




On exploring weakly supervised domain adaptation strategies for semantic segmentation using synthetic data

Roberto Alcover-Couso¹  · Juan C. SanMiguel¹ · Marcos Escudero-Viñolo¹ · Alvaro Garcia-Martin¹

Received: 8 March 2022 / Revised: 15 June 2022 / Accepted: 3 February 2023
© The Author(s) 2023

Abstract

Pixel-wise image segmentation is key for many Computer Vision applications. The training of deep neural networks for this task has expensive pixel-level annotation requirements, thus, motivating a growing interest on synthetic data to provide unlimited data and its annotations. In this paper, we focus on the generation and application of synthetic data as representative training corpuses for semantic segmentation of urban scenes. First, we propose a synthetic data generation protocol, which identifies key features affecting performance and provides datasets with variable complexity. Second, we adapt two popular weakly supervised domain adaptation approaches (combined training, fine-tuning) to employ synthetic and real data. Moreover, we analyze several backbone models, real/synthetic datasets and their proportions when combined. Third, we propose a new curriculum learning strategy to employ several synthetic and real datasets. Our major findings suggest the high performance impact of pace and order of synthetic and real data presentation, achieving state of the art results for well-known models. The results by training with the proposed dataset outperform popular alternatives, thus demonstrating the effectiveness of the proposed protocol. Our code and dataset are available at http://www-vpu.eps.uam.es/publications/WSDA_semantic/

Keywords Synthetic data · Semantic segmentation · Domain adaptation · Weakly supervised domain adaptation

✉ Roberto Alcover-Couso
roberto.alcover@uam.es

Juan C. SanMiguel
juancarlos.sanmiguel@uam.es

Marcos Escudero-Viñolo
marcos.escudero@uam.es

Alvaro Garcia-Martin
alvaro.garcia@uam.es

¹ Universidad Autonoma de Madrid, Escuela Politecnica Superior, Madrid, Madrid, Spain

1 Introduction

Semantic segmentation refers to the task of classifying each pixel in a given image to its semantic category, providing pixel-level masks of images. This task is specially interesting for urban scenes scenarios, where accurate pixel-wise understanding of the image can help in many applications, such as autonomous driving and robot vision.

Deep Neural Networks (DNN) have proven their efficacy for segmentation, being the current state of the art in different tasks, such as semantic segmentation [41], X-ray lung segmentation [53], Brain MR image segmentation [54] and video object segmentation [72]. However, DNN training depends on extensive amounts of labeled images which are expensive to label. To this end, the usage of synthetic data for training neural networks becomes an alternative to get large corpuses at a reduced cost. Therefore, many efforts have shifted the focus onto synthetic data as a plausible solution, [47, 65], using 3D environments with models of semantic objects to classify, [18, 21, 46]. Although promising, synthetic images often present different visual appearance than real images: light reflection, color saturation and shadows make synthetic images distinguishable from the real ones. Specially on urban scenes, where synthetic images are not able to capture the wide variability of real images (e.g., light conditions). While many approaches have been proposed to obtain synthetic data, [6, 7, 20, 25, 50] no methodological generation of synthetic data has been defined for the specific task of semantic segmentation. Thus, a methodology for obtaining synthetic data at different complexity levels would be desirable.

Regarding the visual appearance discrepancy between real and synthetic images, Domain Adaptation (DA) encompasses the class of techniques that extrapolate generalities obtained from synthetic data to real domain. Two main research lines can be differentiated depending on whether or not real images ground truth is available during training. Weakly Supervised Domain Adaptation focuses on abstracting knowledge from both domains [57, 69, 81], using a combination of extensive amounts of labeled data from synthetic images and a small amount of labeled real images to refine the segmentation DNN. Differently, Unsupervised Domain Adaptation (UDA), tries to generalize to the real data without relying on labeled real images by aligning features from both domains [16, 22, 63, 64, 74]. Commonly UDAs literature focus on generating domain agnostic features by aligning outputs from different levels of the model. However, such alignment is defined for specific layers of the model, hence, impeding the straightforward extrapolation of successful proposals to different architectures. Furthermore, most works follow a random image presentation during training due to the absence of a definition for a sample-based complexity in semantic segmentation. However, smart image presentation protocols have proven effective in other fields [2]. Therefore, comparing the existing approaches and defining a curriculum (applicable to semantic segmentation employing different sources of synthetic and real data) would be beneficial to understand and improve the performance of real data on its own.

In this paper, we address the above-mentioned limitations for semantic segmentation (1) by proposing a protocol for synthetic data generation; (2) by analyzing popular strategies for training and transfer learning using real and synthetic data; (3) and also by defining a curriculum-based strategy to effectively combine multiple sources of synthetic data. We employ the simulation tool Multi-camera System Simulator (*MSS*) [24] to generate synthetic sets with several configuration options (e.g., number of classes, viewpoints). The synthetic data generated by our protocol is compared against widely used datasets [46, 47] for different training strategies such as combined training [43], and fine-tuning [57]; and also for different proportions of real and synthetic data. Moreover, we proposed a curriculum-based

learning strategy relying on the hypothesis that an increasing-complexity data feeding strategy would generalize better to the target real data than a standard-paced (i.e. random) strategy using the same data. We take advantage of our protocol to generate datasets with increasing complexity (defined as the number of instances in the dataset) and use these datasets for curriculum learning. The experimental results show how not only the proposed generation protocol outperforms existing synthetic datasets, but how the combination of different sources and structuring of the training process in an incremental complexity manner can improve state of the art performance. It is noteworthy to highlight that the proposed data generation protocol and training strategies can be applied to any architecture for aligning different synthetic domains to real data, without relying on specific alignment terms.

The contributions of the proposed approach are:

- A new design protocol for synthetic data generation based on virtual scenario simulators.
- Identifying and comparing training strategies for weakly supervised domain adaptation in semantic segmentation, measuring the impact of different synthetic sources and different proportions of real data.
- Proposing a new strategy based on curriculum learning for employing different sources of data, applicable to DNN-based approaches for semantic segmentation.

The paper is organized as follows. Section 2 reviews the state-of-the-art on domain adaptation and synthetic data usage for semantic segmentation. Section 3 introduces the criteria and design protocol for synthetic data generation. Section 4 describes the selected strategies for weakly supervised domain adaptation. Section 5 presents the experimental results, including a comparison with the state-of-the-art. Finally, conclusion remarks are described in Section 6.

