applied sciences

MDPI

Article

Searching for the Ideal Recipe for Preparing Synthetic Data in the Multi-Object Detection Problem

Michał Staniszewski [1,*], Aleksander Kempski [1,2], Michał Marczyk [3,4], Marek Socha [3], Paweł Foszner [1], Mateusz Cebula [1,2], Agnieszka Labus [5], Michał Cogiel [2,6] and Dominik Golba [2,6]

1 Department of Computer Graphics, Vision and Digital Systems, Faculty of Automatic Control, Electronics and Computer Science, Silesian University of Technology, Akademicka 2A, 44-100 Gliwice, Poland; alekkem429@student.polsl.pl (A.K.); pawel.foszner@polsl.pl (P.F.); mateceb250@student.polsl.pl (M.C.)
2 QSystems.pro sp. z o.o. Mochnackiego 34, 41-907 Bytom, Poland; mcogiel@qsystems.pro (M.C.); dgolba@qsystems.pro (D.G.)
3 Department of Data Science and Engineering, Faculty of Automatic Control, Electronics and Computer Science, Silesian University of Technology, Akademicka 16, 44-100 Gliwice, Poland; michal.marczyk@polsl.pl (M.M.); marek.socha@polsl.pl (M.S.)
4 Yale Cancer Center, Yale School of Medicine, New Haven, CT 06510, USA
5 Department of Urban and Spatial Planning, Faculty of Architecture, Silesian University of Technology, Akademicka 7, 44-100 Gliwice, Poland; agnieszka.labus@polsl.pl
6 Blees sp. z o.o. Zygmunta Starego 24a/10, 44-100 Gliwice, Poland
* Correspondence: mstaniszewski@polsl.pl

Abstract: The advancement of deep learning methods across various applications has forced the creation of enormous training datasets. However, obtaining suitable real-world datasets is often challenging for various reasons. Consequently, numerous studies have emerged focusing on the generation and utilization of synthetic data in the training process. Hence, there is no universal formula for preparing synthetic data and leveraging it in network training to maximize the effectiveness of various detection methods. This work provides a comprehensive overview of several synthetic data generation techniques, followed by a thorough investigation into the impact of training methods and the selection of synthetic data quantities. The outcomes of this research enable the formulation of conclusions regarding the recipe for developing synthetic data with high efficacy in enhancing detection methods. The main conclusion for the synthetic data generation methods is to ensure maximum diversity at a high level of photorealism, which allows improving the classification quality by more than 5% to even 19% for different detection metrics.

Keywords: mutli-object detection; synthetic data generation; deep and transfer learning

Academic Editor: Keun Ho Ryu

Received: 7 December 2024
Revised: 28 December 2024
Accepted: 31 December 2024
Published: 2 January 2025

Citation: Staniszewski, M.; Kempski, A.; Marczyk, M.; Socha, M.; Foszner, P.; Cebula, M.; Labus, A.; Cogiel, M.; Golba, D. Searching for the Ideal Recipe for Preparing Synthetic Data in the Multi-Object Detection Problem. *Appl. Sci.* **2025**, *15*, 354. https://doi.org/10.3390/app15010354

1. Introduction

The necessity for annotated data has grown significantly in recent years, as deep learning models heavily depend on large data collection for efficient training. As tasks like people detection and tracking become more complex and precise, the importance of extensive datasets becomes crucial. Object detection methods can be applied widely in various fields, such as manufacturing [1], sports [2], and many others. It is especially essential to have datasets that cover various scenarios, environmental conditions, and diverse demographics to ensure the adaptability and robustness of models. The computer vision community has dedicated resources to create datasets like PASCAL VOC [3] or MS COCO [4], contributing significantly to research on complex tasks across different scenes. However, while beneficial, these datasets do not cover all possible scenarios.

The collection of extensive and diverse real-world data poses various challenges and limitations. Gathering real-world data is a time-consuming process with higher operational costs, particularly when manually annotating thousands of images with numerous objects, inevitably leading to human errors. The current increase in privacy regulations and restrictions on data collection further complicates matters. Real data often contain sensitive information, including people's identities, locations, and activities, raising significant privacy concerns, especially in regions like the US and Europe, where regulations like the EU General Data Protection Regulation [5] have been introduced. Considering the necessity for diverse data to encompass a wide range of scenarios, synthetic data emerges as a valuable solution that can be generated through simulation. The utilization of synthetic data provides the flexibility to create various scenarios with different nuances. With synthetic data, it is possible to have control over the elements present in the scene, visibility of individuals or objects, actions performed by actors, types of locations, and numerous other factors.

Three advanced types of methods generating labeled data can be distinguished, offering richer information about the objects to be detected. The summary of these three categories can be found in Table 1. These methods are performed once before training and can be further augmented. The first type of method is to use background-less images of objects to be learned (named in this work as Img: images generated without 3D software). Such an image is being spawned on another one giving more valuable data. In the work in [6] an algorithm is used in automotive transport for detecting road damage. To ensure the realism of generated data, images of road cracks are only put on specially selected areas. To improve the realism of synthetic data, the work in [7] focuses on using color-grading images and fitting the size of the pasted object. This ensures a better blend with the background, enhancing visual coherence. The work in [8] introduces a cycle-GAN algorithm, creating realistic images based on the provided ones. This is used in multi-organ detection in CT images, generating realistic organ images.

Table 1. Summary of different methods of synthetic generation concerning the Img, Obj, Sim and its application for object detection purposes, where Img: images generated without 3D software, Obj: 3D models generated on a flat background, Sim: images rendered using crowd simulations.

Name	Year	Img	Obj	Sim	Object of Detection
[9]	2023	✓	-	-	Not Specified
[6]	2021	✓	-	-	Road damage and cracks
[8]	2020	✓	-	-	Multi-organ in CT images
[7]	2017	✓	-	-	Groceries
[10]	2023	-	✓	-	Groceries
[11]	2022	-	✓	-	Road Cones
[12]	2019	-	✓	-	Toys and groceries
[13]	2018	-	✓	-	Not specified
[14]	2017	-	✓	-	specific toys in cluttered room
[15]	2015	-	✓	-	Aircraft and drones
[16]	2023	-	-	✓	Vehicles
[17]	2023	-	-	✓	Pedestrians and vehicles
[18]	2023	-	-	✓	Construction yard protection
[19]	2023	-	-	✓	Swimming and drowning people
[20]	2023	-	-	✓	Pedestrians
[21]	2023	-	-	✓	Ants
[22]	2020	-	-	✓	Pedestrians
[23]	2019	-	-	✓	Water closet objects, e.g., toilet, urinal
[24]	2018	-	-	✓	Pedestrians and vehicles
[25]	2018	-	-	✓	Vehicles
[26]	2017	-	-	✓	Pedestrians
[27]	2017	-	-	✓	Groceries

