


Article

Synthetic Training Data for Semantic Segmentation of the Environment from UAV Perspective

Christoph Hinniger^{1,2,†} and Joachim Rüter^{1,*,†} ¹ German Aerospace Center (DLR), Institute of Flight Systems, 38108 Braunschweig, Germany² Faculty of Mechanical Engineering, Technical University Braunschweig, 38106 Braunschweig, Germany

* Correspondence: joachim.rueter@dlr.de

† These authors contributed equally to this work.

Abstract: Autonomous unmanned aircraft need a good semantic understanding of their surroundings to plan safe routes or to find safe landing sites, for example, by means of a semantic segmentation of an image stream. Currently, Neural Networks often give state-of-the-art results on semantic segmentation tasks but need a huge amount of diverse training data to achieve these results. In aviation, this amount of data is hard to acquire but the usage of synthetic data from game engines could solve this problem. However, related work, e.g., in the automotive sector, shows a performance drop when applying these models to real images. In this work, the usage of synthetic training data for semantic segmentation of the environment from a UAV perspective is investigated. A real image dataset from a UAV perspective is stylistically replicated in a game engine and images are extracted to train a Neural Network. The evaluation is carried out on real images and shows that training on synthetic images alone is not sufficient but that when fine-tuning the model, they can reduce the amount of real data needed for training significantly. This research shows that synthetic images may be a promising direction to bring Neural Networks for environment perception into aerospace applications.

Keywords: environment perception; semantic segmentation; synthetic data; game engine; sim-to-real gap; machine learning; unmanned aerial vehicle

**Citation:** Hinniger, C.; Rüter, J.Synthetic Training Data for Semantic Segmentation of the Environment from UAV Perspective. *Aerospace* **2023**, *10*, 604. <https://doi.org/10.3390/aerospace10070604>

Academic Editors: Chao-Yang Lee and Gokhan Inalhan

Received: 27 February 2023

Revised: 21 June 2023

Accepted: 23 June 2023

Published: 30 June 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The ability to recognize obstacles, to find possible emergency landing sites and to plan collision-free flight routes is crucial for the safe autonomy of unmanned aerial vehicles (UAVs). For this reason, it must be ensured that the UAV correctly perceives and interprets its environment. Currently, this is primarily achieved with camera data [1]. In a lot of applications, Neural Networks provide state-of-the-art results for analyzing and interpreting the raw RGB images. They can provide a detailed picture of the surroundings by semantic segmentation of the photos, which means assigning a class to each pixel of the image [2]. However, in order to achieve good results, the Neural Network has to be trained with a large number of diverse images, including the corresponding label masks [3].

The manual generation of such a dataset involves a significant amount of work and is especially difficult in aviation. On the one hand, it is not always possible to take pictures with a UAV. This can be due to regulations such as no-fly zones or privacy reasons, but also due to safety concerns in case of a malfunction of the UAV. On the other hand, the manual creation of labels for semantic segmentation is time-consuming and costly since each pixel of the image has to be assigned to a class. For example, creating a segmentation mask for a single video frame can take up to 90 min [4]. The effort increases for larger and more complex images with a high level of detail.

Instead of recording and labeling real-world images, one possible solution to avoid this problem is to create and extract synthetic images from game engines which are used to create modern video games. Examples of modern game engines include Unreal Engine [5]

and Unity [6]. They offer the possibility to create diverse environments and scenarios. This is also true for situations for which there would be safety and regulatory concerns in the real world. Furthermore, images and corresponding labels can be automatically extracted using existing software tools which mitigates the time-consuming labeling process.

Synthetic image datasets for the training of machine learning models have already been used in the automotive industry, although with moderate success (see, e.g., [7]). A model trained with synthetic images usually performs worse once it is applied to real-world images. This is referred to by the term *domain shift*. In the context of synthetic to real data, this is also known as a *sim-to-real gap*. The reasons for the performance drop are not yet clearly understood. The first assumptions found in literature are that, for example, the textures between the synthetic training dataset and the real evaluation dataset are too dissimilar [8]. However, in the context of UAVs, such texture differences may have less influence on the domain shift due to the higher altitude and the limited resolution of image sensors.

In this context, this paper investigates the question of whether synthetic images can be a suitable alternative to real-world images for training Neural Networks for semantic segmentation of the environment of a UAV. To the best of our knowledge, this has not yet been researched, although it can offer an enormous benefit for the deployment and validation of UAVs due to the elimination of the time and effort required for labeling the images as well as the elimination of the above-mentioned safety and regulatory concerns. To address this question, this work follows the concept presented in Figure 1. The grey boxes represent the datasets used to investigate the stated question. For evaluation, a real-world dataset (*Ruralscapes* [9]) is used. For training the machine learning model, the real dataset is stylistically recreated within a simulation environment (green box) and synthetic images are extracted and used for training. The synthetically created dataset will also be published. To evaluate if the synthetic data can be an alternative when only real-world data from a different geographical region are available, a third dataset from a different geographical region (UAVid [10]) is used. The corresponding trainings are symbolized in the yellow boxes. Besides regular training, the influence of fine-tuning (blue box) with the real training images on the performance of the model trained on synthetic images is investigated. By comparing the results, a conclusion can be drawn regarding the possible existence of a domain shift or sim-to-real gap. Furthermore, it can be concluded to what extent synthetic images can be used to train a machine learning model for semantic segmentation from a UAV perspective. This step is symbolized with the orange box.

Our main contributions are as follows: (i) give a short overview of existing real-world datasets for semantic segmentation from a UAV perspective using a front-facing camera; (ii) stylistically recreate an existing real-world dataset for semantic segmentation from a UAV perspective using a game engine, describe the process and possible problems, publish the dataset for further research; (iii) investigate to what extent synthetic images can be used to improve the environment perception of UAVs or to minimize the need of real data; and (iv) investigate whether synthetic images are a suitable alternative for the training of a machine learning model for environment perception of UAVs in case only real images from a different region are available.

