24]. Building on existing modeling tool capabilities, our work implements an iterative generation pipeline using the Unity Perception package [3] and focus on improving the quality of synthetic data to mitigate Domain Gaps while simultaneously identifying the optimal fine-tuning strategy. This dual approach aims to minimize reliance on costly real-world annotations while maintaining high detection accuracy.

3. Research methodology

This study focuses on the use of synthetic data to solve complex computer vision problems for UAVs applications. The main objectives are to determine the optimal training strategy for neural networks using synthetic data and to evaluate the comparative effectiveness of different data generation approaches.

3.1. VisDrone dataset

The VisDrone dataset, a comprehensive collection of images captured by UAV, was chosen as a baseline for object detection tasks. Although the standard VisDrone 2019-DET dataset contains 10 object classes, preliminary training with the YOLO v8 model indicated that the "Car" class exhibits the highest instance representativeness and the best F1 score performance, see Figure 1. Consequently, this study focuses on single class detection for the "Car"' category. The dataset was filtered to include only images containing this specific class, resulting in the following distribution: 5244 images for training, 652 for validation, and 1005 for testing.

3.2. Evaluation strategy

The experimental task was defined as a single-class detection of the "Car" category. In accordance with the official VisDrone evaluation protocol, special ignore regions, labeled as class 0, were retained. Predictions overlapping these regions were excluded from the scoring process to ensure the accuracy of metric calculations. The YOLOv8n model was used for all comparative experiments, training parameters were set to 100 epochs with an input image size of 640x640 pixels.

Three different training modes were evaluated:

- Real data (The model was trained exclusively on the real VisDrone dataset)
- Synthetic data (The models were trained exclusively on generated synthetic data)

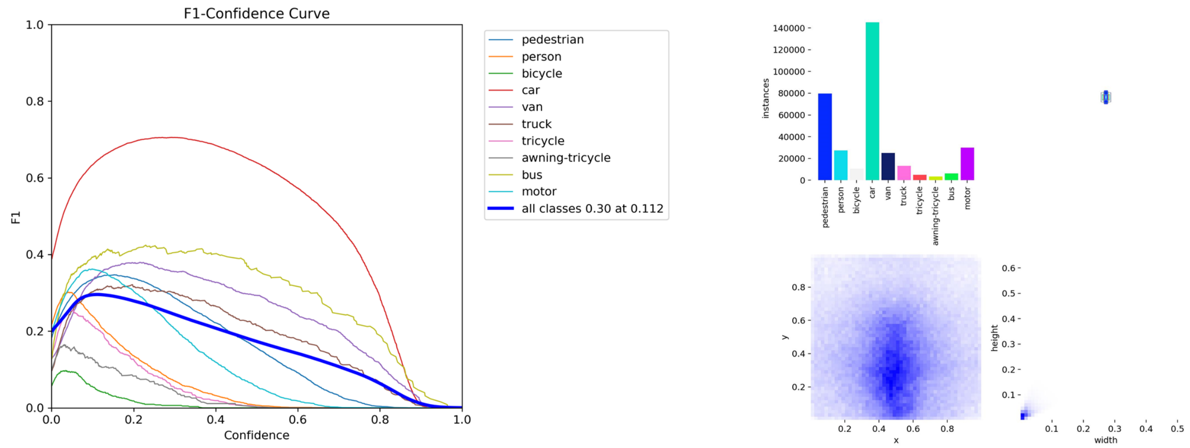

Figure 1: Class statistics of the original VisDrone 2019-DET dataset.