Table 1. *Cont.*

Name	Year	Img	Obj	Sim	Object of Detection
[28]	2017	-	-	✓	Vehicles
[29]	2016	-	-	✓	Various elements of a city
[30]	2016	-	-	✓	Vehicles

The second popular category of methods presented in [11,14] is to use 3D software to render 3D objects on a specific background (Obj: 3D models generated on a flat background). This method works on the assumption that synthetic images do not have to be realistic and ensures that the object is being rendered from multiple angles. Such an approach diversifies the background of an image and provides information about the object from different points of view, which is crucial while detecting more complex shapes. The work in [13] suggests that the best results are obtained when the rendered objects used become background-less, giving the possibility to change the background, as well as to rotate or change their positions. Such an operation should be performed multiple times to ensure filters learn the features of the object, not the background. The authors of [10] use an especially prepared environment with various cameras and visual effects. Objects can be rendered from various camera angles with different light settings. Blur and noise effects are also added to 3D scenes.

The last noted group of methods uses a Game Engine for creating a full crowd simulation (Sim: images rendered using simulation; refer to Figure 1), which is proven to be a valuable tool for validating object detection [17], person detection [22], construction site issues [18], and tracking methods [20,31]. However, it faces limitations in preparing diverse data for training. The challenge lies in the difficulty of achieving a wide range of situations essential for effective training. The complexity lies in addressing the challenge of data diversity in synthetic data generation [13], as certain methods may lead to biased datasets under specific input conditions. Models trained on diverse datasets exhibit enhanced performance when dealing with data beyond the target domain. This is why simulations can only be used to overcome specific problems. One of the possible usages [21] is to render images of walking ants in a randomly generated environment, which is possible because generating forest ground from a top view is a simple task to overcome. Labels are later used to detect, track, and estimate their positions. Another approach [19] uses an ocean simulation to detect people in the water. The simulation uses multiple ship models, varies the weather conditions, and spawns thrash in the sea. The goal of an algorithm is to detect drowning people and to distinguish them from swimming ones. The trained model will only be used over the ocean, so there is no need to diversify the environment so much.

In the context of Virtual Worlds [30], synthetic data have found applications as proxies for multi-object tracking analysis. In the context of detecting cars and traffic, virtual crowd simulation is utilized to augment training data and expand datasets [23], particularly emphasizing highly reflective objects, with a focus on bathroom utilities. The video game environment [29] serves as a common tool, leveraging synthetic samples to achieve results comparable to models trained on real-world data. The integration of synthetic data generated within virtual simulations into training sets proves instrumental in significantly enhancing the performance of object detection algorithms. An approach involving the generation of synthetic objects on real backgrounds [28], featuring a high density of detectable objects, aims to emulate real-world clutter effectively. By incorporating multiple synthetic and real datasets alongside a simulation tool [32], there is potential to create large volumes of affordably annotated synthetic data. This approach can lead to the establishment of domain similarity among these datasets, contributing to more robust and comprehensive training datasets.

Figure 1. Exemplary images of realistic crowd simulation implemented in Unreal Engine. Such an approach gives an option for generating random pedestrian and car movement along with an option for including additional objects. It also enables the simulation of different light reflections and weather conditions.

Several authors have emphasized the importance of achieving photorealism in simulated data, investigating the impact of synthetic datasets generated through photorealistic rendering techniques. This focus extends to areas such as street scene parsing [24] and transfer learning, with a predominant reliance on synthetic training data [27]. In the work in [15], a new algorithm was designed to enhance realism by initiating from a limited set of real images. Then, it estimates the rendering parameters needed to synthesize similar images when provided with a coarse 3D model of the target object. Furthermore, the generation of synthetic data [12] involves incorporating randomized illumination, blur, and noise to address the challenges of object detection in complex environments. This approach aims to overcome the limitations associated with existing methods, which heavily depend on large volumes of labeled real data. In the realm of cross-modality learning, a framework [26] employs terms utilizing a deep convolutional network to establish a non-linear mapping between RGB and thermal data. This enables the learning of features that are both discriminative and resilient to poor illumination conditions.

Models trained exclusively on synthetic data often fall short in performance when tested on real-world datasets, and the process of data synthesis itself likely contributes to the observed domain gap [33]. To mitigate the disparity between real and synthetic data, two common strategies are employed. One approach involves mixing real data into the training set alongside synthetic data. Other strategies include conducting fine-tuning on mixed data after pretraining on larger uncorrelated with the given problem but in a similar domain dataset (like COCO) or enhancing the quality of synthetic data to align more closely with the target domain. While fine-tuning models with real data can lead to improvements, it does not address the fundamental issue of the domain gap. Addressing the domain gap for semantic segmentation [34] can be achieved by adapting the representations learned by segmentation networks across synthetic and real domains. Alternatively, domain randomization [25] can be applied together with fine-tuning on real data. In this approach, simulator parameters such as lighting, pose, and object textures are randomized in non-realistic ways, compelling the neural network to learn the essential features of the object of interest despite the artificial variations introduced during simulation.

The presented investigation shows various methods for preparing synthetic data, along with exploring techniques for training multi-object detection methods for classes

with a limited number of data. The diversity of generated results is a key focus of synthetic data, with considerations given to the issue of photorealism. This study presents a different approach to training a multi-object detection algorithm using real data, synthetic data, and a transfer learning approach. Then, the impact of the relationship between the number of real and synthetic data on the effectiveness of training and classification was analyzed, which allows for determining the required amount of synthetic data that must be added to the training set. Finally, based on existing solutions for generating synthetic data (Img, Obj, and Sim), the influence of data type on classification parameters was checked. The novel outcome of the conducted research is the most effective recipe for generating synthetic data and a method for utilizing such data in training models to detect large-sized objects. This conclusion is derived from the latest knowledge and insights in the field, providing a comprehensive guide for practitioners involved in similar applications.

2. Materials and Methods

2.1. Real Data

The existing databases of labeled photos have a restricted range of object classes. To highlight the issue of insufficient data, as an example, specific classes of objects commonly present in public transport were selected. These classes include items such as bicycles, trolleys, wheelchairs, boxes, suitcases, and bags (Figure 2). Notably, only the bicycle and suitcase classes are directly represented in the COCO database. The bag class was built from classes of backpacks and handbags for which the differences were relatively small.