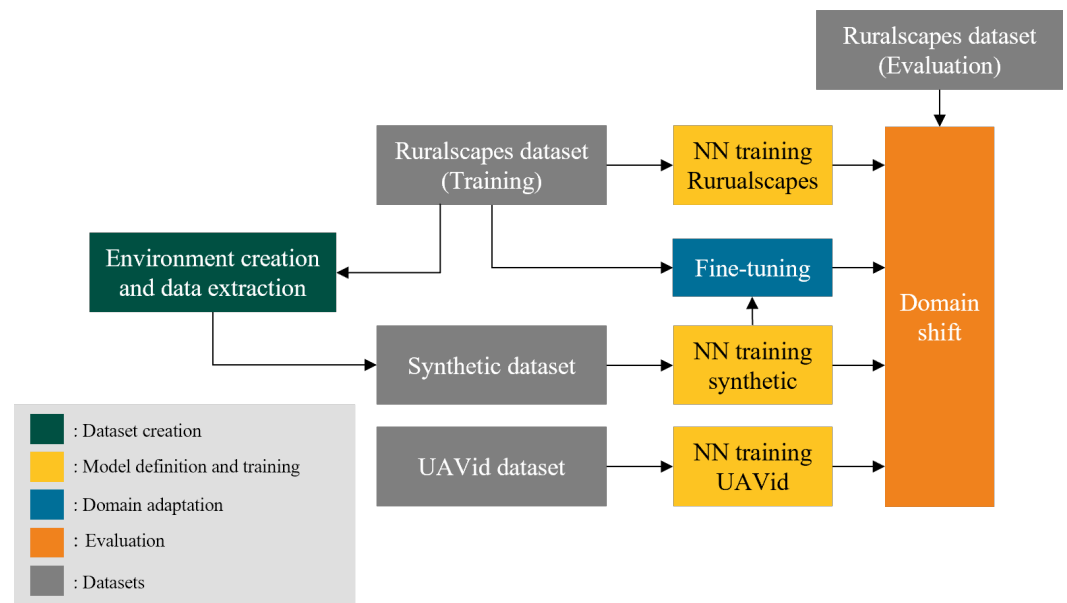


Figure 1. Concept of this work to examine the usefulness of synthetic images for the training of an environment perception ML model. After the creation of the synthetic dataset (green), Neural Networks will be trained (yellow) using different datasets (grey). The influence of fine-tuning is also investigated (blue). An evaluation (orange) of all the trained models is performed using a real-world dataset.

2. Related Work

2.1. Available Datasets for Semantic Segmentation from UAV Perspective

Although there exist some real-world semantic segmentation datasets for car scenes, there are only a few from UAV perspective. In contrast to cars, UAVs are able to move in six degrees of freedom, providing a significantly larger variation of perspectives and scenes [11]. Most of the existing datasets for UAVs are recorded from the *nadir* perspective, meaning the camera is oriented downwards. These include the *ManipallUAVid* dataset [2], the *Semantic Drone* dataset [12] and the *Swiss Drone and Okutama Drone* dataset [13]. However, to safely plan flight paths, to detect other aircraft and to monitor what lies ahead, this camera perspective is not suited.

There are also a few real-world datasets that are recorded with a front-facing camera mounted on a UAV. The *UAVid* dataset was presented by Lyu et al. in 2020 [10]. It has labeled frames extracted from 42 video sequences. Each of these sequences was recorded at a different location to increase the diversity of the dataset. The drone footage was recorded with a camera angled downwards at 45° from a flying altitude of approximately 50 m. Eight classes are distinguished, as shown in Table 1. The classes appear at different distances and thus different scales in the images. The *Aeroscapes* dataset was created by Nigam et al. and contains images with associated segmentation labels extracted from 141 video sequences [14]. The images were taken at altitudes from 5 to 50 m and with the drone camera pointing downwards with an angle. A total of 12 different classes are labeled. Another dataset containing images from an UAV with a tilted camera is the *Ruralscapes* dataset [9]. It contains 20 video sequences, from which 13 are designated for training ML models and 7 for testing them. Of the training videos, 816 frames are manually annotated, and of the testing videos, 311 frames are. In these annotations, a distinction is made between 12 classes. In the dataset, the drone is flying at different altitudes in a region of Romania that is characterized by mountainous terrain. In total, nearly 846 h of work were required to create the manual labels for this dataset [9]. Table 1 summarizes the presented real-world datasets from a UAV perspective with a forward-facing camera and provides a comparison to the synthetic dataset created in this work.

The number of synthetic datasets is even further limited compared to the number of real ones. Most of them come from the automotive sector and include *Virtual KITTI* [15],

Synthscares [16], *SYNTHIA* [7] as well as the dataset created by Richter et al. [4]. Compared to that, there are only a few synthetic semantic segmentation datasets from a UAV perspective. The *VALID* dataset was published by Chen et al. and contains aerial imagery taken from the nadir perspective in six different virtual environments under different weather conditions [17]. In addition to segmentation masks, the dataset contains information needed for object recognition as well as depth maps. Over the course of this research, only one synthetic dataset for semantic segmentation was found in which a front-facing camera is used. This is the so-called *Mid-Air* dataset [11]. In addition to the masks for semantic segmentation, it also contains depth maps and other information from sensors such as GPS or accelerometers. The images were generated using the *Unreal Engine* [5] and the plugin *AirSim* [18]. The camera images were extracted from 54 flown trajectories. To increase the variation of the data, flights were taken in two environments at different times of the year and in different weather conditions [11]. The labels masks make a distinction between 12 classes.

Table 1. Overview of real-world datasets from an UAV perspective with a forward-facing camera compared with our synthetically generated dataset. Listed are the camera orientation, number of annotated images, image resolution and classes for each dataset.

Dataset	Orientation	Images	Resolution	Classes
Aeroscapes [14]	Forward, Nadir	3.269	1280 × 720	Background, Person, Bike, Car, Drone, Boat, Animal, Obstacle, Construction, Vegetation, Road, Sky.
UAVid [10]	Forward	270	3840 × 2160	Building, Road, Static Car, Dynamic Car, Tree, Low Vegetation, Human, Background Clutter.
Ruralscapes [9]	Forward	1.047	3840 × 2160	Forrest, Residential, Land, Sky, Hill, Road, Church, Fence, Water, Car, Person, Haystack.
Ours	Forward	2.242	1920 × 1080	Tree, Grass, Building, Car, Human, Street, Other.

2.2. Training on Synthetic Images

Due to the problems of generating and labeling real-world images described above as well as the potential benefits of using synthetic images to train machine learning algorithms, there is an increasing amount of research focusing on this topic. Given the extensive number of synthetic datasets in the automotive field, most work is carried out in this domain. In the process of creating their dataset, Richter et al. [4] also explored the performance of semantic segmentation models on real-world evaluation datasets. When trained on real images mixed with synthetic ones, a mIoU (mean Intersection over Union) increase of 2.6 percentage points was achieved on the *KITTI* dataset [19]. Using the *CamVid* dataset [20], it was shown that a comparable performance to a model trained exclusively on real-world images can be achieved if only one third of the real-world training dataset is used in combination with synthetic images. The authors concluded that the amount of manually labeled training data can be greatly reduced by using synthetic data in the training process.