2 Related work

2.1 Domain adaptation

A basic assumption in machine learning is having the training and test data sampled independently from an identical distribution. In the context of domain adaptation this assumption is not fulfilled [28], having two domains with clear visual discrepancies: the source data, used for training, and target data, used for fine grained training and testing. Therefore, direct training on the source data leads to a significant performance drop on the target test set. This hindrance is commonly known as domain shift.

Single-source domain adaptation Alternatively to alleviate the domain shift one may extrapolate knowledge from synthetic to real images in the training process of the model. Depending on whether the ground truth of real images is available during training, this can be further classified into Unsupervised Domain Adaptation (UDA), if not labeled real images are available during training, and Weakly-Supervised Domain Adaptation (WSDA), if a small set of labeled real images are available during training. UDA frameworks employ target RGB images during training to align features from both domains [36, 63, 64, 74, 83]. However, some other works show that effective extrapolation to the real domain can be obtained even without including any real image during training by increasing gradually the complexity of the sample images in a curriculum manner [17, 27, 37]. Following this idea,

we propose a synthetic dataset generation protocol to aid the straight-forward implementation of an easy-to-hard image presentation for semantic segmentation. WSDA approaches [9, 57, 69, 81] deploy high performance models for real data adapting from abundant labeled synthetic images but scarce and insufficient labeled real data. Other approaches follow some sort of adversarial learning strategy, [25, 62], which characterizes for the inclusion of an additional, —usually small—, discriminator network which tries to discriminate from the segmentation maps if the input RGB image is real or synthetic. However, adversarial training is generally known as a difficult task due to its instability [63], hence, we do not consider these approaches for this work and will only be used for comparative purposes.

In different computer vision fields such as object localization, some works obtain state of the art performance by defining an easy to hard presentation commonly known as curriculum learning [2]. Nowruzi, E. et al. [43] studied the impact of the real data size in weakly supervised object localization. Similarly, Zheng, Q. et al. [57] studied the impact of different pacing strategies when using different ratios of real images. Following this line of work, our proposal aims at further structuring the pacing showcase of different sources of data when compared to the typical finetuning strategy and combined training. To the best of our knowledge, there is no other similar study for semantic segmentation.

Multiple-source domain adaptation In practice, the source labeled data may come from different domains, such as different simulators for synthetic data or day and night images for real domains, this motivates the research of Multiple Source Domain Adaptation (MSDA) techniques. However, multiple source combined training usually leads to worst performance models than employing one single source for training [19, 77]. In order to overcome this limitation, many authors focus on aligning features from all source domains with target domain features [3, 11, 17, 28, 37, 50, 51, 62, 63, 66, 74, 81]. Three common methods for aligning features in single and multiple source domain adaptation are:

- **Discrepancy based:** These alignment frameworks focus on minimizing an explicit distance measure between features obtained in the target and source domains [60]. Various distribution discrepancy metrics have been introduced, including Maximal Mean Discrepancy (MMD) [26], Correlation Alignment [56] and Wasserstein distance [1]. MMD is currently the most widely used metric to measure the distance between two feature distributions [35].
- **Adversarial based:** These proposals rely on the inclusion of a discriminator model which measures how domain-discriminative the features generated by the segmentator are. Following the typical adversarial scheme of GANs [25, 62], this training paradigm becomes a min-max game, where the segmentator model aims at fooling the discriminator. In essence, by minimizing the performance of the discriminator the segmentator is minimizing the gap between domains in the feature space [3, 28, 50, 62–64, 74].

Recently, this idea has been made more explicit by adversarial methods defining strategies to translate image appearance from one domain to another. These proposals tend to combine adversarial training with an additional term of consistency. This consistency term measures the discrepancy between the output produced from the original image and the translated image [76]. Intuitively, these proposals minimize the domain shift by aligning features across real and synthetic domain, while maximizing the performance in a mutual domain.

- **Entropy based:** These works minimize the entropy on the target domain. As the labels of the target images are not available during training, minimizing the entropy is in a way a self supervision mechanism. Being C as the number of classes, H and W the height

and width of the input image, Y_t the one-hot encoded C -vector label and the prediction entropy, E , $E = \sum_C \sum_{H,W} P(X_t) \log(P(X_t))$ and the classical cross-entropy loss [11, 63]:

$$L_{seg} = \sum_C \sum_{H,W} Y_t \log(P(X_t)) \quad (1)$$

Summary of domain adaptation proposals Table 1 summarizes the explored proposals dealing with Domain Adaptation in computer vision. All the proposals for semantic segmentation [3, 11, 28, 37, 50, 51, 62, 63, 66, 74, 81] tasks employ deeplab and/or FCN as the segmentator of choice. Hence, for a fair comparison, we employ both architectures in our experiments.

In particular, we propose to define a protocol to sequentially include different source domains to generalize in a more stable and reliable manner than adversarial and entropy based frameworks [63, 64, 74]. Furthermore, when comparing our protocol to discrepancy based frameworks [35, 60] we do not impose normality nor homogeneity, hence, providing a more relaxed framework extrapolable to any MSDA problem. Alignment free adaptation has proven useful in other computer vision fields such as Object detection. Hintertoisser. D. et al. [27], show that effective extrapolation to the real domain can be obtained without employing any alignment metric to the real images. Specifically, domain adaptation is obtained by generating increasingly complex synthetic images, modifying the scale and point of view of a 3D model of the target object over random backgrounds, thereby, generalizing to the real domain only by structuring the training. In this work, we propose a similar protocol for urban scenes segmentation. We argue that by generating increasingly complex

Table 1 Summary of state of art domain adaptation proposals

Method	Multi-Source	Source	Target	Task	Alignment	Supervision
Zheng et al [81]	×	Synthetic	Real	S	A	Unsupervised
Zhang et al [76]	×	Synthetic	Real	S	C	Unsupervised
Zhang et al [74]	×	Synthetic	Real	S	D	Unsupervised
Toldo et al [60]	×	Synthetic	Real	S	D & E	Unsupervised
Russo et al [48]	✓	Synthetic	Real	S	A	Unsupervised
Zhao et al [78]	✓	Synthetic	Real	S	A	Unsupervised
Hinterstoisser et al [27]	✓	Synthetic	Real	L	–	Unsupervised
Gong et al [23]	✓	Real	Real	Clas	–	Weakly
Saito et al [49]	×	Real	Real	Clas	A & E	Weakly
Doersch et al [17]	×	Synthetic	Real	L	–	Weakly
Zheng et al [81]	×	Synthetic	Real	S	A	Weakly
Wang et al [67]	×	Synthetic	Real	S	A	Weakly
Wen et al [69]	×	Synthetic	Real	S	C	Weakly
Kumar et al [33]	✓	Real	Real	S	–	Weakly
Sun et al [57]	✓	Synthetic	Real	S	A	Weakly
Ours	✓	Synthetic	Real	S	–	Weakly

(KEY: A: Adversarial, C: Consistency, D: Discrepancy, E: Entropy, -: No alignment, S: Image Segmentation, L: Object detection, Clas: Image Classification).

images by modifying factors such as the foreground and background scale, the capture point of view and the number of types of objects, we can define an effective curriculum that generalizes better to the real domain.