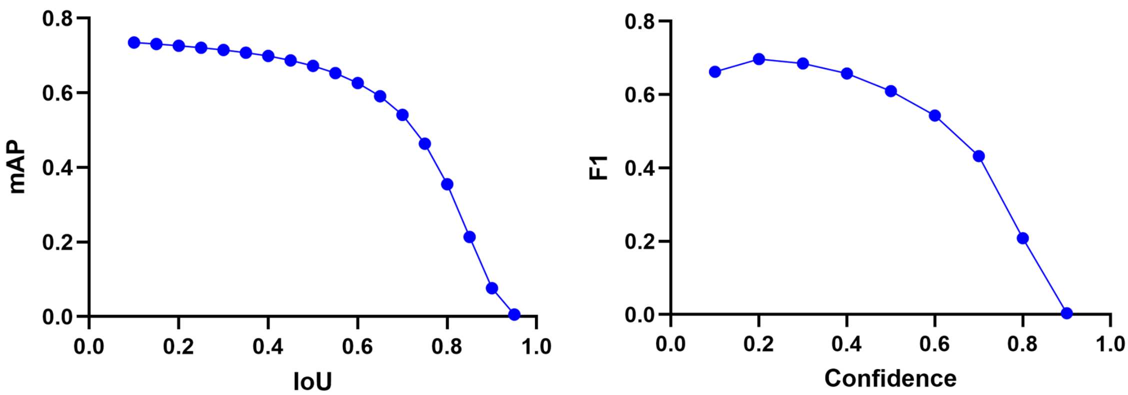

Figure 2: mAP vs IoU and F1 score vs Confidence curves for the network trained on real data.

- Synthetic + Real data with Fine-Tuning (The models were pre-trained on synthetic data and subsequently fine-tuned on the real dataset)

Performance was evaluated using the following metrics for each frame across all confidence levels and IoU thresholds:

- True Positives (TP), False Positives (FP), and False Negatives (FN)
- Precision, Recall, and F1-score
- Mean Average Precision (mAP) ranging from IoU 0.1 to 0.95

The baseline model trained on the real dataset achieved mAP@0.5 of 0.6723. The dependence of mAP on IoU and F1 score on Confidence for this baseline network are presented in Figure 2.

3.3. Synthetic dataset generation

To generate the synthetic datasets, virtual environments were divided into individual scenes representing different locations, environmental conditions, and day/night cycles. The Unity Perception Package was utilized to automatically generate precise annotations for objects in each frame. To achieve a high degree of realism, the generation pipeline incorporated an integrated system for weather, lighting, and sky simulation, which allows for specific date and time settings to replicate real-world environmental conditions. Furthermore, the scenes were complemented by a traffic system, simulating the behavior of both vehicles and pedestrians. Finally, the High Definition Render Pipeline (HDRP) was employed for advanced post-processing to enhance the visual fidelity of the generated images.

Figure 3: Examples of scenes generated in Unity.

4. Experiments and results

The development process was structured around short iterative cycles, consisting of dataset generation, model testing, and subsequent refinement of the generation pipeline. The experimental phase included three distinct iterations:

- Iteration 1, consisted of 6282 images (640x640), utilizing 34 3D car models and 15 other transport models
- Iteration 2, expanded to 9158 images with increased resolution (1400x788), utilizing 63 car models and 17 other transport models
- Iteration 3, comprised 6289 images (1400x788), utilizing 84 car models and 17 other transport models

4.1. Iteration 1 evaluation results

The first iteration served as a rapid test designed to identify the fundamental challenges of the synthetic approach. Testing revealed initially low performance metrics, with the synthetic model achieving an mAP@0.50 of 0.2367, compared to 0.6723 for the baseline real-data model. However, as illustrated in Figure 4, the primary performance discrepancy was observed in the Recall metric, while Precision remained relatively comparable to the baseline.

Detailed analysis identified several critical detection issues. First, the camera angle in synthetic scenes was often directed vertically downward, which differed from the intended UAV perspective. Second, there was a significant domain gap regarding vehicle colors; while the synthetic dataset utilized high-contrast colors (white, black, red, blue), most real-world vehicles appeared in shades of gray, dark brown, or black. Additionally, the model struggled with detection in chaotic parking arrangements,

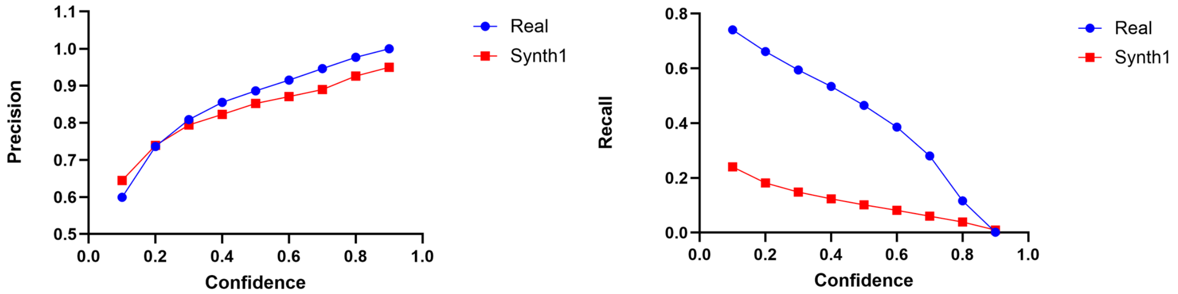

Figure 4: Comparison of Precision-Confidence and Recall-Confidence curves for networks trained on real data with the first synthetic dataset.