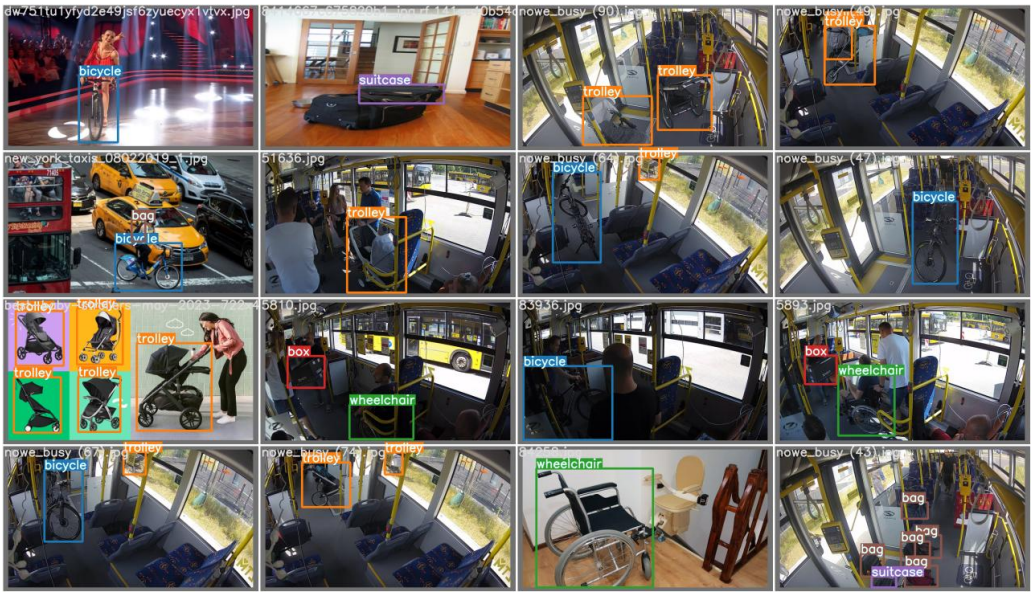

Figure 2. Exemplary set of real data for applied classes of bicycles, trolleys, wheelchairs, boxes, suitcases, and bags composed of data from COCO, searching public databases of unlabeled photos and own public transport dataset.

To create a dataset for network training, a thorough review of photos was conducted to ensure a balanced distribution of the number of photos for each class. The actual data were constructed by using photos from the COCO database and searching publicly available databases of unlabeled photos (which were then manually tagged) and a database of public transport photos [35]. This step aimed to address the challenge of data scarcity and provide a more comprehensive and representative set for training the network. The total amount of data concerning the mentioned classes and division into subsets is given in Table 2.

Table 2. A summary of the amount of data for individual classes used in research (Total), with division into training (Train), validation (Val), and test (Test) sets. Additionally, information about the class's availability in the COCO benchmark is included.

Class Name	Total/Train/Val/Test	COCO
Bicycle	3236/2268/651/317	Yes
Trolley	3249/2262/651/336	No
Wheelchair	3236/2270/654/312	No
Box	3277/2280/627/370	No
Suitcase	3281/2290/674/317	Yes
Bag	3251/2270/671/310	Mix

2.2. Synthetic Data Generation

In the process of generating synthetic data, a primary consideration was given to ensuring diversity in the images. The Unreal Engine 5 was chosen as a guarantee of high data quality. Unreal Engine provides unparalleled graphics quality, advanced physics, and world-building utilities, thereby expediting work processes and elevating the overall quality of generated datasets.

Unreal Engine 5 offers sophisticated tools for managing physics, creating realistic human simulations, and controlling object behavior within a scene. It empowers developers to achieve industry-leading graphics quality. The 3D models of classes listed in Table 2 were created in the Blender environment with various materials (Figure 3). To retrieve detections of specific generated objects, the Unreal GT library [36] was employed. This library enables the direct extraction of detections from crowd simulations (Figure 4d), a tool built for Unreal Engine, during the synthetic data generation process.

Figure 3. View of 3D models used in synthetic data with sample materials.

As mentioned above, there are three common ways of generating synthetic images for object detection. The Img method consists of adding background-less images to the background image. Its main feature is, generating hundreds of images in a short time. The problem lies in varying the pose of an object, which is crucial while learning more complex shapes. Obj method uses the possibilities given by modern 3D graphic engines and is an extension of an Img method, but instead of using flat images of objects to be learned, it uses 3D models rendered on a random background. The main advantage of such an approach is

that an object is being seen from multiple views, which provides more information during the learning process. This is the reason why a decision was made to skip the simple Img approach and use Obj instead. The Sim method is based on a dedicated simulation, and its main power is allowing us to recreate the destination environment, which is the reason why it is being so often used in unusual problems. The main disadvantage of Sim is the difficulty in varying the environment, which is in many cases very time-consuming. To overcome this problem, the 3rd method (Square) was developed, which is a mixture of Obj and Sim methods. The last mentioned method is the mix of images from Obj, Square, and Sim approaches, which provide a rich and diverse dataset. A detailed summary and description of general strategies of data generation are given below:

(**a**) Realistic city crowd simulation (**b**) Static square with changing background

(**c**) Random object and variable background (**d**) Object detection segmentation

Figure 4. Example of generated synthetic data for multiple object detection concerning the different scenarios of generation and object segmentation.

- Realistic city crowd simulation (Sim: Figure 4a)—Features a realistic crowd simulation in an urban setting with moving people, vehicles, weather conditions, and proper lighting effects ensured by various sources of light positions. The scene includes 3D buildings and additional elements like paper, waste bins, letter boxes, and trees. Large objects on sidewalks are registered and marked, and cameras move around objects with randomized positions. Such an approach results in a scene with numerous objects blocking each other, situated at a relatively large distance from the camera.
- Static square with changing background (Square: Figure 4b)—Takes place in a static square with a changing background, where a plane contains randomly placed objects such as benches, lampposts, trees, garbage bins, and static cars, along with pedestrians. In this case, large photos fill the background, randomly positioned around the stage, and sizable objects are placed randomly on the stage. The camera view is random relative to the center of the square, resulting in a multitude of objects obscured by a randomly changing background, and closer proximity to the camera.

- Random object and variable background (Obj: Figure 4c)—The scene comprises nine photos serving as backgrounds (randomly chosen from a large dataset ensuring no connection to our problem and its diversity), with objects generated in empty spaces for photo capture against the specified background. Each photo contains one or a maximum of two objects close to the camera. Generated objects are realistic, but pictures in the background are not related to realistic scenes. Finally, no occlusions are present, but the background changes dynamically.
- Mixed method (Mix)—Combines elements from the three approaches mentioned above to generate a diverse dataset. It utilizes a mixture of realistic crowd simulation, changing backgrounds, and random object placement techniques.