2.2 Curriculum learning

Bengio et al. [2] inspired by schooling principles, proposed to train machine learning algorithms by training with basic (easy) samples sooner and the advanced (hard) samples later. In order to define which samples should be included first and which should be included in the training last, Curriculum Learning (CL) needs to define some sort of complexity measure. This complexity can target different hyper-parameters or inputs of the training process such as the target task and the performance measure [55]. However, the complexity is typically measured on a sample-basis, and, as the training continues, the probability to select hard samples for training is increased. In the context of sample-based complexity, different approaches to measure complexity have been proposed, with manual annotation [30, 45] and the performance of a teacher model (e.g., generally a model that has been trained in a standard fashion and is used to probe the samples complexity) [23] as the most used strategies.

In the context of domain adaptation, sample-based curriculum learning has been effectively employed in different tasks such as object detection [27], sentiment classification [80] and image classification [71]. These sample-based curricula can be further classified into sampling-focused [27] and weighting-focused [71, 80]. Weighting-focused curricula attempts to weight the importance of each sample in a batch depending on the training stage, e.g., giving smaller weights to harder samples at its beginning. Luyu et al. [71] and Sicheng et al. [80] propose to train a *Manager* function which —given an input batch, outputs a scalar as a weight for each sample. These frameworks have the drawback of the computational overhead required for training the *Manager*. On the other hand, sampling-focused curricula use predefined sample complexities and attempt to automatically select the optimal set of samples given the current *status* of the model, e.g., somehow defining binary weights for each sample. Hinterstoisser et al. [27] attempt to define a sampling curriculum by generating increasingly complex synthetic images for object localization. By defining a formulation for object localization in terms of scale and rotation angle of a 3D model of a target object, they are capable of generating a sample-based curriculum in which the sample complexity is quantified by these factors. In a similar manner, we propose to generate a dataset which is structured in different levels of sample complexity for semantic segmentation to define a sample-based curriculum for semantic segmentation.

Although employing some sort of curriculum into semantic segmentation has been already attempted, a sample-based curricula for semantic segmentation is yet unattempted. Whereas [74, 75] attempt a curriculum for semantic segmentation by degrading the output of the segmentator from the label distribution to a pixel-wise classification. This curriculum is defined on a task-level, rather than on a sample-basis. Furthermore, they focus on a single source unsupervised domain adaptation compared to our multi-source weakly supervised domain adaptation framework.

2.3 Synthetic data generation for semantic segmentation

Generative-based Some authors propose to enhance synthetic images realism from simulator to alleviate the domain shift [7, 59, 69]. To this aim, a style transfer architecture is

trained to generate new images, following a generator-discriminator approach established by [25]. Although promising, one problem is still unsolved: the generator network can hallucinate new objects [6], which as the ground truth is not modified, will not be present in the ground truth map. Furthermore, requires additional computational efforts to train the generator DNN and currently presents a worse performance than training with synthetic data from simulators [76].

Simulation-based Two different sources of synthetic data are commonly used, the *GTAV* [46], the *Synthia* [47]. Figure 1 includes visual examples of *GTAV*, *Synthia* and the proposed *MSS* datasets, illustrating distinct light reflection, textures and design for their representations. *GTAV* is composed of 25K images from the game Grand Theft Auto V. In contrast with the other synthetic datasets, *GTAV* is composed from individual images rather than video sequences. *Synthia* is generated with a virtual camera placed on a virtual car driving through the city with different weather conditions. All synthetic datasets present images with an appearance aligned to sequences of the *Cityscapes* dataset. However, *GTAV* and *Synthia* do not provide a predefined structure of complexity, impeding the formalization of an incrementally easy to hard presentation of images [2, 27]. In this situation, the *MSS* provides a functionality to address the above mentioned shortcoming. While the code for annotating a *GTAV* like dataset is publicly available, one have to buy the original game in order to be able to render and annotate images, hence, impeding the universal usage of the engine.

2.4 Real datasets for semantic segmentation

In semantic segmentation, three real datasets are usually selected as target domain: *Kitti* [21], *Cityscapes* [14] and *Mapillary* [42], Fig. 2 includes visual examples for real datasets (*Kitti*, *Cityscapes* and *Mapillary*). In spite of depicting all urban scenarios, these datasets present inherent visual discrepancies attributed to the different geographic location of each dataset, the car models, the street disposition, the traffic lights and the buildings architecture. Furthermore, each dataset follows distinct design criteria: *Kitty* provides a smaller dataset with frames captured from the top of a car unlike with other datasets which were filmed with a camera inside of a car. *Kitty* visually differs not only because of camera position, but due to the lighting conditions, where burnt patches can be found on some of the instances due to a bigger exposure of the camera and drastic light changes from turns of the car facing



(a) *GTAV* with a car egocentric point of view in an open space with a straight angle. (b) *Synthia* with a car egocentric point of view in a road with a straight angle. (c) *MSS* with a fixed camera point of view in an open space with an upward angle. (d) *MSS* with a fixed camera point of view in a turn with a downward angle.

Fig. 1 Images of *GTAV*, *Synthia* and *MSS* datasets with different capture point of views and spatial distributions



(a) Kitti with a car (b) *Cityscapes* with a car (c) *Mapillary* with a car (d) *Mapillary* with a pedestrian
 egocentric point of view in a centered point of view in a centered point of view in a centered point of view
 of road with a straight angle. of road with a straight angle. of road with a downward angle. of road with a straight angle.

Fig. 2 Images of *Kitti*, *Cityscapes* and *Mapillary* datasets with different capture point of views and spatial distributions

the sun light directly. *Cityscapes* was generated by filming with a camera inside of a Mercedes while driving through different German cities. This design implicitly brings unique biases, such as having the Mercedes Logo and the front of the car always at the bottom of the picture. In addition, due to the camera position, little to no sky is present in the frames, in contrast with the other real datasets. *Mapillary* on the other hand was generated by filming with different points of view. Most of the sequences were filmed inside a car looking straight through the windshield, however different capturing angles were used, in contrast with *Cityscapes*, in which the position and angle is consistent through the dataset. In addition, some sequences are filmed from a pedestrian, motorcycle and a touristic bus point of view. Almost 90% of the *Mapillary* dataset was filmed from road/sidewalk views in urban areas, the remaining ones are from highways, rural areas and off-road. When comparing real datasets further discrepancies can be found such as: *Cityscapes* presenting fewer poles when compared to *Mapillary*. This is due to the *Cityscapes* presenting cities where the wiring is located underground, unlike *Mapillary* which is obtained from cities where the city wiring tends to be supported by utility poles. Due to the small size of *Kitti*, 500 images, typically it is only used for testing in the literature. In this work we follow this pattern by only using this dataset to assert some hypothesis and not for training.