Due to the need of environment perception on UAVs and the rise of tools for extracting synthetic data from game engines, the usage of synthetic data from a UAV perspective has also received some attention in the last years. The authors of [21] conducted a study on the performance of object detection of cars from a low-flying UAV using the nadir perspective. They investigated the question whether the robustness of the detections can be improved by training with synthetic training images when the differences between available real datasets and the target datasets are too big. One model was trained with synthetic images, one with real images and a model pre-trained with synthetic data was fine-tuned with real data. During the evaluation, only very small differences between the

model trained with synthetic data and the model trained with real data could be detected. By fine-tuning the models, the domain gap was reduced and the robustness for object recognition increased [21]. In a study about object detection through instance segmentation using UAV images, the authors of [22] were able to train a Mask R-CNN model that can detect and segment pedestrians and vehicles. The model trained in this study was able to achieve good detection results on real-world drone footage, even without using any real data in the training process. The authors of [23] also studied the impact on Neural Network performance when real training images are replaced by synthetic images. They were able to demonstrate that models trained with synthetic data show good performance on test datasets, but poor performance on real-world datasets. This performance degradation could not be observed when a model was trained with real-world data and applied to other real datasets. In another experiment, synthetic and real data were mixed together to form a new training dataset. It was shown that the performance of the models on real images was higher than the performance without the data mixing. As a third experiment, a model pre-trained with synthetic data was re-trained with real-world data during a fine-tuning process. Here, it was shown that the performance increases when a larger number of real images is used for fine-tuning. However, none of these works considered semantic segmentation tasks.

3. Generation of Synthetic Images

In order to achieve a good semantic segmentation and the smallest possible sim-to-real gap, it is reasonable to assume that the synthetic training images should look as similar to the evaluation data as possible. As described above, the evaluation images come from the Ruralscapes dataset. Figure 2 shows exemplary scenes.



Figure 2. Comparison of exemplary scenes from the Ruralscapes dataset in the first row [9] with our reproduced scenarios within the simulation environment in the second row. The third row shows exemplary images from the UAVid dataset [10], which were also taken from a UAV perspective, but in a different geographic region.

The simulation environment designed and used in this work is created using the *Unreal Engine* [5] 4.27 and various additional contents from the *Unreal Marketplace*. Figure 3 shows the stages of building the simulation environment. The starting point is a previously created environment of a mountainous landscape which is then modified. For this purpose, objects such as houses and vegetation that do not look similar to the objects of the Ruralscapes dataset are removed from the environment, leaving only the terrain with the landscape materials. The environment is then modified by flattening the ground inside the valley to accommodate houses that are created later. Additionally, the mountains are reduced in size, smoothed and sharp edges and cliffs are removed to better resemble the hills in the Ruralscapes dataset. A small hill with a meadow and a stream are also added. Using the built-in Unreal Engine tools, road segments are created and automatically placed along a spline. Similar to the Ruralscapes dataset, the resulting road network consists of a main

road with lane markings and smaller streets without markings or sidewalks. Objects such as cars, lanterns, fences and people are placed along those. Houses with a wide range of shapes are also placed along the streets and textures of roofs and facades are adjusted to increase variation in the dataset. To extract the camera images and labels from the simulation environment, the plugin *AirSim* [18] is used. Figure 2 shows some exemplary images extracted from our simulation environment compared to those from the Ruralscapes dataset. It can be seen that an attempt was made to stylistically recreate some scenes from the Ruralscapes dataset and visual similarities can be seen clearly.



Figure 3. Steps to create the environment for the synthetic image dataset. From top left to bottom right: starting from a previously created environment (**top left**), inserting a road network and other objects (**top right**), creating various houses and inserting larger forests in the background (**bottom left**), adjusting the brightness and color tones (**bottom right**).

To use the labels extracted from the Unreal Engine using *AirSim*, some post-processing steps need to be performed. As *AirSim* outputs instance masks for the objects, these are post-processed in a first step to correspond to semantic segmentation masks. When comparing the masks of the synthetic dataset with those of the Ruralscapes dataset, it is noticeable that there is a large difference in the level of detail. This is especially noticeable for the trees, as shown in Figure 4. The annotations of the synthetic images are pixel-precise, while those of the real-world dataset are significantly coarser. In case of the trees, this is mainly shown by the fact that the background, that can be seen between the branches and leaves of the synthetic tree, is labeled as grass, whereas in the real-world dataset, all pixels within the silhouette of the tree are assigned to the class *tree*.



Figure 4. Comparison of the level of detail of the segmentation masks for the class *tree*. (**Left**): section of a camera image with the corresponding segmentation mask from the real-world Ruralscapes dataset [9]. (**Right**): camera image and segmentation mask of a tree from the synthetic dataset with the original fine labels and post-processed coarse labels.

If the training is performed with the fine labels, it is suspected that this will lead to worse results in terms of mIoU during the evaluation. Therefore, the level of detail of the trees in the labels of the synthetic dataset is reduced in order to increase the similarity to the labels of the Ruralscapes evaluation dataset as shown in Figure 4. For this, a binary mask was first created for each segmentation mask of the dataset, which only depicts the labeled trees. A MaxPool filter was applied to this mask to reduce the level of detail of the trees in the binary image depending on the kernel size. The trees in the modified binary image were then colored with the corresponding color of the class *tree* and overlaid with the original mask.

4. Experiments and Results

4.1. Model and Training Settings

For the following experiments, the *DeepLabv3 ResNet101* architecture presented by Chen et al. [24] is used. An implementation from the PyTorch model collection which is pre-trained on the COCO dataset [25] is used. The Neural Network uses a *ResNet101* [26] backbone to extract features and *Atrous Spatial Pyramid Pooling* (ASPP) [24] for the segmentation. As described in the paper, the ResNet101 backbone consists of a convolutional layer followed by four blocks, each block consisting of three convolutional and three batch normalization layers with skip connections. The fourth block uses atrous convolutional layers to increase the area of coverage of the filter [24]. The features for the semantic segmentation are extracted from four convolutional layers with different levels of dilation [27]. In addition, a pooling layer is used. The feature maps of these layers are concatenated. A final convolutional layer outputs the correct number of classes [24].

Before the images are used in the respective training run, a pre-processing of the images is performed. First, the images and masks are rescaled to 640 × 480 pixels to reduce the memory requirements. A nearest-neighbor interpolation is used to reduce the size of the mask to avoid blending of colors at the class boundaries when scaling. Furthermore, data augmentation is used to help to increase the ML model capability to generalize. In this work, we use random vertical flipping, sharpening the image, blurring, changing the brightness and changing the saturation. The batch size during training is eight. For the training of the models, a variable learning rate is used according to the algorithm by Smith and Topin [28]. Cross-Entropy Loss is used as error function and AdamW [29] as optimizer. The models are trained over 150 epochs and the models with the best validation losses are used for the final evaluation. As evaluation metric, *Intersection-over-Union* (IoU) is used which is calculated by dividing the *area of overlap* by the *area of union*. It describes the amount of overlap of two regions, respectively, two sets of pixels. Building on that, the *mean Intersection-over-Union* (mIoU) describes the average IoU over all distinguished classes. The per-class mIoU is calculated for each class as the average IoU of that class over all images of the evaluation dataset. All trained models are evaluated on the same images of the evaluation dataset.