2.3. Multiple Object Detection Strategy

To demonstrate the practical application of synthetic data, the chosen object detection method is You Only Look Once, version 7 (YOLOv7). YOLOv7 [37] is a state-of-the-art real-time object detector that has a good relation of speed and accuracy concerning other object detectors. YOLOv7 method was also chosen due to the clarity of the source code and its possible modifications in the training method. It has surpassed its predecessors and introduced several key features, including model reparameterization, dynamic label assignment, and extended and compound scaling. The computational block in the YOLOv7 backbone is known as the Extended Efficient Layer Aggregation Network (E-ELAN). The utilization of YOLOv7, with its advanced features and optimized architecture, aims to showcase the effectiveness of synthetic data in enhancing the performance of a cutting-edge object detection method for real-world applications.

The model was trained with different proportions of synthetic to real data. The first model was conducted using only real data and was used for comparison with the other approaches. All models trained using synthetic data use the same amount of real data. Generated images were added only to the training dataset with different ratios. Training data were augmented with the use of already implemented YOLOv7 methods such as rotation, merge, shift, rescale, hue change, and noise addition.

During experiments, two training techniques were used: a standard one, training a model from beginning to end, and a transfer-learning method. For the transfer learning approach, a YOLOv7 model trained on a COCO [4] dataset was taken, then the feature extractor (backbone) was frozen and the rest (front bone) was trained regularly. Such an approach not only allows the model to train faster but also improves the accuracy of detections. To test different ways of generating data, models were trained with each synthetic data method separately, and then one was trained with a mix of all approaches. All models were trained for 300 epochs, and hyperparameters were found experimentally and then fine-tuned using a default evolutionary algorithm.

3. Results

All calculations were performed on a computer with the following parameters: Xeon W-3200 processor, number of cores/threads 12/24, processor clock 3.3 GHz, cache memory 19.25 MB, 32GB RAM, and GPU NVIDIA Quadro RTX6000 24GB. The dataset was split in the following manner: training set: 50% of labels, validation set: 30% of labels, test set: 20% of labels. Later, cross-validation of the chosen models was performed. To create hybrid datasets, synthetic data were added with various ratios to the training set. It was ensured that the correct ratio was obtained by calculating the number of labels instead of images.

To evaluate the quality of the models' outcomes, the following evaluation metrics were applied: (a) Precision; (b) Recall; (c) Mean Average Precision (mAP), calculated at an Intersection over Union (IoU) threshold of 0.55 (mAP@0.5); (d) mAP, evaluated using a

series of IoU thresholds ranging from 0.5 to 0.954 (mAP@0.5:0.95); (e) F1 score; (f) Box loss metric, indicating how accurately the algorithm can detect an object's center; (g) Objectness, measured as the probability of an object existing in a proposed region of interest; and (h) Classification, utilizing the assignment of a class label to the detected object.

3.1. Evaluation of Different Training Approaches

As intended, the multi-object detection method was trained first on real data. Subsequently, a transfer learning procedure was executed using solely real data. In the second phase, a combination of synthetic data and real data, called hybrid data, was employed. The transfer learning approach was then repeated with this hybrid data, ensuring an equal distribution of synthetic and real data in the hybrid set. The results demonstrate that incorporating synthetic data in the form of hybrid data significantly enhances detection quality (refer to Table 3).

Table 3. Results of different approaches in the training of object detection models with the application of real and hybrid data concerning Precision, Recall, mAP parameters, F1 score, Box loss, Objectness, and Classification. The best results were marked in bold.

Method	Prec. ↑	Rec. ↑	mAP0.5 ↑	mAP:0.95 ↑	F1 sc. ↑	Box ↓	Object. ↓	Class. ↓
real	0.9162	0.8908	0.9360	0.7052	0.9033	0.0422	0.0081	0.0076
tf. real	0.9562	0.9472	0.9718	0.8018	0.9517	0.0350	0.0091	0.0039
hybrid	0.9646	0.9656	0.9848	0.8350	0.9651	0.0320	**0.0074**	0.0027
tf. hybr.	**0.9850**	**0.9862**	**0.9952**	**0.8940**	**0.9856**	**0.0228**	0.0077	**0.0018**

Training detection methods exclusively with real data proved to be the least effective, prompting the utilization of the transfer learning approach in this scenario. Further improvements were observed when transfer learning (tf. real), hybrid data (hybrid), and transfer learning for hybrid data (tf. hybr.) were applied, respectively. The results for the Precision, Recall, F1 score, and mAP 0.5 parameters are above the value of 0.9, and only a large difference is visible for the precise detection of the mAP 0.5:0.95 parameter, where the transfer learning approach for hybrid data performs relatively best.

On an individual class basis, YOLOv7 demonstrated commendable results for approximate detection (mAP 0.5). However, a marked enhancement in the classification efficiency of individual classes was achieved through the introduction of the transfer learning approach for hybrid data, evident in the mAP parameter 0.5:0.95 in Table 4.

Table 4. Results of mAP parameters (0.5 and 0.5:0.95) for selected classes in multi-object detection, proving that transfer learning applied for hybrid data is more accurate than real data with a limited number of images. The best results were marked in bold.

Class	Bicycle		Trolley		Wheelchair		Box		Suitcase		Bag	
mAP ↑	0.5	:0.95	0.5	:0.95	0.5	:0.95	0.5	:0.95	0.5	:0.95	0.5	:0.95
real	0.98	0.79	0.98	0.81	0.96	0.67	0.89	0.64	0.94	0.71	0.86	0.61
tf. real	0.99	0.85	0.99	0.87	0.98	0.77	0.96	0.77	0.97	0.80	0.94	0.74
hybrid	0.99	0.87	0.99	0.88	0.99	0.81	0.98	0.81	0.99	0.85	0.97	0.77
tf. hybr.	**1.00**	**0.91**	**1.00**	**0.92**	**1.00**	**0.87**	**0.99**	**0.89**	**1.0**	**0.90**	**0.99**	**0.86**

3.2. Estimation of Synthetic Data Amount

This research also investigated the optimal proportion of generated synthetic data in the training set. For hybrid data and the transfer learning approach, various ratios of synthetic data to real data were examined. Firstly, only synthetic data were used for training, indicating the worst results. On the other hand, when only real data were used,

the highest mAP was observed (refer to Figure 5, top plot, and Table 5). Still, when an equal amount of synthetic and real data are used, the model performance is slightly worse than for a full real dataset. Even in extreme cases, where 90% synthetic data and 10% real data are used, the results are worse only by an average of 7–11% compared with the full real data. In the second step, synthetic data were added to the full real dataset (refer to Figure 5, bottom plot, and Table 5). There is a trend showing that adding synthetic data significantly improves the results. The major boost is observed after adding 25% of synthetic data. Adding synthetic data, and thus enlarging the training set, definitely improves the results. This confirms the usefulness of synthetic data in the context of supplementing real data.