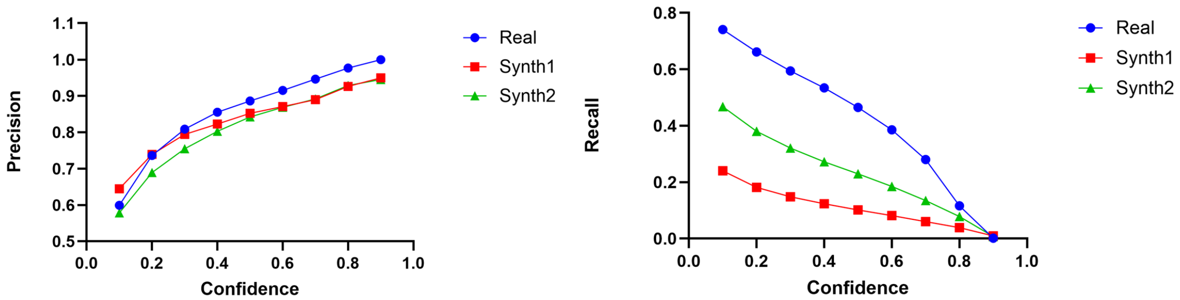

Figure 5: Comparison of Precision and Recall curves for networks trained on real data, Iteration 1, and Iteration 2 synthetic datasets.

identifying small objects from high altitudes, and recognizing local specific vehicles such as taxis and police cars, which possessed distinct color schemes and models not present in the training set.

4.2. Iteration 2 evaluation results

In the second iteration, the image resolution was adjusted to match the VisDrone standard, and the dataset size was increased. Extensive improvements were made to bridge domain gaps. To bridge the appearance gap, models of local transport were added, the vehicle color palette was expanded to include missing color tones, and realistic headlight illumination was implemented for night scenes. To address the content gap, the environment was enriched with parking lots, new locations featuring trees and occlusions, and Bezier paths were implemented to simulate realistic drone flight patterns across three different camera angle ranges.

Testing of the second dataset demonstrated improved results. The mAP@0.50 rose to 0.402, compared to 0.2367 in the first iteration and 0.6723 as baseline. As shown in Figure 5, while Recall improved, it remained the primary limiting factor.

Despite the improvements, new detection challenges emerged. A serious issue was the discrepancy between amodal and modal annotations. Unity annotates only the visible portion of a vehicle behind occlusion (modal), while VisDrone's annotations cover the entire object (amodal). This discrepancy resulted in the model simultaneously producing both false positives and false negatives due to bounding box mismatches. Besides, the model struggled with vehicles with sunroofs or panoramic roofs, as well as a lack of examples with vehicles occluded by trees or with multiple occlusions in traffic jams. A major problem was the limitation of rendering high-density traffic at night due to real-time performance limitations.

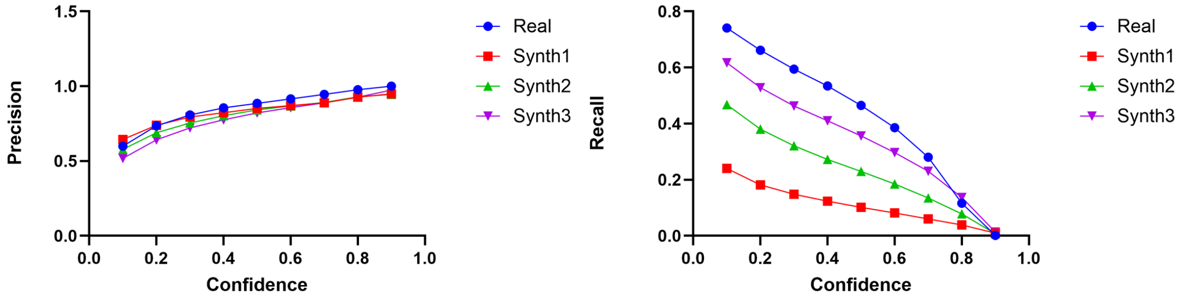

Figure 6: Comparison of Precision and Recall curves for networks trained on real data versus networks trained on the first, second, and third synthetic datasets.

