# A synthetic data approach for object detection in super low-resolution images

Mosayeb Davoudi, Kashkoli
University of the West of England Bristol, UK

Asad, JAvied
University of the West of England Bristol, UK

Ari Yair, Barrera-Animas
Universidad Panamericana, México

Juan Manuel, Davila Delgado*
Birmingham City University, UK

## ABSTRACT

This paper presents a synthetic data approach to train object detection models to address the challenges with object detection in super low-resolution images. With a particular emphasis on person detection, the study uses 28 photorealistic 3D models of individuals, optimised for efficient rendering and minimal memory consumption. These models are seamlessly integrated into a 3D terrain model, mimicking diverse real-world situations. To ensure scalability and diversity, the methodology incorporates domain randomisation techniques, encompassing variations in factors like lighting conditions, seasonal effects, camera angles, lens specifications, and different image resolutions. The process of dataset generation is automated through a Python script in Blender, offering systematic scene configuration and camera positioning. The dataset created consists of 10,560 images across four resolutions. The evaluation was carried out using popular object detection algorithms, including Faster RCNN and RetinaNet, within the Detectron2 framework. Results highlight the effectiveness of synthetic datasets in training and testing object detection algorithms, showcasing visual comparisons, Average Precision (AP) metrics, and training performance statistics. Notably, RetinaNet outperforms Faster RCNN, achieving higher accuracy. This research offers invaluable insights into synthetic dataset generation and its application for object detection in low-resolution images.

## CCS CONCEPTS

• **Machine learning algorithms**;

## KEYWORDS

Computer Vision, Object detection, Synthetic approaches

**ACM Reference Format:**
Mosayeb Davoudi, Kashkoli, Asad, JAvied, Ari Yair, Barrera-Animas, and Juan Manuel, Davila Delgado*. 2024. A synthetic data approach for object detection in super low-resolution images. In *2024 the 8th International Conference on Innovation in Artificial Intelligence (ICIAI 2024), March 16–18, 2024, Tokyo, Japan.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3655497.3655502



*ICIAI 2024, March 16–18, 2024, Tokyo, Japan*

ACM ISBN 979-8-4007-0930-2/24/03
https://doi.org/10.1145/3655497.3655502

## 1 INTRODUCTION

Object detection powered by synthetic data presents a scalable, controlled, and versatile approach for training and evaluating computer vision models across a wide range of applications, from autonomous vehicles and robotics to security and healthcare [1]. In the realm of synthetic data for object detection, this research outlines a versatile methodology applicable to a range of object detection scenarios.

In the realm of computer vision and machine learning, the availability of high-quality, labelled datasets holds paramount significance in the development and assessment of object detection algorithms. Nevertheless, obtaining such datasets, particularly for specialised applications involving niche scenarios, poses formidable challenges in terms of resources and effort. This is especially true in the domain of person detection, where a multitude of settings and conditions must be considered to ensure robust performance and generalisation. Synthetic datasets have emerged as a potent alternative to traditional data collection techniques, offering a spectrum of advantages such as flexibility, scalability, and precise control over data generation. In this research study, an approach is presented for the creation and utilisation of synthetic datasets tailored explicitly for object detection in low-resolution images, with a keen focus on person detection applications [2].

The foundation of this approach is grounded in the adept application of 3D modelling and rendering techniques [3]. A curated collection of 3D models comprising 28 intricately designed individual representations. These models exhibit a wide array of attributes, encompassing photorealistic textures and a diverse spectrum of body poses. What sets them apart is their efficiency in rendering and minimal memory consumption, rendering them amenable to large-scale dataset generation. To foster the creation of a realistic and diverse testing environment, these 3D models are seamlessly integrated into a 3D terrain model replicating a residential site. This amalgamation empowers us to simulate real-world scenarios, encapsulating the intricacies and variations encountered in person detection applications. Beyond individual representations, this synthetic environment encompasses common residential objects like transmission towers and existing structures, further enhancing the authenticity of the synthetic datasets.

Ensuring that the synthetic data accurately mirrors real-world proportions stands as a pivotal facet of our methodology. To accomplish this, the judicious employment of Blender's 'Dimensions' feature allows the scaling of all 3D models according to predetermined real-world units. This meticulous calibration assures that the generated datasets not only appear visually convincing but also adhere to physical accuracy. Diversity is injected into the synthetic datasets through the employment of domain randomisation techniques, encompassing variations in factors such as lighting conditions, seasonal effects, camera orientations, lens specifications, and varying resolutions. These variances are systematically documented and integrated into the datasets, providing a broad spectrum of testing and evaluation conditions.