Table 5. Results of different ratios of real and synthetic data in the training of models concerning Precision, Recall, mAP parameters, F1 score, Box loss, Objectness, and Classification for the raw and transfer learning approach. The best results were marked in bold.

Real	Synt.	Prec. ↑	Rec. ↑	mAP 0.5 ↑	:0.95 ↑	F1 sc. ↑	Box ↓	Object. ↓	Class. ↓
0	1	0.156	0.320	0.210	0.0521	0.032	0.076	0.0167	0.0359
0.05	0.95	0.652	0.584	0.616	0.598	0.337	0.041	0.0079	0.0133
0.1	0.9	0.842	0.766	0.802	0.814	0.539	0.052	0.0109	0.0111
0.2	0.8	0.841	0.773	0.806	0.825	0.549	0.050	0.0109	0.0107
0.25	0.75	0.869	0.763	0.813	0.823	0.551	0.041	0.0096	0.0098
0.5	0.5	0.850	0.822	0.836	0.863	0.590	0.044	0.0086	0.0090
0.75	0.25	0.882	0.828	0.854	0.88	0.621	0.041	0.0086	0.0075
1	0	0.916	0.891	0.903	0.936	0.705	0.042	0.0081	0.0076
1	0.25	0.959	0.971	0.965	0.985	0.823	0.032	0.0082	0.0050
1	0.5	0.962	0.975	0.968	0.989	0.840	0.032	0.0083	0.0046
1	0.75	0.965	0.972	0.968	0.987	0.843	0.032	0.0082	0.0050
1	1	0.965	0.966	0.965	0.985	0.835	**0.027**	**0.0074**	**0.0027**
1	1.25	0.969	0.970	0.969	**0.990**	0.849	0.033	0.0082	0.0047
1	1.5	**0.974**	0.972	0.973	**0.990**	**0.853**	0.032	0.0081	0.0041
1	1.75	**0.974**	**0.974**	**0.974**	**0.990**	0.852	0.033	0.0083	0.0044
1	2	0.964	**0.979**	0.971	0.989	0.851	0.033	0.0084	0.0050
Transfer learning approach									
0	1	0.354	0.289	0.318	0.261	0.143	0.0635	0.0185	0.0322
0.05	0.95	0.845	0.749	0.794	0.802	0.519	0.0337	0.0085	0.0059
0.1	0.9	0.885	0.792	0.836	0.843	0.568	0.0323	0.0091	0.0063
0.2	0.8	0.892	0.838	0.864	0.878	0.624	0.0381	0.0138	0.0070
0.25	0.75	0.908	0.837	0.871	0.882	0.634	0.0320	0.0115	0.0051
0.5	0.5	0.892	0.844	0.867	0.881	0.635	0.0330	0.0114	0.0052
0.75	0.25	0.907	0.895	0.901	0.924	0.695	0.0306	0.0097	0.0043
1	0	0.956	0.947	0.952	0.972	0.802	0.0350	0.0091	0.0039
1	0.25	**0.988**	0.982	0.985	0.995	0.895	0.0270	0.0095	0.0027
1	0.5	0.986	0.983	0.984	0.995	0.893	0.0280	0.0093	0.0032
1	0.75	0.986	**0.990**	**0.988**	**0.996**	**0.902**	0.0280	0.0092	0.0035
1	1	0.985	0.986	0.986	0.995	0.894	**0.0228**	**0.0077**	**0.0018**
1	1.25	**0.988**	0.987	0.987	0.995	0.897	0.0290	0.0091	0.0031
1	1.5	0.986	0.989	0.987	0.995	0.897	0.0300	0.0089	0.0034
1	1.75	0.986	0.985	0.985	0.995	0.893	0.030	0.0089	0.0037
1	2	0.982	0.988	0.985	0.994	0.894	0.030	0.0089	0.0038

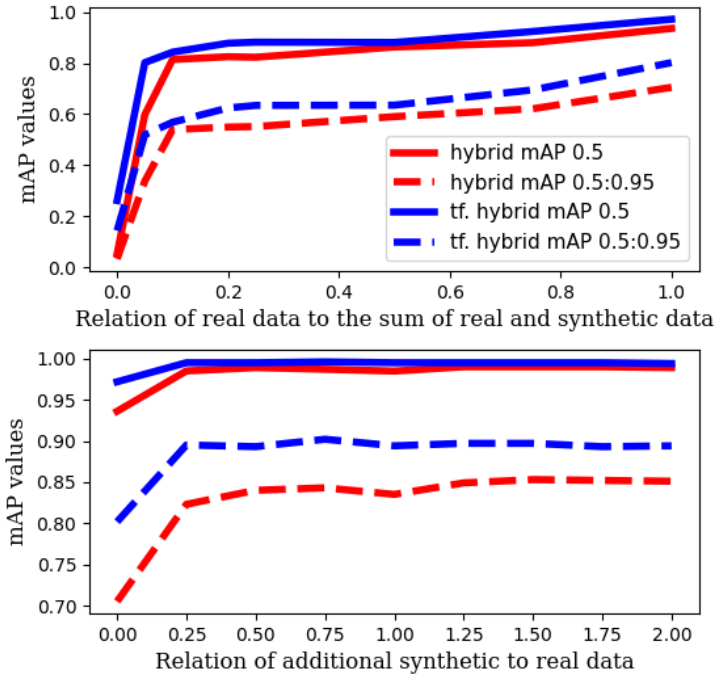

Figure 5. Results of different approaches in the training process with the application of various relations of real and synthetic data concerning the mAP parameters. The upper plot shows the ratio of real data to the sum of real and synthetic data. In contrast, the bottom shows the relation of additional synthetic data applied to full real data.

3.3. Choice of Synthetic Method Generation

During this phase, diverse methods for generating synthetic data were rigorously tested and detailed in the corresponding Section 2.2. The YOLOv7 detection method was individually trained on each approach through the transfer learning method. Subsequently, an equivalent number of images were randomly selected from the datasets associated with each approach, ensuring parity in the size of each set. After a comprehensive evaluation of the various strengths and weaknesses inherent in each solution, it was determined that the most favorable results were achieved by combining all approaches of synthetic generation methods (refer to Table 6).