4.2. Defining the Label

Since the real-world datasets distinguish between different segmentation classes, the labels have to be unified before training and evaluation. As the Ruralscapes dataset contains 12 but the UAVid dataset only 8 labeled classes, some of the classes have to be combined, as shown in Table 2. Labels for trees exist in both datasets. The classes *haystacks* and *hills* of the Ruralscapes dataset are included here because there are mostly trees growing in the regions labeled as hills. The *ground* class in the Ruralscapes dataset includes for example grass and is comparable to the *low vegetation* class of the UAVid dataset. In the synthetic dataset, these elements are grouped under the term *grass*. Both the UAVid and the Ruralscapes dataset feature annotated buildings. The latter additionally distinguishes between churches and inhabited buildings. These are summarized under the term *buildings*. People and road classes exist in both datasets. Both datasets have annotated vehicles. The UAVid dataset also differentiates between static and dynamic vehicles, which are summarized

here. The remaining classes of the Ruralscapes dataset *water*, *sky* and *fences* are combined into the class *other*. The synthetic classes are adapted to them.

Table 2. Combined labels of the datasets used in the experiments.

Ruralscapes	UAVid	Synthetic (Ours)
Forrest Hill Haystack	Tree	Tree
Land	Low vegetation	Grass
Residential Church	Building	Building
Car	Static car Dynamic car	Car
Person	Human	Human
Road	Road	Street
Fence Water Sky	Background Clutter	Other

4.3. Training on Ruralscapes Dataset

For the training of the model with real-world images from the Ruralscapes dataset, its training subset described above is used. Training a model on this data allows us to investigate how well the chosen model architecture can perform a semantic segmentation task when the training and evaluation data are from the same domain and, therefore, no domain shift is occurring. The average mIoU as well as the mIoUs for each segmented class are summarized in Table 3. As expected, the trained model performs well with an average mIoU of 60.9%. This performance is also apparent when examining the mIoUs of each individual class. The model performs particularly well when segmenting trees, grass and buildings, but has difficulties when segmenting cars. A possible reason for this may be that these classes usually only contain comparatively few pixels and, therefore, are underrepresented in the training process. Figure 5 shows exemplary segmentations by the model, compared to the ground truth mask. This confirms the results that were shown in the evaluation metric. The classes *building*, *tree* and *other* are segmented well, only the class boundaries are slightly more rounded in the prediction. In addition, smaller classes in the background are occasionally segmented incorrectly.

Table 3. Results on the Ruralscapes evaluation dataset. Shown are the mIoU and the per-class mIoU in percent. Bold numbers represent the best results. Legend: Ruralscapes = Ruralscapes training dataset. Synth = Our Synthetic training dataset. Synth. + X = Pre-training with the synthetic training dataset followed by a fine-tuning with X% of the Ruralscapes training dataset. UAVid = UAVid training dataset.

Training Data	Average mIoU [%]	per-Class mIoU [%]						
		Street	Building	Car	Human	Tree	Grass	Other
Ruralscapes	60.9	52.7	78.3	20.5	46.5	76.7	65.3	86.5
Synth	32.0	15.7	55.6	9.9	12.9	22.7	37.7	69.5
Synth + 1%	49.8	28.1	73.1	10.3	32.6	70.3	56.9	77.2
Synth + 2.5%	53.8	39.8	75.8	13.2	38.1	74.6	57.7	77.5
Synth + 5%	56.7	44.3	76.7	25.0	35.2	74.3	62.3	79.3
Synth + 10%	57.5	49.3	77.5	26.8	30.9	74.4	61.7	82.2

Table 3. Cont.

Training Data	Average mIoU [%]	per-Class mIoU [%]						
		Street	Building	Car	Human	Tree	Grass	Other
Synth + 25%	59.6	51.9	78.2	25.2	41.3	75.0	59.8	83.7
Synth + 50%	61.7	55.9	78.6	26.9	43.9	76.3	66.1	84.6
Synth + 100%	61.2	53.1	79.5	21.7	47.1	77.0	64.0	85.8
UAVid	31.7	14.9	48.7	14.0	0.0	54.8	44.2	45.4