The process of dataset generation is automated through a Python script embedded within the Blender software, facilitating the configuration of scene parameters and camera positioning. This streamlined approach ensures the reproducibility and scalability of the dataset creation process. To expedite the utility of the synthetic datasets for object detection, associated text files are generated in the Darknet (YoloV4) format. These text files encapsulate vital information, including model designations, bounding box coordinates, and category IDs, rendering them compatible with a wide array of object detection algorithms. Subsequent sections of this paper delve into the intricate details of the dataset generation methodology, the experimental setup, and the evaluation of object detection algorithms using these synthetic datasets. The results gleaned from these experiments underscore the efficacy of this approach, underscoring the considerable potential of synthetic datasets in training and validating object detection algorithms tailored to person detection scenarios.

## 2 LITERATURE REVIEW

In the ever-evolving domain of computer vision and object detection, the past few years have witnessed a remarkable surge in the development of cutting-edge algorithms and techniques. This surge owes much of its momentum to the availability of high-quality labelled datasets. However, acquiring such datasets, especially for niche applications like person detection in diverse scenarios, presents a formidable challenge, characterized by its resource-intensive nature. Synthetic datasets have emerged as a promising remedy to address these challenges and serve as catalysts for advancements in computer vision. This comprehensive literature review delves into the current landscape of research related to synthetic dataset generation and its versatile application in object detection.

### 2.1 Synthetic Datasets in Computer Vision

The use of synthetic datasets has become increasingly prominent in computer vision research. In their work, "Using Synthetic Data for Text Detection in Natural Images" [4], the authors advocate the utilisation of synthetic data for text detection in natural images, highlighting the myriad benefits of controlled data generation. Our research echoes these approaches as we harness the potential of synthetic datasets for object detection, with a particular emphasis on person detection.

### 2.2 3D Modeling and Rendering for Synthetic Datasets

The realm of 3D modeling and rendering has been instrumental in the creation of realistic synthetic datasets. In their work titled "Generating 3D Adversarial Point Clouds" [5], the authors embark on an exploration of 3D synthetic data generation for adversarial training. Our approach closely aligns with this paradigm, relying on sophisticated 3D modeling techniques to represent individuals and their surrounding environments, ensuring an exceptionally high degree of realism within the generated datasets.

### 2.3 Domain Randomization for Enhanced Diversity

Domain randomisation is a widely employed technique to inject diversity into synthetic datasets. In "Domain Randomization for Sim-to-Real Transfer of Robot Control" [6], domain randomisation takes center stage to enhance sim-to-real transfer capabilities in robotics. In our research, we apply domain randomisation to introduce variations in crucial aspects such as lighting conditions, camera settings, the time of year, and image resolution. This, in turn, significantly enriches the diversity of scenarios featured in our synthetic datasets.

### 2.4 Object Detection in Synthetic Data

The concept of using synthetic datasets for object detection has been a particular subject of exploration. In their work on "Synthetic Data for Text Detection in Natural Scenes" [7], the authors employ synthetic data for text detection, underlining the potential for models trained on synthetic data to be effectively transferred to real-world scenarios. Our research takes this notion a step further by extending it to object detection, with a specific focus on person detection scenarios.

### 2.5 Tools and Frameworks

The development of tools and frameworks designed to streamline the generation and application of synthetic datasets has been a pivotal advancement. In their work "Data Synthesis in Deep Learning for Object Detection." [8], the authors introduce a framework for data synthesis in the context of object detection. In our research, we leverage the versatile Blender software and the robust Detectron2 framework to facilitate the processes of data generation and object detection training.

In summation, the use of synthetic datasets in the realm of computer vision, particularly for object detection, has witnessed a remarkable surge in interest over recent years. Our research contributes significantly to this expanding body of knowledge by presenting a comprehensive methodology for the creation of synthetic datasets tailored to the nuances of person detection scenarios in super low-resolution images. Through our innovative use of 3D modeling, domain randomisation, and automated scripting, we not only generate diverse and realistic datasets but also evaluate the effectiveness of object detection algorithms within this context. Our work is firmly aligned with the overarching trend of harnessing synthetic data to propel the field of computer vision and advance the frontiers of object detection.