4.3. Iteration 3 evaluation results

Before the third iteration, an analysis of the second synthetic dataset was performed by splitting it into individual scenes and training small neural networks on each scene separately. This testing revealed that increasing the number of images without diversifying locations and camera angles yielded diminishing returns. Specifically, it was found that 700-1000 images per location are sufficient for our conditions, generating data beyond this threshold without adding diversity does not improve performance. Analysis also identified that certain scenes in the second iteration were ineffective. Consequently, for the third dataset, 2550 images were retained from the second iteration, while four scenes were redesigned, and 3739 newly generated images were added. A primary challenge involved night scenes. While realism was considerably improved in the second iteration, the Unity engine imposes real-time limits on the number of light sources per tile. Simulating night scenes with high object density and traffic jams required a compromise between lighting realism and performance, which ultimately yielded positive results.

Testing of the third dataset demonstrated further improvements. The mAP@0.50 reached 0.4915, compared to 0.402 in iteration 2, 0.2367 in iteration 1, and real base 0.6723. Notably, as illustrated in Figure 6, at high confidence thresholds, the F1-score of the synthetic network outperforms that of the real-data model. This suggests that synthetic annotations can be extremely useful in tasks requiring high detection confidence.

4.4. Image generation benchmarks

A consistent performance increase was observed with each iteration, but the rate of improvement notably slowed, as shown in Figure 7. Conversely, the effort and resources required to generate each subsequent iteration increased substantially. It remains challenging to create synthetic data that fully outperforms or matches a large real-world dataset across all metrics. On the other hand, synthetic data of varying quality levels may be suitable for different tasks. Figure 8 shows detection results on the same test frame using networks from different iterations. Crucially, in specific operational ranges, such as high confidence detection (Confidence > 0.8), high-quality pure synthetic data can outperform models trained on manually labeled real data, as illustrated in Figure 7.

4.5. Fine-tune, synthetic to real training

This section examines the effectiveness of knowledge transfer from synthetic models to real-world domains using small subsets of real data. Experiments focused on methods for selecting real-world images for fine-tuning and analyzing the impact of synthetic data quality on the size of the required real-world dataset.

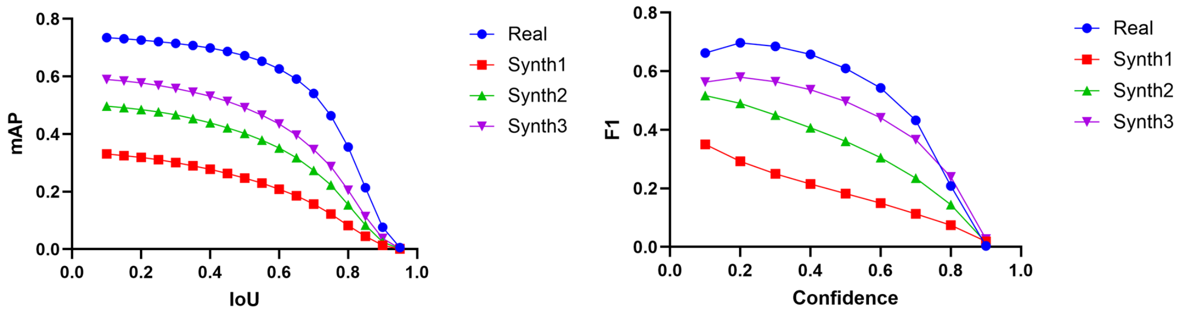

Figure 7: Comparison of mAP with IoU and F1-score with Confidence curves for networks trained on real data and networks trained on the first, second, and third synthetic datasets.

Figure 8: Object detection results on a test image for networks trained on synthetic data from the first, second, and third iterations (left to right). Legend: TP (green box), FP (yellow box), FN (red box).