Figure 3 illustrates how state of art datasets lack a predefined structure of complexity for semantic segmentation, as all datasets are captured using similar points of views with similar objects scales and scene distribution.

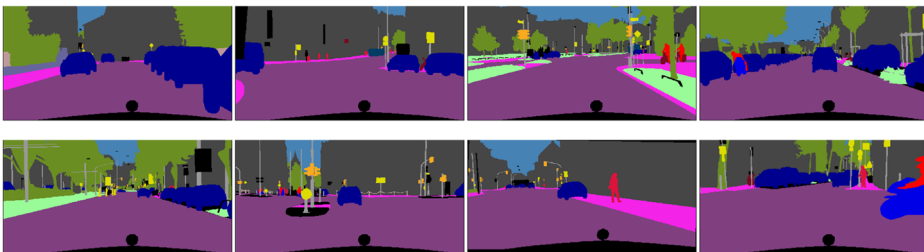


Fig. 3 Samples of GT labels of existing popular semantic segmentation datasets for urban scenes: *Cityscapes*, *Mapillary*, *GTAV*, *Synthia*. Top row presents synthetic images, bottom row presents real images

3 Synthetic dataset based on the MSS simulator

As previously mentioned, current real and synthetic datasets lack a procedure for a generation of images of increasing complexity. Here, we discuss about the criteria for such generation and the obtained synthetic dataset.

3.1 Design criteria

Before creating the dataset, we need to understand how the point of view of the camera may impact the extrapolation to the real domain, which may help to define a design criteria for the *MSS* dataset. The *MSS* simulator offers a high degree of freedom for camera placement by allowing wearable cameras on moving objects such as helicopters, cars or pedestrians, and also fixed cameras on specific points in the virtual city. We focus on aligning the point of view to the ones to the target (real) datasets, so synthetic sequences may look similar to the real ones. Table 2 includes the results of an early experiment for training a Deeplab V3 architecture [12] with a dataset composed of one synthetic sequence and a small subset of real images. This experiment allows understanding which capturing point of view has less domain gap as compared to the real domain. As seen in Table 2, we find that fixed cameras and wearable cameras from cars egocentric point of view provided greater improvements than other alternatives. Meanwhile, wearable cameras from pedestrian, helicopter and bus point of view provide points of views which are not present in the target sets. Therefore, training with these sequences yields a worse performance on the target validation set.

By further analysis of the impact of wearable cameras and fixed cameras, see Fig. 4, we find that wearable cameras provide more diversity due to the changing background. However, there is scarcity of some urban elements that are generally less common than straight road sections in the cities, such as turns, roundabouts and intersections. This scarcity turned into models with poor results on unseen spatial allocations, such as intersections where the sidewalk is divided by a road lane without continuation. On the other hand, placing fixed cameras on less common spatial allocations, leads to a better representation of them, however it results into models less accurate on common scenarios as compared to the one trained with car wearable camera sequences.

Table 2 Preliminary study on the impact of the inclusion of each type of sequence to a 5% random selection of the *Cityscapes* and *Mapillary* train sets for training a Deeplab V3 architecture [12]

Cityscapes			Mapillary		
Sequence	Wearable	MIoU	Sequence	Wearable	MIoU
Fixed	×	0.34	Fixed	×	0.36
Pedestrian	✓	0.32	Pedestrian	✓	0.30
Helicopter	✓	0.30	Helicopter	✓	0.30
Car	✓	0.34	Car	✓	0.36
Bus	✓	0.30	Bus	✓	0.29

Tested on their respective validation sets

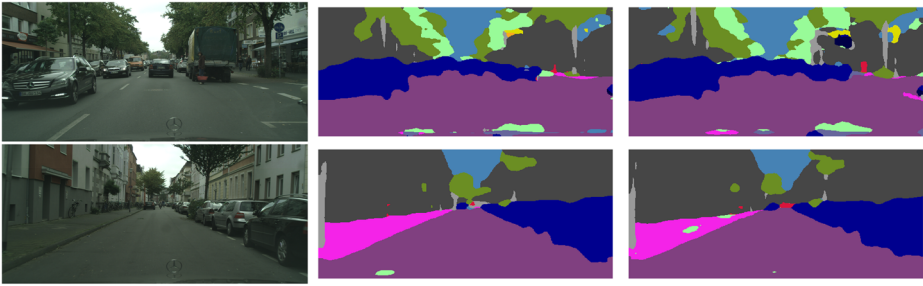


Fig. 4 RGB validation images (left column), results of training with fixed cameras (middle column) and wearable cameras from cars (right column)

The proposed dataset contains sequences from the following types: fixed, car and pedestrian. These sequences are created according to the proposed protocol explained in the next section

3.2 Protocol

The *MSS* dataset is structured into subsets by the amount of objects present in the virtual scenario, with aims at creating subsets with an increasing complexity. This protocol is inspired by [27] applied to object detection, which accomplishes an increase in performance by sorting the order in which images are shown to the model during training. We hypothesize that learning the general structure of urban scenes can be facilitated by starting with easy examples with few hard instances such as cars, poles and pedestrians. Later, the learned urban layout can be refined with more complex examples. This data ordering gives the model a distinct advantage over using a protocol where the model needs to understand the whole structure from scratch. We captured sequences in the virtual scenario with different points of views and different amount of cars and pedestrians present in the virtual scenario. This allowed a classification of the sequences by point of view and amount of cars and pedestrians. This categorization of the sequences provides an straightforward implementation of a learning protocol where complexity is periodically increased through the inclusion of more complex training examples.