Table 6. Results of different approaches in the generation of synthetic data compared with all parameters. The mixed scenario of all kinds of data generation gives the most promising results confirming the diversity of data.

Method	Prec. ↑	Rec. ↑	mAP0.5 ↑	mAP:0.95 ↑	F1 sc. ↑	Box ↓	Object. ↓	Class. ↓
Sim	0.927	0.907	0.936	0.721	0.9169	0.029	0.0087	0.0032
Obj	0.925	0.890	0.935	0.716	0.9072	0.0277	0.0100	0.0026
Square	0.918	0.878	0.925	0.696	0.8976	0.032	0.0083	0.0042
Mix	**0.985**	**0.986**	**0.995**	**0.894**	**0.9856**	**0.0228**	**0.0077**	**0.0018**

4. Discussion

The presented work explores diverse methods of generating synthetic data, including crowd simulation, utilizing a plane with a variable background, and incorporating objects onto photo backgrounds. Each approach has its own set of advantages and drawbacks, with typical crowd simulation offering a limited amount of repeatable data, potentially leading to more random outcomes.

The research involved the inclusion of object classes represented in the COCO database, as well as those requiring manual search and marking. Initially, the YOLOv7 network was trained from scratch using real data. Subsequently, a pretrained model was employed to enhance results. Finally, synthetic data, a blend of real and simulation data (hybrid data) in varying proportions, were introduced, and the network training method was repeated similarly to real data. The research also investigated the impact of options for generating simulation data and the quantity of synthetic data relative to real data on classification quality.

The results of the research show that the use of real data in the training process is insufficient. Particularly for classes with limited or no available source data, the use of synthetic data emerges as a highly effective alternative. This study employed the well-established method of training the YOLOv7 network, with an extensive analysis of various data generation approaches. Table 3 illustrates that employing hybrid data for the transfer learning approach yields the best results across all indicated parameters. Notably, this approach significantly enhances the precision of object detection, as evident in the mAP parameter 0.5:0.95.

Furthermore, this research reveals that the influence of the amount of synthetic data relative to real data is minimal (refer to Figure 5). However, it is worth noting that synthetic data can be a good replacement and complement to real data when balancing data from multiple classes is required. The research results present the same amount of real and synthetic data in a hybrid set. However, in scenarios with limited real data, supplementing the set with a larger volume of synthetic data proves beneficial. However, in scenarios with limited real data, supplementing the set with a larger volume of synthetic data proves beneficial. That is crucial for ensuring a proper balance of data in case not all classes are represented by a sufficient number of real data. Synthetic data, with their capacity to generate diverse yet realistic images, stand out as a key feature.

The primary criterion for data generation was maximizing diversity, and the use of the Unreal Engine game engine helped maintain photorealism. Assessing the impact of photo realism on subsequent work proved challenging due to the absence of clear parameters for evaluating such realism. Diversity in the data was achieved through the incorporation of different 3D models, multiple models in a single scene, object occlusions, random arrangement of models, random positioning of camera views, diverse backgrounds, and lighting. This diversity, practically unreachable in reality, fills a crucial gap when actual images are unavailable.

Promising methods for data generation are generative models, such as generative adversarial networks [38] and diffusion models [39], which are currently used in many applications. This type of methodology currently serves mainly as a data augmentation method, which creates coherent and logical images based on already provided pictures. Future development of these methods may make them an alternative to generating synthetic data and will speed up the process of preparing training data.

The strategy of training on synthetic data significantly affects the results and influences the layers of neural networks [16]. Notably, the comparison between a detector trained on real data and one trained on synthetic data revealed the highest similarity in the early layers, while the most significant difference was observed in the head part of the network. Feature extractors (first layers) are only responsible for detecting various shapes that are identical for similar problems. This observation was used for a simple yet effective approach [14], where the layers responsible for feature extraction were frozen in pretrained models on real images. Subsequently, only the remaining layers were trained on synthetic data.

The effective application of simulation data, treated as synthetic data, plays a pivotal role in enhancing the quality of training data preparation. On one hand, it facilitates the

generation of substantial datasets that enrich real data with diverse sets in various aspects. On the other hand, synthetic data become crucial when real data for a specific class is lacking in any existing dataset. However, a significant challenge lies in appropriately structuring the classifier architecture to bridge the gap between the domains of real and synthetic data. The presented work delves into extensive research on the application of synthetic data in the detection of numerous objects. The selected list of objects includes classes with available data in publicly accessible benchmarks and classes requiring manual resource search.

The utilization of synthetic data as a form of data augmentation is increasingly prevalent today. Image augmentation, as discussed in [9], is a popular way to boost dataset quality. It involves rotating, scaling, adjusting color, and blending images to create more labeled data, preventing overfitting during training. Such a process can be carried out in each epoch, providing rich data every iteration. With numerous data generation scenarios available, synthetic data finds practical applications in detecting various objects. Current knowledge enables the generation of realistic data, and as successive graphics engines are released, the photorealism of simulations is expected to improve. Nevertheless, a significant challenge for the future lies in achieving appropriate domain transfer, domain bias, and solutions for overfitting to construct a classifier based solely on synthetic images that remain effective on real data.

5. Conclusions

Synthetic data generation serves as an ideal solution for training deep learning methods, especially in scenarios where accessing real data is impractical or restricted. Synthetic data can also be a good complement to real data. In the course of the research, diverse methods for generating simulation data were explored, along with assessing the training approach for the existing detection method while considering the balance between generated and real data. The main limitation of synthetic data generation methods is the need to have even a small amount of real data, as well as the evaluation of the domain gap, and this is the main challenge for future research work. The distinctive characteristics of synthetic data lie in their diversity, realism, and blended generation approach, making them a perfect recipe for effective utilization in various applications.

Author Contributions: The presented work was finished by cooperation between IT specialists and software developers. Conceptualization, M.S., A.K. and M.M.; methodology, M.S. and A.K.; resources, P.F. and M.S.; software, M.C. (Michał Cogiel), D.G., M.S. and M.C. (Mateusz Cebula); supervision, M.S. and M.M.; data curation, A.K. and M.C. (Mateusz Cebula); validation, M.S. and M.M.; visualization, P.F. and A.L.; investigation, M.S. and A.K.; writing—original draft, M.S., A.K. and M.C. (Mateusz Cebula); writing—review and editing, M.S., A.K. and M.M.; funding acquisition, M.S., M.C. (Michał Cogiel) and D.G. All authors have read and agreed to the published version of the manuscript.