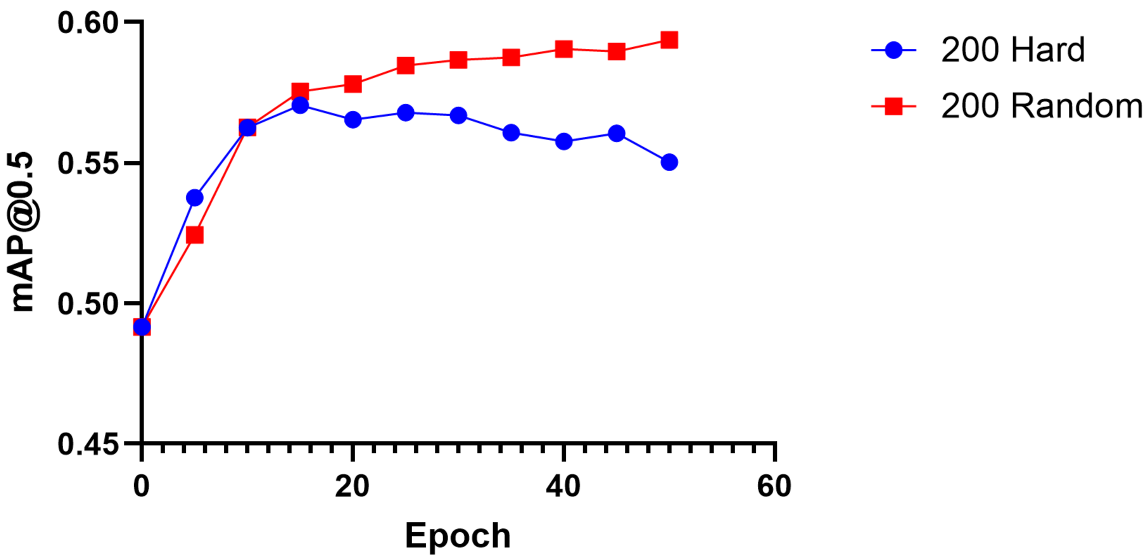

Figure 9: Comparison of mAP with Training Epoch curves for image subsets selected randomly versus by difficulty.

4.6. Fine-tune, real images acquisition

To determine the optimal strategy for selecting real data, we compared two sampling methods using a subset of 200 images from the real dataset. The first set (200Hard), consisted of the 200 most difficult frames where the purely synthetic model failed. The second set (200Random), was a uniform random selection proportional to all scenes in the training set. Experimental results indicated that a uniform random distribution is superior for fine-tuning, as illustrated in Figure 9. Based on this finding, we generated randomized subsets of 50, 100, 200, and 400 real images for subsequent experiments.

Table 1
mAP Results for Synthetic Networks (Iteration 1, 2, 3) Fine-tuned on Real Dataset Subsets of Varying Sizes

Fine-Tune Dataset Size	Iteration 1 mAP	Iteration 2 mAP	Iteration 3 mAP
50	0.4553	0.5515	0.5636
50	0.5025	0.5622	0.5864
50	0.5348	0.5914	0.6121
50	0.5722	0.6113	0.6266

4.7. Fine-tune Results

Following extensive experimentation with hyperparameter tuning, a two-stage training protocol was established. In the first stage, the first 10 layers of the network are frozen, and the model is trained for 20 epochs with an initial learning rate of lr0=0.01 and a final learning rate factor of lrf=0.01. In the second stage, all layers are unfrozen, and training continues for an additional 40 epochs with a reduced initial learning rate of lr0=0.001 and a final factor of lrf=0.1. The fine-tuning results for networks pre-trained on synthetic datasets (Iterations 1, 2, and 3) using varying amounts of real data are presented in Table 1.

The experimental results highlight a substantial improvement in data processing efficiency driven by the quality of synthetic data. Specifically, the model pre-trained on the high-quality synthetic dataset (Iteration 3) and refined with only 50 real images achieved performance metrics comparable to the weaker model from Iteration 1 refined with 400 real images (0.5636 versus 0.5722). This indicates that improvements in synthetic data generation reduced the need for annotated real-world data by nearly 8 times to achieve comparable accuracy. Additionally, it was observed that high-quality synthetic data reaches a performance saturation point much earlier.