Complexity is parametrized in order of importance as follows: First, the amount of moving agents in the scenario. The amount of moving agents refers to the amount of vehicles and pedestrians acting in the virtual city while filming the sequence. This parameter regulates the maximum amount of non static objects present per image, ranging from 50 to 750. Second, the amount of included points of view. Straight poses are easier to understand for CNNs compared to rotations [27]. Therefore, straight views are considered easier and are predominant in the less complex datasets, while more complex ones include different scales of objects and different points of view. The harder the sequence is intended to be, the further the camera is placed, in terms of degrees with respect to the road and meters from the agents, ranging from +70 to -70 degrees. This change in the camera angle affects the shape and appearance of objects, hence, increasing the complexity [27]. Finally, the predominant spatial distribution of the sequences. The spatial distribution is graded by the type of sequences employed, ranging from 0-40% of wearable cameras. Pedestrian wearable cameras include predominantly buildings and sidewalks rather than the predominant centered road distribution. Furthermore, pedestrian wearable cameras present wider

rotation freedom when compared to a car, which can only turn on specific points of the scenario, making the sequences less stable and harder to extrapolate to the general spatial distribution found when driving a car, see Table 2. Figure 5 includes RGB images (Fig. 5a–c) and labels (Fig. 5d–h) generated following the described protocol and generated using the *MSS* simulator. Figure 5d and e provide examples of the effect of increasing the complexity by modifying the background, i.e., samples with a more diverse background but the same amount of foreground elements. Figure 5e and f exemplify an increase in complexity by foreground, specifically, an increase in complexity is achieved by introducing more foreground elements (cars and pedestrians) in various scales while keeping the same background objects. Figure 5g and h display modifications on the complexity of a scene without modifying its elements, here an increase in complexity is obtained by moving the camera closer or further away from the scene, evenly changing the scale of all the objects.

Table 3 details the criteria for determining the complexity levels for each *MSS* subset. Table 4 provides a comparison with related datasets in terms of the proportion of labeled semantic classes. Table 5 presents the comparison of points of views with related datasets. As we can see, the proposed dataset has similar proportions of labeled data as compared

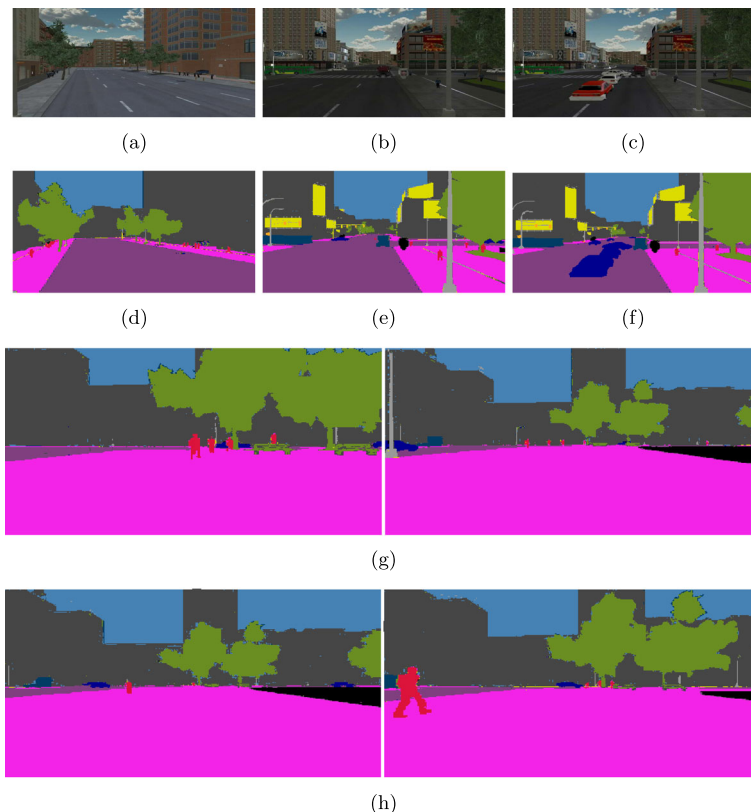


Fig. 5 Comparison of proposed synthetic GT labels of a,b,c) RGB images of the samples, d) Easy complexity image GT e) Medium complexity image GT f) Hard complexity image GT g & h) Same background different foreground scale and population

Table 3 Complexity factors present on each of the generated (MSS_i) and studied datasets

Complexity factor	Dataset	MSS ₅₀	MSS ₁₀₀	MSS ₂₅₀	MSS ₇₅₀	Synthia	GTAV	Cityscapes	Mapillary
Instances of objects	Bus	Low	Low	Low	Low	Medium	Low	Medium	Low
	Pedestrian	Low	Low	Low	Medium	Medium	Low	Medium	Medium
	Car	Medium	Medium	Medium	Medium	High	Very High	High	Very High
Point of view	Wearable Car	✓	✓	✓	✓	✓	✓	✓	✓
	Fixed camera	✓	✓	✓	✓	✓	✓	✓	✓
	Wearable pedestrian			✓	✓	✓			✓
Spatial bias	Centered road	High	High	High	High	High	High	High	High
	Turn	Low	Low	Medium	High	High	High	High	High
	Open space	Very Low	Low	Low	Medium	High	Low	Very Low	Very Low
Camera angle	Straight	✓	✓	✓	✓	✓	✓	✓	✓
	Downwards			✓	✓	✓			✓
	Upwards					✓			
Number of semantic labels		11	11	11	11	11	16	19	19
									152

Subsets are differentiated by the subindex after *MSS*, this subindex refers to the amount of instances present in the virtual scenario

Table 4 Summary table comparing the generated dataset with current state of the art datasets by representation of each of the MSS classes

Dataset	Image size	Frames	Proportion of pixels per class											
			Unlabeled	road	sidewalk	building	sign	pole	light	vegetation	sky	pedestrian	car	bus
<i>MSS₅₀</i>	480 × 640	61229	.00	.35	.05	.37	.03	.01	.01	.05	.10	.00	.03	.01
<i>MSS₁₀₀</i>	480 × 640	39430	.00	.39	.03	.33	.02	.01	.01	.07	.12	.01	.02	.00
<i>MSS₂₅₀</i>	480 × 640	30481	.00	.29	.11	.40	.01	.01	.00	.08	.06	.01	.02	.01
<i>MSS₇₅₀</i>	480 × 640	27390	.00	.05	.05	.30	.04	.01	.00	.24	.23	.02	.04	.02
<i>MSS_{Full}</i>	480 × 640	200000	.00	.37	.06	.35	.02	.01	.01	.07	.09	.01	.02	.01
<i>Synthia</i>	480 × 640	220000	.02	.29	.03	.20	.00	.01	.00	.21	.13	.00	.09	.01
<i>GTA</i>	1052 × 1914	25000	.05	.43	.08	.17	.00	.01	.00	.11	.08	.01	.04	.02
<i>Mapillary</i>	1024 × 2048	25000	.04	.16	.03	.16	.02	.02	.00	.19	.31	.01	.05	.02
<i>Cityscapes</i>	1024 × 2048	5000	.04	.49	.03	.16	.01	.01	.00	.14	.02	.02	.08	.01
<i>Kitty</i>	480 × 640	500	.02	.36	.02	.06	.00	.01	.00	.32	.11	.00	.09	.01