## 3 DEVELOPMENT OF THE DATASET FOR LOW-RESOLUTION DETECTION

This section details our developed method for creating synthetic datasets in a step-by-step manner and carrying out the labeling of specific assets automatically. The asset used in this research exclusively pertains to individuals, and this dataset is specifically generated for applications related to person detection. In total, a

**Table 1: Domain randomisation parameters**

| Sun times: 'clock' | Time of year: (day, month) | Rotation orientations | Cameras lens specifications |
|---|---|---|---|
| Twilight= '6:30',<br>Morning= '9:30',<br>Midday= '12',<br>Afternoon= '14',<br>Evening='16:30' | Summer solstice: June 21<br><br>Winter solstice: December 21 | North<br>Northeast<br>East<br>Southeast<br>South<br>Southwest<br>West<br>Northwest | Camera number: (1-4)<br>Camera location:<br>cameras are placed 7.5m above the ground (by default the ground has no elevation) |

collection of 28 3D models representing people were used, each characterised by several essential properties. These models feature diverse rigged figures with photo realistic textures, specifically engineered for swift rendering and minimal memory usage. Each figure comprises approximately 15,000 faces, utilising a single texture map. Moreover, all figures come pre-rigged and possess an original "A" pose as well as two casual poses, providing flexibility in their application. To further enhance their adaptability, the models include skeletons with weights/skin, enabling adjustments to pose or the creation of entirely new ones. Additionally, the incorporation of unique "namespaces" for each figure ensures seamless integration into a single scene, optimising their utility for various purposes.

Blender [9] version 2.83 was chosen for the creation of synthetic datasets due to its extensive documentation and widespread usage as a freely available open-source software package in the domains of animation, video editing, and simulation. Furthermore, its compatibility with Python scripts enables automated processes, enhancing its utility for this purpose.

To generate the datasets, 3D models of people were used in conjunction with a 3D terrain model depicting a residential site. This amalgamation was employed to generate visuals mirroring a real-life residential site scenario. Asset models were acquired from the TurboSquid marketplace. The layout of the site scenario was created using the '3D View: Blender-GIS' add-on, and subsequently refined in Blender to maintain the realistic elevation scale of the actual terrain in conjunction with the asset models. To ensure the accurate scaling of all 3D models based on real-world proportions, Blender's 'Dimensions' feature was employed. This functionality, located in the Transform panel, enables the adjustment of an object's size according to the Scene's specified units.

The approach incorporated domain randomisation to introduce diversity into the generated models. This diversity was achieved by varying factors such as sun times, time of year, rotation orientations, and camera lens specifications. These variations are detailed in Table 1. The approach to develop the dataset is based in our previous work described in literature [10].

To generate the images, four different resolutions have been selected including: (1) 832×832 pixels, refers to the original, full-resolution images, (2) 432×432 pixels, medium, (3) 208×208 pixels, low, (4) 84×84 pixels, super low. To enable the automatic creation of synthetic datasets and manage various configuration options such as lighting, camera settings, dates, and angles of asset placement in a systematic way, the suggested method is executed through a Python script. While rendering, after arranging all site assets within the 3D scene and setting up scene features, Blender's Python API is employed to determine the position and bounding box of each asset based on its placement and orientation in relation to the rendering camera's perspective.

To produce an individual text file using the Darknet (YoloV4) format for every rendered image, the connection among the model names, their respective bounding boxes, and category ID (in this study, the category "person" has the ID 0) is utilised [11]. Each text file contains solely the associations of the construction assets visible in the image.

This method enables the creation of accurately annotated rendered images and associated text files in the Darknet (YoloV4) format, suitable for utilisation in object detection algorithms. To showcase the bounding boxes for each site asset in the synthetic images, the rendered image alongside its corresponding text file can be utilised within a labeling tool or library, such as OpenCV, to visually delineate the boxes. Examples of generated images are shown in Figure 1.

## 4 EXPERIMENTS

A total of 10560 images and their associated label text files (for each resolution) are utilised for the experimentation process. After generating the synthetic dataset, two of the most common object detection algorithms and pretrained models are utilised to train and test the dataset. These are Faster RCNN which is a deep convolutional network designed for object detection, providing predicted class IDs, confidence scores, and bounding box coordinates, and RetinaNet which employs a feature pyramid network, this model efficiently detects objects at multiple scales and introduces the Focal loss function to mitigate the extreme foreground-background class imbalance.