Conclusions

In relatively simple object detection tasks, synthetic datasets can produce results close to those achieved with real data [1]. However, in particularly complex environmental conditions, creating synthetic data of sufficient quality to match the efficacy of a model trained solely on real data remains a formidable challenge [2]. Nevertheless, our results indicate that at high confidence thresholds, synthetic data can produce excellent F1-scores even in challenging scenarios. Creating synthetic image datasets using 3D engines requires a constant trade-off between minimizing the content gap and the appearance gap [8]. While it is often argued that the content gap has a greater impact than the appearance gap, and that the primary goal of synthetic data is maximum diversity rather than photorealism [6], our experience shows that certain visual attributes are crucial. Without the necessary vehicle colors and configurations, and without active light sources on vehicles in night scenes, detection fails. Simultaneously, real-time rendering constraints prevent the simulation of hundreds of moving vehicles at night with full dynamic lighting. Therefore, technical compromises were required, such as reducing the number of light sources in streetlights and surrounding objects, and implementing Level of Detail (LOD) systems for long-range lighting.

Ultimately, the addition of a small amount of real data significantly boosts performance metrics. High-quality synthetic data proves to be most critical specifically in conditions of severe data scarcity, acting as a powerful multiplier for limited real-world annotations.

Declaration on generative AI

The authors have not employed any Generative AI tools.

References

[1] M. Wisniewski, Z. A. Rana, I. Petrunin, A. Holt, S. Harman, Drone detection using deep neural networks trained on pure synthetic data, arXiv preprint arXiv:2411.09077 (2024).

[2] C. Hinniger, J. Rüter, Synthetic training data for semantic segmentation of the environment from uav perspective, Aerospace 10 (2023) 604. doi:10.3390/aerospace10070604.

[3] S. Borkman, A. Crespi, S. Dhakad, S. Ganguly, J. Hogins, Y.-C. Jhang, M. Kamalzadeh, B. Li, S. Leal, P. Parisi, C. Romero, W. Smith, A. Thaman, S. Warren, N. Yadav, Unity perception: Generate synthetic data for computer vision, arXiv preprint arXiv:2107.04259 (2021).

[4] T. Nikitina, B. Kuznetsov, Y. Averyanova, O. Sushchenko, I. Ostroumov, N. Kuzmenko, M. Zaliskyi, O. Solomentsev, N. Ruzhentsev, O. Havrylenko, A. Popov, V. Volosyuk, O. Shmatko, S. Zhyla, V. Pavlikov, E. Tserne, Method for design of magnetic field active silencing system based on robust meta model, in: S. Shukla, H. Sayama, J. V. Kureethara, D. K. Mishra (Eds.), Data Science and Security, Springer Nature Singapore, Singapore, 2024, pp. 103–111.

[5] I. Ostroumov, O. Ivashchuk, N. Kuzmenko, Preliminary estimation of war impact in ukraine on the global air transportation, in: 2022 12th International Conference on Advanced Computer Information Technologies (ACIT), 2022, pp. 281–284. doi:10.1109/ACIT54803.2022.9913092.

[6] B. N. Jacobsen, Machine learning and the politics of synthetic data, Big Data & Society 10 (2023). doi:10.1177/2053951722114537.

[7] K. Dergachov, S. Bahinskii, I. Piavka, The algorithm of uav automatic landing system using computer vision, in: 2020 IEEE 11th International Conference on Dependable Systems, Services and Technologies (DESSERT), IEEE, 2020, pp. 247–252. doi:10.1109/DESSERT50317.2020.9124998.

[8] O. Hurtovyi, K. Dergachov, A. Yaremenko, Visual positioning of the uav in conditions of impossibility of using external navigation signals, in: I. Ostroumov, K. Marais, M. Zaliskyi (Eds.), Advances in Civil Aviation Systems Development (ACASD 2025), volume 1418 of *Lecture Notes in Networks and Systems*, Springer, Cham, 2025. doi:10.1007/978-3-031-91992-3_36.

[9] V. Pavlikov, V. Volosyuk, S. Zhyla, K. Dergachov, O. Havrylenko, Y. Averyanova, A. Popov, I. Ostroumov, N. Ruzhentsev, E. Tserne, T. Nikitina, O. Shmatko, N. Kuzmenko, O. Sushchenko, M. Zaliskyi, O. Solomentsev, B. Kuznetsov, Optimal algorithm of sar raw data processing for radar cross section estimation, in: R. Voliansky, O. Solomentsev, N. Slimani, K.and Kuzmenko (Eds.), Ceur Workshop Proceedings, volume 3981, CEUR-WS, 2025, pp. 170–181. URL: https://ceur-ws.org/Vol-3981/paper15.pdf.