Subsets are differentiated by the subindex after *MSS*, this subindex refers to the amount of instances present in the virtual city

Table 5 Comparison of real and synthetic datasets for urban scenes segmentation

Name	Type	Open source	Point of view					Free view	Number of images
			Wearable cameras		Fixed view				
			Ground	Aerial	Ground	Aerial			
Kitti	Real	—	✓					200	
Cityscapes	Real	—	✓					25000	
Mapillary	Real	—	✓		✓			25000	
VKITTI2	Hybrid	×	✓					21260	
Synthia	Synthetic	×	✓		✓			220000	
GTAV	Synthetic	×	✓		✓			25000	
MSS _{full}	Synthetic	✓	✓	×	✓	✓	✓	200000	

The open source column indicates whether the source code is publicly available (✓) or not (×), for real datasets as there is no source code a - is assigned

to real/synthetic datasets and also allows ranking sequences/subsets by their complexity, which is not available in existing datasets.

4 Weakly-supervised strategies for training

Three different strategies are studied to handle weakly supervised domain adaptation: direct combined training, fine-tuning and curriculum learning. The first two strategies are derived from [43, 59, 61], we aim at mimicking scenarios where there is little available data from the target to train. Finally, we propose a curriculum learning strategy aiming at understanding how all synthetic sources can be used in conjunction to reduce the domain gap with respect to the real domain.

4.1 Combined training

Inspired by [43, 59, 61], this strategy trains with only a fraction of the real data in combination with all synthetic data. This strategy allows to measure the impact of each synthetic dataset and also which quantity of real data for training is sufficient to get an acceptable performance level. We train from scratch with a percentage of the real data varying from 5% to 100% of the original dataset mixed with the full synthetic dataset. The results are measured by testing on the corresponding real validation set. When using combined training sets, we expect the model to learn the general concepts from simulated images, and use the real samples to adapt. However, there is no scheduling nor structure in the combined training approach: samples from synthetic and real data are presented at a random pace. Therefore, the fulfillment of these expectations is not guaranteed.

4.2 Fine-tuning

Inspired by [43, 59, 61], this strategy consists on four stages. First, we start with a model with a backbone randomly initialized. Second, we train the full model with synthetic data

until convergence, following the procedure described in [43]. Third, we proceed by freezing the weights of the backbone and training the classifier head with the real images until convergence. Finally, we unfreeze the backbone, reduce the learning rate (lr_0) by 10 and train with the real subset until the validation MIoU stalls. Figure 6 depicts a graphical representation of the fine-tuning strategy. As the combined training strategy, we also consider different fractions of the full real datasets and perform testing using the real data validation set.

4.3 Curriculum learning

Inspired by curriculum learning for object detection and image classification [27, 30, 32, 33, 45], we propose a new curriculum strategy based on progressively feeding the different datasets and subsets to the model sorted by a predefined complexity order. The proposed complexity is defined by the number of semantic classes present on each synthetic set, and the complexity of each of the datasets. Formally, let $X_s, s \in [1, N]$ be each source dataset, with N the number of synthetic datasets, $\{x_s, y_s\}_{i=1}^{n_s} \in X_s$ the input images and their ground truth tensors, composed of one one-hot encoded C -length vector label per image pixel, respectively. n_s the number of labeled samples for dataset X_s .

Being Ω the trainable set of parameters of the segmentation architecture and x the input sample image, the prediction probability is obtained by $G(x; \Omega) = P_x \in (0, 1)^C$, so that $\sum_{c=1}^C P_x^{h,b,c} = 1$ for any $(h, b) \in ([0, H] \times [0, B])$, where C are the number of semantic classes and $H \times B$ is the image size.

Ω are optimized through stochastic gradient descent by minimizing the cross-entropy loss, (1):

$$\Omega_{t+1} = \Omega_t - lr_{step} \sum_{i=1}^n \nabla L_{seg}(x_i, y_i; \Omega_t)$$

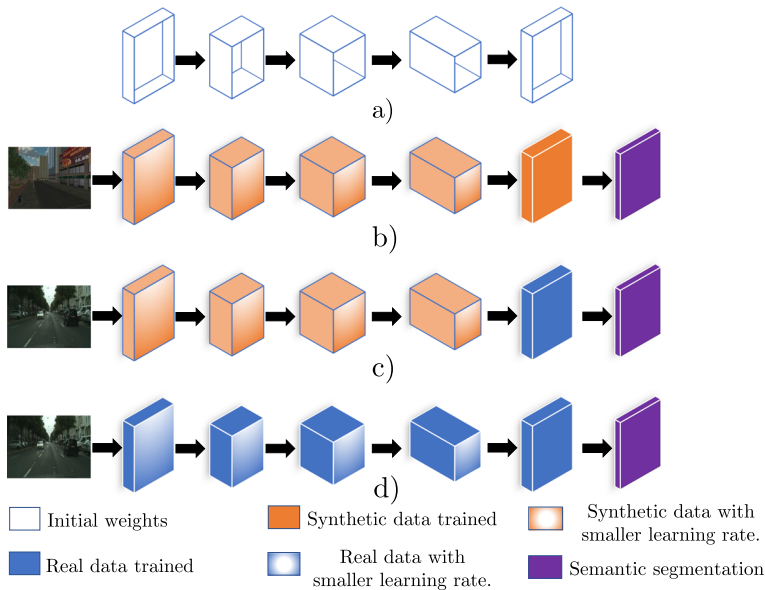


Fig. 6 Fine-tuning strategy stages. a) Initial Semantic segmentation model with randomly initialized backbone. b) Training of the full model using synthetic data. c) Backbone freezing and classifier head training with real data. d) Training of the full model using real data with a learning rate ten times smaller than b)

The whole training process is controlled by three hyper-parameters, γ , lr_0 and β : We define a training subset $X_{\{\beta_s^{step}\}_{s=1}^N} = \bigcup_{s=1}^N \bigcup_{i=1}^{n_s \times \beta_s^{step}} \{x_s, y_s\}$, as one containing only the easiest β_s^{step} proportion of each dataset s , β with ranging from 0 – 100%, these proportions are modified at each curriculum step $\tau \in [1, N]$.