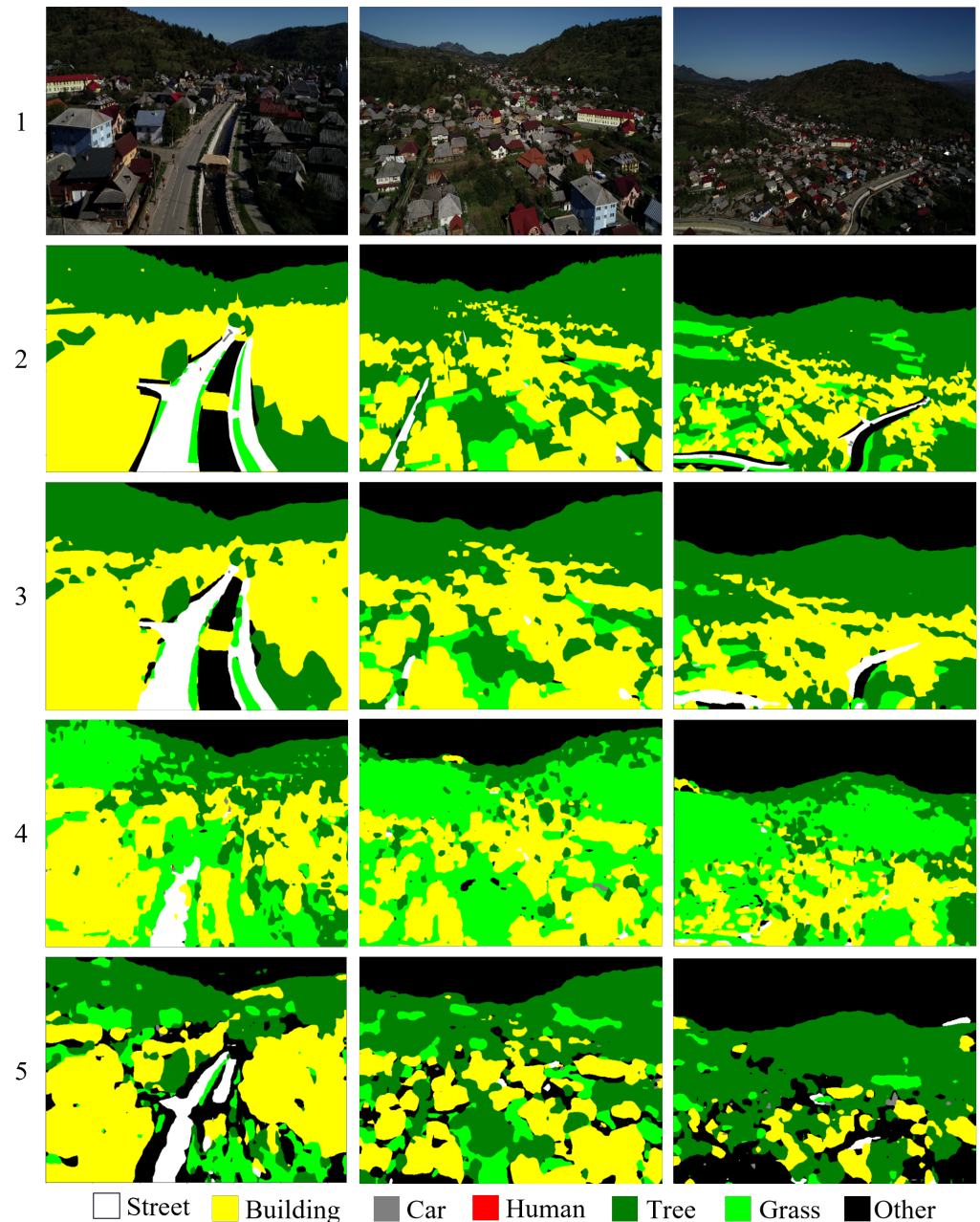


Figure 5. Segmentation results of the trained models from three different images taken from the evaluation dataset. First row: raw camera images from Ruralscapes dataset [9], second row: ground truth mask [9], third row: results of the model trained on Ruralscapes data, fourth row: results of the model trained synthetic data, fifth row: results of the model trained with UAVid data.

4.4. Training on Synthetic Dataset

The model trained with our synthetic images achieves an average mIoU of 32.0% on the Ruralscapes evaluation dataset. Therefore, it is 28.9 percentage points worse in this regard on the same evaluation images than the model trained on the Ruralscapes training data. Hence, a domain shift or sim-to-real gap is clearly evident. The evaluation metrics, are shown in Table 3. When looking at the mIoUs of the individual classes, it can be seen that each class is segmented considerably worse than from the model trained on Ruralscapes data. The model struggles most notably on cars and people. The classes *building* and *other* are segmented most accurately when compared to the other classes. Despite adjusting the level of detail for the trees in the labels, a good result cannot be achieved with respect to the mIoU. Some example predictions of the model are shown in Figure 5. From these it can be seen that especially fine details like cars, people and fences are not segmented correctly compared to the ground truth mask while larger areas such as buildings or the sky are predicted mostly correct. However, similar to the model trained with Ruralscapes data, with less detail than the ground truth. It is noticeable that the class grass is often predicted instead of trees.

4.5. Fine-Tuning

In the previous section, it was shown that the model trained on synthetic images performs worse on the real evaluation dataset than the model trained on real-world images of the same domain. We now investigate to what extent the performance of the synthetic model can be improved by fine-tuning it with Ruralscapes training data and whether the amount of real data needed can be reduced through this combination. This assumption is reasonable since the synthetic dataset created as part of this work bears a closer resemblance to the Ruralscapes evaluation dataset than the COCO dataset used to pre-train the models as described above. Fine-tuning over 150 epochs, where only the model with the lowest validation error is saved, is performed seven times, each time using a subset of different size of the Ruralscapes training dataset. The mIoUs for the trained models are listed in Table 3. A visual representation of the model performance with different amounts of real training data for the fine-tuning is given in Figure 6. Prediction results on an example image are given in Figure 7. As expected, the models perform better when more Ruralscapes data are used during the training process.

Looking at the individual classes, it is apparent that the model fine-tuned with 1% of the real-world training images already achieves considerable improvements with respect to the mIoUs compared to the model trained solely on synthetic images. Especially the classes *building*, *human*, *tree* and *grass* show an improved performance. However, the domain shift was quite large in these cases in the first place. For cars, there is no big difference compared to the domain shift without fine-tuning. This can be explained by the fact that the few images used for fine-tuning were selected randomly and it is possible that there are no or only a few pixels with the label *car* in these images.

When looking at larger sizes of the fine-tuning subset, it can be seen that the amount of real images required for training can be greatly reduced when pre-training on the synthetic images is performed. Even when using only 5% of the real images for fine-tuning, the mIoU differs by no more than five percentage points compared to the model trained on the whole real dataset. Furthermore, it is more than 11 percentage points better compared to the model not pre-trained on synthetic data. The models trained with subsets of size 25%, 50% and 100% show the best mIoUs for the fine-tuned models and are comparable to the model that is not pre-trained on synthetic images but trained exclusively with all Ruralscapes training images. These results, however, can be achieved with significantly fewer real images, so that the time-consuming labeling process for the real images can be greatly reduced.

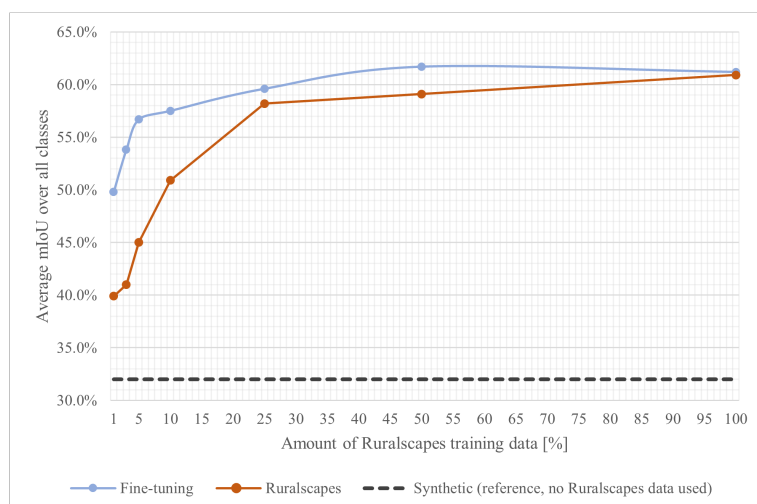


Figure 6. Changes in performance on the Ruralscapes evaluation dataset for the model trained on synthetic images with varying amounts of real training images during the fine-tuning process. The *Ruralscapes* line shows the performance of models trained only on Ruralscapes training data with different amounts of data without fine-tuning. Large dots on the *fine-tuning* and *Ruralscapes* curves show the performance of the trained models. The lines in-between those dots are linear interpolations. The horizontal line *Synthetic* illustrates the performance of the model trained on the synthetic dataset without additional real data as reference.