Funding: This work was partly supported by European Union Funds Awarded to Blees Sp. z o. o. "Development of a system for analyzing vision data captured by public transport vehicles interior monitoring, aimed at detecting undesirable situations/behaviors and passenger counting (including their classification by age group) and the objects they carry" under Grant POIR.01.01.01-00-0952/20-00; and in part by the Silesian University of Technology (SUT) grant for Maintaining and Developing Research Potential under Grant 02/070/BK-24/0052 (MM) and 02/090/BK-24/0043 (MS). This research was supported by the European Union from the European Social Fund in the framework of the project "SUT as a Center of Modern Education based on research and innovation" POWR.03.05.00-00-Z098/17.

Informed Consent Statement: Informed consent was obtained from all subjects involved in the study.

Data Availability Statement: The given work presents a possible method of synthetic data generation and includes data generated in crowd simulation under https://zenodo.org/records/14500676. Access date 30 December 2024.

Conflicts of Interest: Authors Aleksander Kempski and Mateusz Cebula were employed by the company QSystems.pro. The authors declare that they work in QSystems.pro. Authors Michał Cogiel and Dominik Golba were employed by the company QSystems.pro and Blees. The authors declare that they work in QSystems.pro and Blees. The funders participated in the design of the study, as well as in the analyses and interpretation of data, the writing of this article or the decision to submit it for publication. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

Img	Images generated without 3D software
Obj	3D models generated on a flat background
Sim	Images rendered using crowd simulations
Square	Static square with changing background
Mix	Combines elements from different approaches to generate a diverse dataset

References

1. Ahmad, H.M.; Rahimi, A. Deep learning methods for object detection in smart manufacturing: A survey. *J. Manuf. Syst.* **2022**, *64*, 181–196. [CrossRef]
2. Goh, G.L.; Goh, G.D.; Pan, J.W.; Teng, P.S.P.; Kong, P.W. Automated Service Height Fault Detection Using Computer Vision and Machine Learning for Badminton Matches. *Sensors* **2023**, *23*, 9759. [CrossRef] [PubMed]
3. Everingham, M.; Gool, L.V.; Williams, C.K.I.; Winn, J.M.; Zisserman, A. The Pascal Visual Object Classes (VOC) Challenge. *Int. J. Comput. Vis.* **2010**, *88*, 303–338. [CrossRef]
4. Lin, T.; Maire, M.; Belongie, S.J.; Bourdev, L.D.; Girshick, R.B.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; Zitnick, C.L. Microsoft COCO: Common Objects in Context. *arXiv* **2014**, arXiv:1405.0312 .
5. Voigt, P.; Bussche, A.v.d. *The EU General Data Protection Regulation (GDPR): A Practical Guide*, 1st ed.; Springer: Berlin/Heidelberg, Germany, 2017. [CrossRef]
6. Zhou, S.; Bi, Y.; Wei, X.; Liu, J.; Ye, Z.; Li, F.; Du, Y. Automated detection and classification of spilled loads on freeways based on improved YOLO network. *Mach. Vis. Appl.* **2021**, *32*, 44. [CrossRef]
7. Georgakis, G.; Mousavian, A.; Berg, A.C.; Kosecka, J. Synthesizing Training Data for Object Detection in Indoor Scenes. *arXiv* **2017**, arXiv:1702.07836. [CrossRef]
8. Hammami, M.; Friboulet, D.; Kechichian, R. Cycle GAN-Based Data Augmentation For Multi-Organ Detection In CT Images Via Yolo. In Proceedings of the 2020 IEEE International Conference on Image Processing (ICIP), Virtual Conference, 25–28 October 2020; pp. 390–393. [CrossRef]
9. Xu, M.; Yoon, S.; Fuentes, A.; Park, D.S. A Comprehensive Survey of Image Augmentation Techniques for Deep Learning. *Pattern Recognit.* **2023**, *137*, 109347. [CrossRef]
10. Westerski, A.; Teck, F.W. Synthetic Data for Object Detection with Neural Networks: State of the Art Survey of Domain Randomisation Techniques. *ACM Trans. Multimedia Comput. Commun. Appl.* 2023, *just accepted*. [CrossRef]
11. Adam, R.; Janciauskas, P.; Ebel, T.; Adam, J. Synthetic Training Data Generation and Domain Randomization for Object Detection in the Formula Student Driverless Framework. In Proceedings of the 2022 International Conference on Electrical, Computer, Communications and Mechatronics Engineering (ICECCME), Male, Maldives, 16–18 November 2022; pp. 1–6. [CrossRef]
12. Hinterstoisser, S.; Pauly, O.; Heibel, H.; Marek, M.; Bokeloh, M. An Annotation Saved is an Annotation Earned: Using Fully Synthetic Training for Object Instance Detection. *arXiv* **2019**, arXiv:1902.09967. [CrossRef]
13. Mayer, N.; Ilg, E.; Fischer, P.; Hazırbaş, C.; Cremers, D.; Dosovitskiy, A.; Brox, T. What Makes Good Synthetic Training Data for Learning Disparity and Optical Flow Estimation? *Int. J. Comput. Vis.* **2018**, *126*, 942–960. [CrossRef]
14. Hinterstoisser, S.; Lepetit, V.; Wohlhart, P.; Konolige, K. On Pre-Trained Image Features and Synthetic Images for Deep Learning. *arXiv* **2017**, arXiv:1710.10710. [CrossRef]
15. Rozantsev, A.; Lepetit, V.; Fua, P. On rendering synthetic images for training an object detector. *Comput. Vis. Image Underst.* **2015**, *137*, 24–37. [CrossRef]