The Synthetic dataset was trained using Detectron2, a framework developed by Facebook AI Research (FAIR) [12], known for its extensive support across various object detection algorithms. Detectron2 encompasses a range of features, including training recipes catering to object detection, instance segmentation, panoptic segmentation, semantic segmentation, and key point detection. Additionally, it offers a repository housing over 80 pre-trained models available for fine-tuning or building models from the ground up. Notably, it extends support for well-known vision datasets like COCO, Cityscapes, LVIS, PASCAL VOC, and ADE20k. The

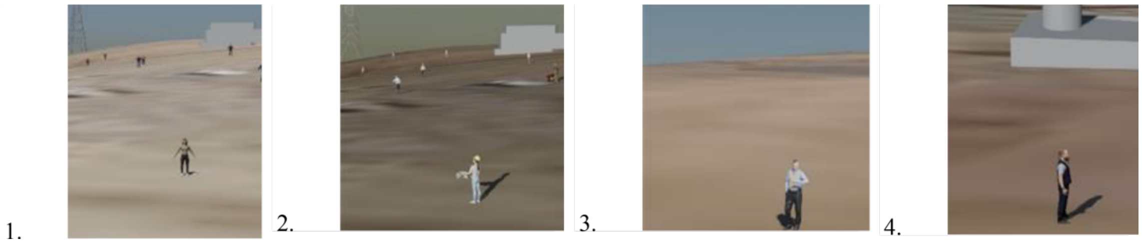

**Figure 1: Examples of generated images with different illumination parameters**

(a) Faster RCNN

(b) RetinaNet

**Figure 2: Examples of results using (a) FasterRCNN and (b) RetinaNet**

dataset itself is split into training, testing, and validation subsets, partitioned at a ratio of 90%, 5%, and 5%, respectively.

## 5 RESULTS

Visual comparisons displaying the detections conducted by the two trained detectors on the test subset of the synthetic datasets are depicted in Figure 2. Across all three figures, the algorithm-generated detections are represented by the red boxes, while the ground-truth labels, assigned by the proposed automatic label process, are depicted by the green boxes. Figure 3 illustrates the effect of varying resolutions on the inference and display of the models, which shows good accuracy in low-resolution s.


**Figure 3: Examples of detections on varying resolutions**

### 5.1 Evaluation process

To assess the effectiveness of the detection algorithms trained using the Detectron2 library, the evaluation employed the COCO Evaluator [13]. Detectron2 utilises COCO's metrics [14], including Average Precision (AP), AP50, AP75, $AP^{small}$, $AP^{medium}$, and $AP^{large}$. Conversely, the mean Average Precision ($_m$AP) metric was applied to gauge the training performance of the detector trained with the Darknet framework. This training performance was assessed using the respective validation subsets within the synthetic dataset.

Table 2 displays the Accuracy Precision (AP) achieved during the training of the Faster R-CNN and RetinaNet detectors. According to this table, the Accuracy Precision (AP) of the RetinaNet detector is higher than Faster R-CNN. When comparing just the test subsets, the synthetic dataset resulted in higher values for $\text{PASCAL}_m$AP than for COCO AP in both training models. These results show that the detectors have acceptable accuracy in detecting the person in test dataset. The training and validation curves for the two object detectors that were trained using the dataset are displayed in Figure 9.

For this evaluation process and for training the detection models, a Windows system with 8 logical processors Intel Core i7-7700K CPU @ 4.20 GHz, 32 GB of RAM, and 2 NVIDIA GeForce GTX 1080 Ti graphic cards was used to perform all the experiments.

## 6 CONCLUSIONS

Synthetic data-driven object detection not only addresses data scarcity but also enables the exploration of diverse scenarios, ensuring robust and adaptable computer vision systems. In this study, we have introduced a robust methodology for the generation and application of synthetic datasets tailored for object detection, with a specific emphasis on person detection scenarios. Our approach leverages 3D modeling and rendering techniques to create diverse, photorealistic 3D models of individuals, providing a foundation for realistic data synthesis. By amalgamating these models with 3D terrain models of residential sites, we have successfully simulated complex, real-world scenarios, enhancing the practicality of our synthetic datasets. Domain randomization techniques, encompassing variations in lighting, camera settings, and time of year, have been employed to introduce diversity and complexity into our datasets, ensuring their relevance for real-world applications. Automation through Python scripting within Blender has streamlined the dataset generation process, ensuring reproducibility and scalability. Furthermore, we have demonstrated the utility of these synthetic datasets by evaluating two common object detection algorithms, Faster RCNN and RetinaNet, on the generated data. The results underscore the effectiveness of synthetic data for training and testing object detection models, with RetinaNet outperforming Faster RCNN, and achieving high accuracy in person detection. Our work contributes to the broader trend of leveraging synthetic data in computer vision research. By providing a comprehensive framework for data generation and evaluation, we aim to advance the field of object detection, offering researchers a practical and controlled alternative to traditional data collection methods. Synthetic datasets hold promise for addressing data scarcity and privacy concerns, and we foresee their growing impact on computer vision and machine learning applications.