To verify that these results are because of the pre-training on the synthetic images and cannot be achieved using the real data alone, we trained the model also on same-sized subsets of the real images but without pre-training on the synthetic images. The results are shown in Figure 6. It can be seen that the model trained on the reduced subset without pre-training on the synthetic images performs worse than the pre-trained version. Therefore, the performance benefit seems to be a result of the pre-training on synthetic images. It follows that pre-training on synthetic data can help to reduce the needed amount of real-world data. This effect is even larger when only a small amount of real-world data are available.

While the pre-trained model performs much better when we do not have the whole Ruralscapes trainings dataset available, this effect diminishes the more real data we add. However, even when the whole real dataset is used, some minor improvements can be made by pre-training on synthetic images. The best overall evaluation results are obtained with the pre-trained model that is fine-tuned on the 50% subset of the Ruralscapes training dataset. It achieves an average mIoU of 61.7% which is still 0.8 percentage points better compared to the model trained on the whole Ruralscapes training dataset. Looking at the individual classes, it can be seen that for the classes *street*, *car* and *grass*, this model achieves the best mIoU scores of all the trained models. For the classes *human*, *building*, *grass* and *other*, it is only outperformed by the fine-tuning model with the whole Ruralscapes training dataset.

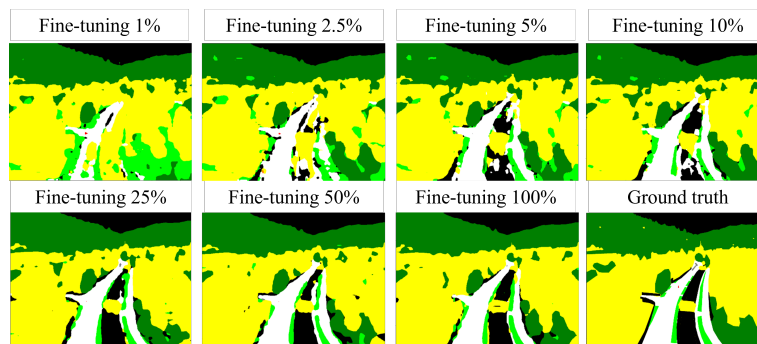


Figure 7. Predictions for a real evaluation image of the models fine-tuned with subsets of different sizes of the real training data. Ground Truth mask from [9]. White: street, yellow: building, grey: car, red: human, dark green: tree, light green: grass, black: other

4.6. Training on UAVid Dataset

Another question relevant in practice occurs when real-world data are available but from a different geographical region. Namely, whether it is better to use this dataset or to generate a synthetic dataset, which closely resembles the intended region. To answer this question, we also train a model on the UAVid dataset described above. In technical terms, this evaluates how large the domain shift from a different geographical region is compared to the sim-to-real gap.

After splitting the UAVid dataset, 180 images are used for training and 67 images for validation. Note that the size of the dataset is much smaller than the synthetic one. However, this is common for practical applications due to the difficulty and needed effort of generating and labeling the real-world images described above. The evaluation metrics of this model are shown in Table 3. The trained model achieves an average mIoU of 31.7% on the Ruralscapes evaluation dataset. It has problems especially with the segmentation of cars and people. The model does not detect the latter at all. The classes *other*, *building* and *grass* are classified comparatively well. Based on the exemplary segmentations of the model in Figure 5, it can be seen that the model classifies a lot of regions as *other*. This is possibly due to the labeling of the UAVid dataset. Sidewalks, for example, are not included in the class road but are labeled as *other*. Comparing the results, it can be seen that the average mIoUs of the individual classes are noticeably smaller than from the Ruralscapes model, i.e., there is a significant domain shift.

It can also be seen that the model trained on synthetic data performs slightly better than the model trained on the UAVid data. The synthetic model performs better on four classes while the UAVid models performs better on three classes. Technical speaking, the domain shift is comparable to the sim-to-real gap but slightly worse. For practical applications, a slight benefit can be achieved by using a synthetic dataset which closely resembles the intended domain but a real-world dataset from a different region might perform almost the same.

5. Discussion

The presented results show that the use of only synthetic training images obtained from a simulation environment is currently not sufficient to train a good Neural Network for semantic segmentation to be applied on real-world camera images. Technically speaking, there is a severe sim-to-real gap. However, by applying fine-tuning with a small set of real-world images, the gap can be significantly reduced.

One of the assumed factors for the performance drop is that the level of detail of the labels of the Ruralscapes dataset differs significantly from that of the labels of the synthetic dataset. For the manual labels of the Ruralscapes dataset, it can be seen that the quality of the masks sometimes vary greatly for consecutive frames. This makes a correct semantic segmentation of the frames difficult for the model.

In addition, hills where trees are visible have been labeled as *hill* in the Ruralscapes dataset, even though large patches of trees are on them and thus could also be mostly labeled as *tree* but not solely. Due to this simplification, the ML model may not be able to learn correct features. It is also important to note here that, as mentioned above, because of the dominance of the trees in the hill class, we added the hill class to the *tree* class. This was performed to unify the labels between the used datasets. However, this obviously does not minimize the described problem of label quality. In contrast, the automatically generated labels from the synthetic dataset naturally distinguish between all objects. When applying the model trained on the synthetic images to real evaluation data, it can therefore happen that it detects classes which exist in the image but are not labeled in the ground truth. Since they are not labeled in the ground truth mask, this is considered a misclassification which has a negative impact on the mIoUs described in Table 3. Likewise for other classes, inaccurate drawings of class boundaries could result in incorrect ground truths, making the metrics described in the previous section less conclusive.

To validate this statement, some images of the Ruralscapes dataset were chosen and relabeled. Figure 8 compares the relabeled mask with the original one. Here, the significant differences in the level of detail are visible. In addition to the distinctions between classes on the hill, there are significantly more trees labeled between houses and the class boundaries show a greater amount of detail. The mIoU of the class *grass* thus might be distorted and could turn out better by using "correct" labels compared to the ones described in the previous sections. The model also segments regions between buildings that are not included in the original labels but clearly stand out in the relabeled image. Therefore, it is suggested that the quality of the models in terms of the evaluation metrics can be increased if the entire Ruralscapes dataset is relabeled with more detailed masks, which thereby more closely resemble the exact synthetic masks and overall level of detail of reality.