At each curriculum step $\tau \in [1, N]$ G is trained with $X_{\{\beta_s^\tau\}_{s=1}^N}$ until convergence with learning rate $lr_\tau = lr_0 \times \gamma^\tau$, being β_0, lr_0 and γ methods' hyper-parameters, the proportion of sampels used at each step is defined by:

$$\beta_s^\tau = \begin{cases} (\beta_0)^{(s-\tau)} & \text{if } s \leq \tau \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

This *dataset addition and re-training* is repeated until each dataset has been used for training. Finally, once all the training stages are concluded, we perform a final fine-tuning stage, as depicted previously, with only the target real training dataset until convergence.

Algorithm 1 summarizes the steps of this training strategy

```

Require: Labeled synthetic datasets  $\mathcal{S} = \mathcal{S}_{1,\dots,N}$ 
Require: Target real labeled dataset =  $T$ 
Require: Model  $G = \{backbone, classifier\}$ 
Require: Initial learning rate  $lr_0$ 
Require:  $0 \leq \gamma \leq 1$ 
Require:  $0 \leq \beta \leq 1$ 
   $lr \leftarrow lr_0$ 
  Included sets  $Included \leftarrow$  empty set
  for dataset  $\in \mathcal{S}$  do ▷ Curriculum loop
     $Included \leftarrow Included \cup$  dataset
    Train  $G$  with  $Included$ 
     $Rearrange \leftarrow$  empty set ▷ Change proportions
    for subset  $\in Included$  do
       $Rearrange \leftarrow Rearrange \cup \beta\%subset$ 
    end for
     $lr \leftarrow lr \cdot \gamma$ 
     $Included \leftarrow Rearrange$ 
  end for
   $lr \leftarrow lr_0$  ▷ Finetune Stage
  for layer  $\in backbone$  do ▷ Backbone freezing
     $layer.requires\_grad \leftarrow False$ 
  end for
  Train  $G$  with  $T$ 
  for layer  $\in backbone$  do ▷ Backbone un-freezing
     $layer.requires\_grad \leftarrow True$ 
  end for
   $lr \leftarrow lr_0 \cdot 0.1$ 
  Train  $G$  with  $T$ 

```

Algorithm 1 Curriculum learning procedure for domain adaptation in semantic segmentation.

5 Experimental results

We provide an analysis of the semantic gap between real and synthetic domains (Section 5.2). Then, we validate the utility of the MSS dataset for combined training (Section 5.3) and fine-tuning (Section 5.4). We also compare the results against Synthia, the largest publicly available synthetic dataset similar in size to the proposed MSS dataset. We present the results for the proposed curriculum learning strategy (Section 5.5), that combines several synthetic datasets employed in the literature. Finally, best achieved results are compared with the state of the art (Section 5.6) followed by a discussion of mayor findings and a qualitative comparisons (Section 5.7).

5.1 Experimental setup

Two performance evaluation metrics are employed: Mean Intersection over Union (MIoU) and Mean Pixel Accuracy (MPA) [44]. MIoU is widely used to measure the similarity between two subsets as the area of their overlap against the union of the areas of each subset. MPA refers to the percentage of correctly classified pixels. For all strategies, the common and specific training parameters are detailed in Table 6. These parameters have been determined considering similar state-of-art proposals [10, 12, 40].

Five datasets of data are employed: *Mapillary* and *Cityscapes* compose the real sources whereas *MSS*, *Synthia* and *GTAV* compose the synthetic sources. See Section 2 for further details. In the different experiments, we also make use of proportions for each dataset (see Table 7) to assess the performance impact for varying-size sets of data.

5.2 Baseline: training with only real or synthetic data

As baseline for further comparisons, we consider models trained from scratch with only one source of data. In this experiment we analyze the domain gap between pairs of train-test datasets (synthetic-real and real-real). Table 8 includes the results of training a Deeplab V3 architecture [10] using only one source dataset until the MIoU stalls on their own validation set [70]. Then, we validate on each of the real target validation sets. These results show a noticeable drop in performance when testing on a different validation set rather than the source one. The only one which consistently presents better results on all three test sets is

Table 6 Training configuration

Image size	400 × 800
Backbone	ResNet 101
Optimizer	SGD
lr_0	1e-4
Weight decay	1e-5
Step size	5
Scheduler	Reduce on Plateau
Batch size	13
Loss function	Cross entropy
Weighted loss	Effective #samples [15]
Data augmentations	Color jitter & Random horizontal flip
Number of epoch	50

Table 7 Size and proportions of datasets

Dataset	Domain	Percentage	#Images	
<i>Cityscapes</i>		100	2975	
		50	1488	
	Real	25	744	
		15	446	
		5	149	
<i>Mapillary</i>		100	25000	
		50	12500	
	Real	25	6250	
		15	3750	
		5	1250	
<i>MSS</i>	Synthetic	100	200000	
	Synthetic	50	100000	
	Synthetic	25	50000	
	Synthetic	15	30000	
	Synthetic	5	10000	
<i>Synthia</i>	Synthetic	100	220000	
	Synthetic	50	110000	
	Synthetic	25	55000	
	Synthetic	15	33000	
	Synthetic	5	11000	
The images are selected randomly	<i>GTAV</i>	Synthetic	100	25000

the *Mapillary* train dataset. We believe that the better transfer capabilities of the *Mapillary* case is mainly due to the big size gap between the datasets, refer to Table 7 for a size comparison. When comparing synthetic sets, we find that *MSS* dataset presents a smaller domain gap when compared to the *Synthia* dataset (see Table 8). Additionally, it can be seen how the combination of both synthetic datasets, *All synthetic*, outperforms using any of the synthetic datasets. Despite the clear domain shift between real and synthetic data, the performances obtained training only synthetic data do not differ drastically from the ones obtained when using different real datasets as source and target domains, a situation that has been already observed in object detection [43]. In the context of this paper, aiming to improve semantic segmentation in urban scenarios, these differences are aggravated by factors such as the high diversity in car models, street appearances and lighting conditions between source and target datasets.