**Table 2: Faster R-CNN and RetinaNet performance on synthetic datasets.**

| Training Model | Faster R-CNN | | RetinaNet | |
|---|---|---|---|---|
| Performance on COCO eval. metrics | Training | Testing | Training | Testing |
| AP (%) | 96.06 | 94.8 | 97.62 | 97.70 |
| AP50 (%) | 100 | 100 | 100 | 100 |
| AP75 (%) | 100 | 100 | 100 | 100 |
| APsmall (%) | nan | nan | nan | nan |
| APmedium (%) | nan | nan | nan | nan |
| APlarge (%) | 96.05 | 94.8 | 97.64 | 97.71 |
| AR1 (%) | - | 97.01 | - | 98.50 |
| AR10 (%) | - | 97.01 | - | 98.50 |
| AR100 (%) | - | 97.01 | - | 98.50 |
| ARsmall (%) | - | nan | - | nan |
| ARmedium (%) | - | nan | - | nan |
| ARlarge (%) | - | 97.01 | - | 98.50 |
| PASCAL eval. Metrics | | | | |
| mAP (%) | - | 100 | - | 100 |

## REFERENCES

[1] Lidberg, Love. ”Object Detection using deep learning and synthetic data.” (2018).
[2] Aranjuelo, Nerea, Sara García, Estíbaliz Loyo, Luis Unzueta, and Oihana Otaegui. ”Key strategies for synthetic data generation for training intelligent systems based on people detection from omnidirectional cameras.” Computers & Electrical Engineering 92 (2021): 107105.
[3] Verma, Vishal, and Ekta Walia. ”3D Rendering-Techniques and challenges.” International Journal of Engineering and Technology 2, no. 2 (2010): 29-33.
[4] Gupta, Ankush, Andrea Vedaldi, and Andrew Zisserman. ”Synthetic data for text localisation in natural images.” In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 2315-2324. 2016.
[5] Xiang, Chong, Charles R. Qi, and Bo Li. ”Generating 3d adversarial point clouds.” In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 9136-9144. 2019.
[6] Chen, Xiaoyu, Jiachen Hu, Chi Jin, Lihong Li, and Liwei Wang. ”Understanding domain randomization for sim-to-real transfer.” arXiv preprint arXiv:2110.03239 (2021).
[7] Gupta, Ankush, Andrea Vedaldi, and Andrew Zisserman. ”Synthetic data for text localisation in natural images.” In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 2315-2324. 2016.
[8] Haddad, Josef. ”Data Synthesis in Deep Learning for Object Detection.” (2021).
[9] Blender Online Community, Blender - a 3D modelling and rendering package, Blender Foundation, Stitching Blender Foundation, Amsterdam, 2018, http://www.blender.org, last accessed: 2021-09-08.
[10] A. Y. Barrera-Animas, J. M. Davila Delgado, Generating real-world-like labelled synthetic datasets for construction site applications, Automation in Construction, 151, July, (2023), 104850.
[11] T. Young Kim, Y. Kwon, M. Tunahan Tuna, C. Bgeskov Hansen, Yolo-to-COCO format- converter, 2021, https://github.com/Taeyoung96/Yolo-to-COCO-formatconverter, last accessed: 2021-09-08.
[12] Y. Wu, A. Kirillov, F. Massa, W.-Y. Lo, R. Girshick, Detectron2, 2019, https://github.com/facebookresearch/detectron2, last accessed: 2021-09-20.
[13] Detectron2 contributors, Detectron2’s documentation, 2019-2020, https:// detectron2.readthedocs.io/en/latest/, last accessed: 2021-09-20.
[14] COCO, COCO Detection Evaluation, 2015, https://cocodataset.org/#detection-eval, last accessed: 2021-09-20.