Another possible cause for incorrect segmentation by the synthetic model is that the synthetic images used for training differ too much from the real ones. Differences can, e.g., be seen in textures. For example, the roads have markings in the simulation environment but not in the real images. Other properties of the synthetic images, such as image noise, exposure, or the hue of the images, may also have an impact on the domain shift.

Especially based on the segmentation results of the UAVid model, it is suspected that the class distributions in the datasets also have significant effects on the segmentation results. Thus, it can be assumed that this effect also occurs for the model trained with synthetic training data and that it has an impact on the domain shift. In the synthetic dataset, the number of pixels with the class *grass* is greater than in the Ruralscapes dataset. This class is predicted more frequently by the model with synthetic training data. Classes which could only hardly be detected, such as people, roads or buildings located in the background, are under-represented in the synthetic dataset compared to the Ruralscapes dataset. However, it is not possible to make a definite statement as to whether this is true for all classes, since, for example, fences are not segmented well from the synthetic model, even though the *other* class, which includes fences, occurs often in the synthetic dataset. To make a definite statement, the class would have to be relabeled.

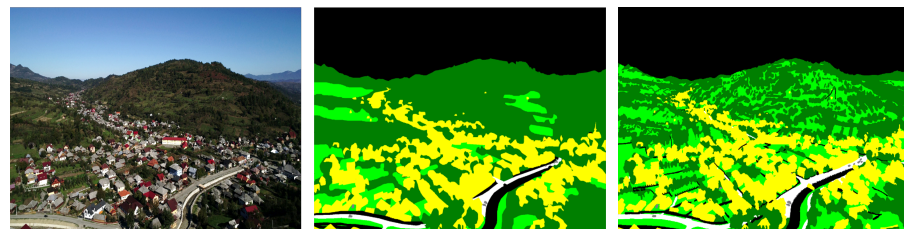


Figure 8. Comparison of the ground truth mask from an image of the Ruralscapes evaluation dataset [9] with a manually relabeled mask. (Left): video frame (increased brightness for better visualization), (center): ground truth label, (right): manually relabeled mask. White: street, yellow: building, grey: car, red: human, dark green: tree, light green: grass, black: other

6. Conclusions and Future Work

This paper shows that synthetic images generated with a game engine provide benefits for the training of Neural Networks for semantic segmentation of the environment of an UAV. By automatically generating training images with associated pixel-precise labels, a large dataset can be created in a short period of time. The limitations and disadvantages which occur when creating a real-world dataset from UAV perspective, such as costs and regulatory barriers, can thus be avoided. This holds true for almost all use-cases in which visual sensor data for UAV environment perception is needed.

By using them to pre-train a model and afterwards fine-tuning it on real-world data, the amount of needed real data can be reduced and the model performance can be improved slightly. Overall, comparable results to those obtained with the real training dataset alone can already be achieved by pre-training on synthetic images and fine-tuning it with only half or even a quarter of the real training data.

We also show that synthetic images alone do not yield good results on real images yet. A severe domain shift or more precise sim-to-real gap can be observed. Although the assessment of the sim-to-real gap shown in this work is difficult due to the differences or even lack of detail in the labels of the real images, the basic problem and tendency is clearly recognizable.

Furthermore, we show that synthetic images that resemble the target domain perform slightly better than using images from a UAV perspective from a different geographical region. If the effort required to create a simulation environment is smaller than that needed to generate and manually label a dataset acquired in the real world, synthetic images can be a suitable alternative that provides similar results while reducing the cost and time required. Overall, significant cost and time savings could be achieved by using synthetic training images from a game engine, as long as the effort required to create the simulation environment is low.

Our findings are consistent with those for semantic segmentation from the automotive sector. For street scenes, Ref. [4] was able to show that a comparable mIoU increase can be achieved when adding synthetic training data. They also showed that comparable performance to a model trained exclusively with real-world images can be achieved when only one-third of the real images is used in the training dataset in combination with synthetic ones. The authors concluded that the amount of manually labeled training data can thus be greatly reduced by using synthetic data in the training process. Similarly, Ref. [16] trained a model for semantic segmentation on synthetic images and showed that it achieves worse results on real images compared to a model trained on real images. When fine-tuning the model afterwards with real images, the results are better for most classes than when only real data are used during training. Our work adds the connection to the aviation sector and shows that these findings are transferable to cameras mounted on UAVs despite differences such as camera perspective, huge scale variations and distance to objects mentioned above. We also show that the sim-to-real gap exists also for these situations even though one could argue that texture differences may have been smaller because of the altitude of the UAV and the resulting lack of perceived details of textures.

All in all, future research on the definite influences on the sim-to-real gap appears to be useful and needed. For example, the differences on image-level between the synthetic and the real images can be investigated. Furthermore, it can be assessed whether the realism of the simulation environment can be increased further and whether it has a positive impact on the sim-to-real gap. There are also a number of possible mitigation strategies, mainly from the overlaying problem of domain shift, that can be applied to the problem at hand and can be evaluated. This can help to make synthetic data more usable in the aviation sector and, therefore, allow to develop better environment perception models, which ultimately could allow UAVs to fly autonomously.

Author Contributions: Conceptualization, J.R.; methodology, C.H. and J.R.; software, C.H.; validation, C.H.; investigation, C.H. and J.R.; data curation, C.H. and J.R.; writing—original draft preparation, C.H.; writing—review and editing, J.R.; visualization, C.H.; supervision, J.R. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The generated synthetic dataset will be made publicly available once this paper is accepted. The link is <https://zenodo.org/record/8133761> (accessed on 22 June 2023). The Ruralscapes dataset is a third-party dataset and is publicly available at <https://sites.google.com/site/aerialimageunderstanding/semantics-through-time-semi-supervised-segmentation-of-aerial-videos> (accessed on 21 October 2022), see [9]. The UAVid dataset is a third-party dataset and is publicly available at <https://uavid.nl> (accessed on 21 October 2022), see [10].