[10] R. Alghonaim, E. Johns, Benchmarking domain randomisation for visual sim-to-real transfer, in: Proceedings of the International Conference on Robotics and Automation (ICRA), IEEE, 2021.

[11] A. Krizhevsky, I. Sutskever, G. E. Hinton, Imagenet classification with deep convolutional neural networks, Communications of the ACM 60 (2017) 84–90.

[12] A. Prakash, S. Boochoon, M. Brophy, D. Acuna, E. Cameracci, G. State, Structured domain randomization: Bridging the reality gap by context-aware synthetic data, in: Proceedings of the International Conference on Robotics and Automation (ICRA), IEEE, Montreal, Canada, 2019.

[13] J. Redmon, S. Divvala, R. Girshick, A. Farhadi, You only look once: Unified, real-time object detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, 2016, pp. 779–788.

[14] J. Tremblay, A. Prakash, D. Acuna, M. Brophy, V. Jampani, C. Anil, T. To, E. Cameracci, S. Boochoon, S. Birchfield, Training deep networks with synthetic data: Bridging the reality gap by domain randomization, in: Proceedings of the CVPR 2018 Workshop on Autonomous Driving, IEEE, Salt Lake City, UT, USA, 2018.

[15] G. Ros, L. Sellart, J. Materzynska, D. Vazquez, A. M. Lopez, The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes, in: Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, Las Vegas, NV, USA, 2016, pp. 3234–3243.

[16] M. Wrenninge, J. Unger, Synscapes: A photorealistic synthetic dataset for street scene parsing,

arXiv preprint arXiv:1810.08705 (2018). doi:10.48550/ARXIV.1810.08705.

[17] K. Konen, T. Hecking, Increased robustness of object detection on aerial image datasets using simulated imagery, in: Proceedings of the 2021 IEEE Fourth International Conference on Artificial Intelligence and Knowledge Engineering (AIKE), IEEE, Laguna Hills, CA, USA, 2021, pp. 1–8.

[18] Q. Li, L. Mou, Q. Xu, Y. Zhang, X. X. Zhu, R3-net: A deep network for multi-oriented vehicle detection in aerial images and videos, IEEE Transactions on Geoscience and Remote Sensing 57 (2019) 5028–5042. doi:10.1109/TGRS.2019.2895697.

[19] M. Krump, P. Stütz, Deep learning based vehicle detection on real and synthetic aerial images: Training data composition and statistical influence analysis, Sensors 23 (2023) 3769. doi:10.3390/s23073769.

[20] N. Morrical, J. Tremblay, Y. Lin, S. Tyree, S. Birchfield, V. Pascucci, I. Wald, Nvisii: A scriptable tool for photorealistic image generation, arXiv preprint arXiv:2105.13962 (2021) 9.

[21] E. Kolve, R. Mottaghi, W. Han, E. VanderBilt, L. Weihs, A. Herrasti, M. Deitke, K. Ehsani, D. Gordon, Y. Zhu, A. Kembhavi, A. Gupta, A. Farhadi, Ai2-thor: An interactive 3d environment for visual ai, arXiv preprint arXiv:1712.05474 (2019).

[22] T. Nikitina, B. Kuznetsov, N. Ruzhentsev, O. Havrylenko, V. Volosyuk, O. Shmatko, A. Popov, S. Zhyla, V. Pavlikov, E. Tserne, Y. Averyanova, O. Sushchenko, M. Zaliskyi, O. Solomentsev, I. Ostroumov, N. Kuzmenko, Algorithm of robust control for multi-stand rolling mill strip based on stochastic multi-swarm multi-agent optimization, in: S. Shukla, H. Sayama, J. V. Kureethara, D. K. Mishra (Eds.), Data Science and Security, Springer Nature Singapore, Singapore, 2024, pp. 247–255.

[23] G. Rizzoli, F. Barbato, M. Caligiuri, P. Zanuttigh, Syndrone - multi-modal uav dataset for urban scenarios, in: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops, 2023. doi:10.48550/arXiv.2308.10491.

[24] H. Lechgar, H. Bekkar, H. Rhinane, Detection of cities vehicle fleet using yolo v2 and aerial images, Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci. 42 (2019) 121–126. doi:10.5194/isprs-archives-XLII-4-W12-121-2019.