16. Ljungqvist, M.G.; Nordander, O.; Skans, M.; Mildner, A.; Liu, T.; Nugues, P. Object Detector Differences when using Synthetic and Real Training Data. *SN Comput. Sci.* **2023**, *4*, 302. [CrossRef]
17. Foszner, P.; Szczęsna, A.; Ciampi, L.; Messina, N.; Cygan, A.; Bizoń, B.; Cogiel, M.; Golba, D.; Macioszek, E.; Staniszewski, M. CrowdSim2: An Open Synthetic Benchmark for Object Detectors. In Proceedings of the 18th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP 2023), Lisbon, Portugal, 19–21 February 2023; Volume 5: VISAPP, INSTICC; SciTePress: Setubal, Portugal, 2023; pp. 676–683. [CrossRef]
18. Quattrocchi, C.; Mauro, D.D.; Furnari, A.; Lopes, A.; Moltisanti, M.; Farinella, G. Put Your PPE on: A Tool for Synthetic Data Generation and Related Benchmark in Construction Site Scenarios. In Proceedings of the 18th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP 2023), Lisbon, Portugal, 19–21 February 2023; Volume 4: VISAPP, INSTICC; SciTePress: Setubal, Portugal, 2023; pp. 656–663. [CrossRef]
19. Poudel, R.; Lima, L.; Andrade, F. A Novel Framework To Evaluate and Train Object Detection Models for Real-Time Victims Search and Rescue at Sea with Autonomous Unmanned Aerial Systems Using High-Fidelity Dynamic Marine Simulation Environment. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV) Workshops, Waikoloa, HI, USA, 2–7 January 2023; pp. 239–247. [CrossRef]
20. Foszner, P.; Szczęsna, A.; Ciampi, L.; Messina, N.; Cygan, A.; Bizoń, B.; Cogiel, M.; Golba, D.; Macioszek, E.; Staniszewski, M. Development of a Realistic Crowd Simulation Environment for Fine-Grained Validation of People Tracking Methods. In Proceedings of the 18th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP 2023), Lisbon, Portugal, 19–21 February 2023; Volume 1: GRAPP, INSTICC; SciTePress: Setubal, Portugal, 2023; pp. 222–229. [CrossRef]
21. Plum, F.; Bulla, R.; Beck, H.K.; Imirzian, N.; Labonte, D. replicAnt: A pipeline for generating annotated images of animals in complex environments using Unreal Engine. *Nat. Commun.* **2023**, *14*, 7195. [CrossRef]
22. Ciampi, L.; Messina, N.; Falchi, F.; Gennaro, C.; Amato, G. Virtual to Real Adaptation of Pedestrian Detectors. *Sensors* **2020**, *20*, 5250. [CrossRef]
23. Hartwig, S.; Ropinski, T. Training Object Detectors on Synthetic Images Containing Reflecting Materials. *arXiv* **2019**, arXiv:1904.00824. [CrossRef]
24. Wrenninge, M.; Unger, J. Synscapes: A Photorealistic Synthetic Dataset for Street Scene Parsing. *arXiv* **2018**, arXiv:1810.08705. [CrossRef]
25. Tremblay, J.; Prakash, A.; Acuna, D.; Brophy, M.; Jampani, V.; Anil, C.; To, T.; Cameracci, E.; Boochoon, S.; Birchfield, S. Training Deep Networks with Synthetic Data: Bridging the Reality Gap by Domain Randomization. *arXiv* **2018**, arXiv:1804.06516. [CrossRef]
26. Xu, D.; Ouyang, W.; Ricci, E.; Wang, X.; Sebe, N. *Learning Cross-Modal Deep Representations for Robust Pedestrian Detection*; IEEE Computer Society: Washington, DC, USA, 2017. [CrossRef]
27. Rajpura, P.S.; Bojinov, H.; Hegde, R.S. Object Detection Using Deep CNNs Trained on Synthetic Images. *arXiv* **2017**, arXiv:1706.06782. [CrossRef]
28. Alhaija, H.A.; Mustikovela, S.K.; Mescheder, L.; Geiger, A.; Rother, C. Augmented Reality Meets Computer Vision: Efficient Data Generation for Urban Driving Scenes. *arXiv* **2017**, arXiv:1708.01566. [CrossRef]
29. Shafaei, A.; Little, J.J.; Schmidt, M. Play and Learn: Using Video Games to Train Computer Vision Models. *arXiv* **2016**, arXiv:1608.01745. [CrossRef]
30. Gaidon, A.; Wang, Q.; Cabon, Y.; Vig, E. Virtual Worlds as Proxy for Multi-Object Tracking Analysis. *arXiv* **2016**, arXiv:1605.06457.
31. Staniszewski, M.; Foszner, P.; Kostorz, K.; Michalczuk, A.; Wereszczyński, K.; Cogiel, M.; Golba, D.; Wojciechowski, K.; Polański, A. Application of Crowd Simulations in the Evaluation of Tracking Algorithms. *Sensors* **2020**, *20*, 4960. [CrossRef]
32. Nowruzi, F.E.; Kapoor, P.; Kolhatkar, D.; Hassanat, F.A.; Laganiere, R.; Rebut, J. How much real data do we actually need: Analyzing object detection performance using synthetic and real data. *arXiv* **2019**, arXiv:1907.07061. [CrossRef]
33. Tsirikoglou, A.; Eilertsen, G.; Unger, J. A Survey of Image Synthesis Methods for Visual Machine Learning. *Comput. Graph. Forum* **2020**, *39*, 426–451. [CrossRef]
34. Sankar, S.; Balaji, Y.; Jain, A.; Lim, S.N.; Chellappa, R. *Learning from Synthetic Data: Addressing Domain Shift for Semantic Segmentation*; IEEE Computer Society: Washington, DC, USA, 2018; p. 3761. [CrossRef]
35. Ciampi, L.; Foszner, P.; Messina, N.; Staniszewski, M.; Gennaro, C.; Falchi, F.; Serao, G.; Cogiel, M.; Golba, D.; Szczęsna, A.; et al. Bus Violence: An Open Benchmark for Video Violence Detection on Public Transport. *Sensors* **2022**, *22*, 8345. [CrossRef] [PubMed]
36. Pollok, T.; Junglas, L.; Ruf, B.; Schumann, A. UnrealGT: Using Unreal Engine to Generate Ground Truth Datasets. In Proceedings of the Advances in Visual Computing: 14th International Symposium on Visual Computing, ISVC 2019, Lake Tahoe, NV, USA, 7–9 October 2019; Proceedings, Part I; Springer: Berlin/Heidelberg, Germany, 2019; pp. 670–682. [CrossRef]
37. Wang, C.Y.; Bochkovskiy, A.; Liao, H.Y.M. YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. *arXiv* **2022**, arXiv:2207.02696. [CrossRef]

38. Akkem, Y.; Biswas, S.K.; Varanasi, A. A comprehensive review of synthetic data generation in smart farming by using variational autoencoder and generative adversarial network. *Eng. Appl. Artif. Intell.* **2024**, *131*, 107881. [CrossRef]
39. Chen, D.; Qi, X.; Zheng, Y.; Lu, Y.; Huang, Y.; Li, Z. Synthetic data augmentation by diffusion probabilistic models to enhance weed recognition. *Comput. Electron. Agric.* **2024**, *216*, 108517. [CrossRef]