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Perritt, H.; Sprague, E. *Domesticating Drones: The Technology, Law, and Economics of Unmanned Aircraft*; Routledge: Abingdon, Oxon, UK; New York, NY, USA 2017.
2. Girisha, S.; Pai, M.M.M.; Verma, U.; Pai, R.M. Performance Analysis of Semantic Segmentation Algorithms for Finely Annotated New UAV Aerial Video Dataset (ManipalUAVid). *IEEE Access* **2019**, *7*, 136239–136253. [[CrossRef](#)]
3. Alberti, E.; Tavera, A.; Masone, C.; Caputo, B. IDDA: A Large-Scale Multi-Domain Dataset for Autonomous Driving. *IEEE Robot. Autom. Lett.* **2020**, *5*, 5526–5533. [[CrossRef](#)]
4. Richter, S.R.; Vineet, V.; Roth, S.; Koltun, V. Playing for Data: Ground Truth from Computer Games. *arXiv* **2016**, arXiv:1608.02192. <https://doi.org/10.48550/ARXIV.1608.02192>.
5. Epic Games Inc. Unreal Engine. Available online: <https://www.unrealengine.com/en-US> (accessed on 19 June 2023).
6. Unity Technologies. Unity. Available online: <https://www.unity.com> (accessed on 12 April 2023).
7. Ros, G.; Sellart, L.; Materzynska, J.; Vazquez, D.; Lopez, A.M. 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), Las Vegas, NV, USA, 27–30 June 2016; pp. 3234–3243. [[CrossRef](#)]
8. Kiefer, B.; Ott, D.; Zell, A. Leveraging Synthetic Data in Object Detection on Unmanned Aerial Vehicles. *arXiv* **2021**, arXiv:2112.12252. <https://doi.org/10.48550/ARXIV.2112.12252>.
9. Marcu, A.; Licaret, V.; Costea, D.; Leordeanu, M. Semantics through Time: Semi-supervised Segmentation of Aerial Videos with Iterative Label Propagation. In Proceedings of the Asian Conference on Computer Vision, Kyoto, Japan, 30 November–4 December 2020; Corresponding Image Dataset. Available online: <https://sites.google.com/site/aerialimageunderstanding/semantics-through-time-semi-supervised-segmentation-of-aerial-videos> (accessed on 21 October 2022).
10. Lyu, Y.; Vosselman, G.; Xia, G.S.; Yilmaz, A.; Yang, M.Y. UAVid: A semantic segmentation dataset for UAV imagery. *ISPRS J. Photogramm. Remote Sens.* **2020**, *165*, 108–119. Corresponding Image Dataset. Available online: <https://uavid.nl> (accessed on 21 October 2022). [[CrossRef](#)]
11. Fonder, M.; Van Droogenbroeck, M. Mid-Air: A Multi-Modal Dataset for Extremely Low Altitude Drone Flights. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Long Beach, CA, USA, 16–17 June 2019; pp. 553–562. [[CrossRef](#)]
12. TU Graz, Institute of Computer Graphics and Vision. Semantic Drone Dataset. Available online: <https://www.tugraz.at/index.php?id=22387> (accessed on 16 January 2023).
13. Speth, S.; Alves Gonçalves, A.; Rigault, B.; Suzuki, S.; Bouazizi, M.; Matsuo, Y.; Prendinger, H. Deep learning with RGB and thermal images onboard a drone for monitoring operations. *J. Field Robot.* **2022**, *39*, 840–868. [[CrossRef](#)]
14. Nigam, I.; Huang, C.; Ramanan, D. Ensemble Knowledge Transfer for Semantic Segmentation. In Proceedings of the 2018 IEEE Winter Conference on Applications of Computer Vision (WACV), Lake Tahoe, NV, USA, 12–15 March 2018; pp. 1499–1508. [[CrossRef](#)]
15. Gaidon, A.; Wang, Q.; Cabon, Y.; Vig, E. Virtual Worlds as Proxy for Multi-Object Tracking Analysis. *arXiv* **2016**, arXiv:1605.06457. <https://doi.org/10.48550/ARXIV.1605.06457>.
16. Wrenninge, M.; Unger, J. Synscapes: A Photorealistic Synthetic Dataset for Street Scene Parsing. *arXiv* **2018**, arXiv:1810.08705. <https://doi.org/10.48550/ARXIV.1810.08705>.
17. Chen, L.; Liu, F.; Zhao, Y.; Wang, W.; Yuan, X.; Zhu, J. VALID: A Comprehensive Virtual Aerial Image Dataset. In Proceedings of the 2020 IEEE International Conference on Robotics and Automation (ICRA), Paris, France, 31 May–31 August 2020; pp. 2009–2016. [[CrossRef](#)]
18. Shah, S.; Dey, D.; Lovett, C.; Kapoor, A. AirSim: High-Fidelity Visual and Physical Simulation for Autonomous Vehicles. *arXiv*, **2017**, arXiv:1705.05065. <https://doi.org/10.48550/ARXIV.1705.05065>.
19. Geiger, A.; Lenz, P.; Stiller, C.; Urtasun, R. Vision meets robotics: The KITTI dataset. *Int. J. Robot. Res.* **2013**, *32*, 1231–1237. [[CrossRef](#)]
20. Brostow, G.J.; Fauqueur, J.; Cipolla, R. Semantic object classes in video: A high-definition ground truth database. *Pattern Recognit. Lett.* **2009**, *30*, 88–97. [[CrossRef](#)]
21. Konen, K.; Hecking, T. 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), Laguna Hills, CA, USA, 1–3 December 2021; pp. 1–8. [[CrossRef](#)]
22. Viana, F.X.; Araujo, G.M.; Pinto, M.F.; Colares, J.; Haddad, D.B. Aerial Image Instance Segmentation Through Synthetic Data Using Deep Learning. *Learn. Nonlinear Model.* **2020**, *18*, 35–46. [[CrossRef](#)]
23. Nowruz, F.E.; Kapoor, P.; Kolhatkar, D.; Hassanat, F.A.; Laganieri, 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. <https://doi.org/10.48550/ARXIV.1907.07061>.
24. Chen, L.C.; Papandreou, G.; Schroff, F.; Adam, H. Rethinking Atrous Convolution for Semantic Image Segmentation. *arXiv* **2017**, arXiv:1706.05587. <https://doi.org/10.48550/ARXIV.1706.05587>.

25. Lin, T.Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; Zitnick, C.L. Microsoft coco: Common objects in context. In *Proceedings of the European Conference on Computer Vision (ECCV)*; Springer: Berlin/Heidelberg, Germany, 2014; pp. 740–755. Corresponding Website. Available online: <https://cocodataset.org/> (accessed on 21 October 2022).
26. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778. [[CrossRef](#)]
27. Chen, L.; Papandreou, G.; Kokkinos, I.; Murphy, K.; Yuille, A.L. DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs. *arXiv* **2016**, arXiv:1606.00915.
28. Smith, L.N.; Topin, N. Super-Convergence: Very Fast Training of Neural Networks Using Large Learning Rates. *arXiv* **2017**, arXiv:1708.07120. <https://doi.org/10.48550/ARXIV.1708.07120>.
29. Loshchilov, I.; Hutter, F. Decoupled Weight Decay Regularization. *arXiv* **2017**, arXiv:1711.05101. <https://doi.org/10.48550/ARXIV.1711.05101>.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.