Impact of the synthetic dataset size As we are proposing a new dataset which includes over 200K new synthetic images, we want to ensure that the amount of generated images is relevant. To that aim, we measure the impact of employing only a subset of samples in the synthetic sets, ranging from 5% up to 100% of samples for each of the studied datasets. Figure 7 represents the impact of the synthetic dataset size employed in the downstream performance of the model trained on the target real data. It can be seen how as the number of images is increased, the downstream performance is also increased for all the studied

Table 8 Results of training Deeplab V3 with a ResNet101 backbone [10] with one source and testing on each of the real validation sets

Train		Test		MIoU	MAP
Type	Source	Type	Target		
Real	<i>Cityscapes</i>	Real	<i>Cityscapes</i>	76.2	97.7
Real	<i>Mapillary</i>	Real	<i>Cityscapes</i>	39.1	86.9
Real	<i>Kitty</i>	Real	<i>Cityscapes</i>	17.0	48.4
Synthetic	<i>MSS</i>	Real	<i>Cityscapes</i>	17.2	67.2
Synthetic	<i>Synthia</i>	Real	<i>Cityscapes</i>	16.4	48.8
Synthetic	<i>All Synthetic</i>	Real	<i>Cityscapes</i>	20.1	70.1
Real	<i>Mapillary</i>	Real	<i>Mapillary</i>	42.3	87.8
Real	<i>Cityscapes</i>	Real	<i>Mapillary</i>	30.0	80.4
Real	<i>Kitty</i>	Real	<i>Mapillary</i>	15.5	44.2
Synthetic	<i>MSS</i>	Real	<i>Mapillary</i>	18.8	67.0
Synthetic	<i>Synthia</i>	Real	<i>Mapillary</i>	16.2	45.2
Synthetic	<i>All Synthetic</i>	Real	<i>Mapillary</i>	22.2	71.0
Real	<i>Kitty</i>	Real	<i>Kitty</i>	31.5	85.6
Real	<i>Mapillary</i>	Real	<i>Kitty</i>	29.9	87.0
Real	<i>Cityscapes</i>	Real	<i>Kitty</i>	18.4	60.9
Synthetic	<i>MSS</i>	Real	<i>Kitty</i>	16.3	43.4
Synthetic	<i>Synthia</i>	Real	<i>Kitty</i>	16.2	43.2
Synthetic	<i>All Synthetic</i>	Real	<i>Kitty</i>	25.2	79.6

(KEY. *MSS*: multi-camera System Simulator, *All Synthetic*: both *Synthia* and *MSS*)

synthetic datasets. However the gain in performance is not linearly correlated with the size. For instance, for both synthetic datasets, employing 5 times more of data yields an 13.2% increase in performance (from 17 to 19.5 MIoU), while employing 10 times more samples provides a 21.7% increase in performance (from 17 to 21 MIoU).

5.3 Combined training: concurrent synthetic-real data usage

We apply the combined training strategy defined in the Section 4.1 to train a Deeplab V3 with a ResNet101 backbone [10]. We employ proportions of the synthetic datasets for assessing their influence on the final performance (see Table 7). As for testing, we use only the real validation sets. Tables 9 and 10 compile the performances obtained by training with different proportions of the *Cityscapes* and *Mapillary* datasets respectively in conjunction with the complete synthetic datasets, similar results are obtained when employing a Fully Convolutional Network (FCN) [40]. This initial experiment indicates that the *MSS* dataset can compete favorably with *Synthia*, specially in scenarios where less real data is provided (see Table 9). Differently, as indicated by the performances in Table 10. Furthermore, we can see how the drop in performance is not linear to the amount of real data employed, employing 50% of the target data reduces the performance in less than 15% for both studied real datasets.

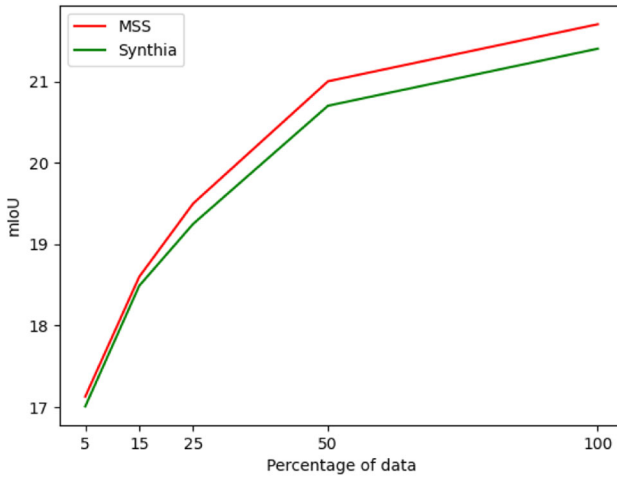


Fig. 7 Impact of synthetic dataset size for baseline training using a DeeplabV3, tested on the target validation set *Cityscapes*

5.4 Fine-tuning: using pre-training from synthetic data

We apply the fine-tuning strategy defined in the Section 4.2 to train two architectures: Deeplab V3 with a ResNet101 backbone [10] and Fully Convolutional Network (FCN) [40]. We employ proportions of the synthetic datasets for assessing their influence on the final performance as defined in Table 7. As for testing, we use only the real validation sets. Consistently with the previous experiment, Fig. 8a and b illustrate that the proposed MSS-based dataset outperforms the *Synthia* dataset on both the *Cityscapes* and *Mapillary* datasets when it is used to train a model that is later refined with real data. Furthermore, in this experiment it is shown that the finetuned models provide better performance than the combined training strategy (see Tables 9 and 10) and the baseline (see Table 8). *All synthetic* refers to the combination of *Synthia* and *MSS*.

Table 9 Combined training with *Cityscapes* using a Deeplab V3 with a ResNet101 backbone [10], tested on the *Cityscapes* validation set

Synthetic	Real %	MAP	MIoU	Δ
—	100	97.7	76.2	0
Synthia	5	89.1	42.2	− 34.1
MSS	5	89.8	42.7	− 33.8
Synthia	15	92.9	45.3	− 31.0
MSS	15	94.7	52.9	− 23.4
Synthia	25	95.6	54.0	− 22.3
MSS	25	95.8	57.3	− 19.0
Synthia	50	96.4	63.2	− 13.1
MSS	50	96.5	64.4	− 11.9
Synthia	100	97.2	70.1	− 6.2
MSS	100	97.2	70.4	− 5.9

Δ refers to the difference to the baseline

Table 10 Combined training with *Mapillary* using a Deeplab V3 with a ResNet101 backbone [10], tested on the *Mapillary* validation set

	Synthetic	Real %	MAP	MIoU	Δ
	—	100	89.8	42.3	0
	Synthia	5	85.8	32.1	- 10.2
	MSS	5	86.0	32.8	- 9.5
	Synthia	15	86.3	33.1	- 9.2
	MSS	15	86.7	33.4	- 8.9
	Synthia	25	86.9	34.4	- 7.9
	MSS	25	87.0	34.6	- 7.7
	Synthia	50	89.0	41.1	- 1.2
	MSS	50	89.2	42.2	- 0.1
	Synthia	100	89.4	42.4	+0.2
	MSS	100	89.6	42.6	+0.4

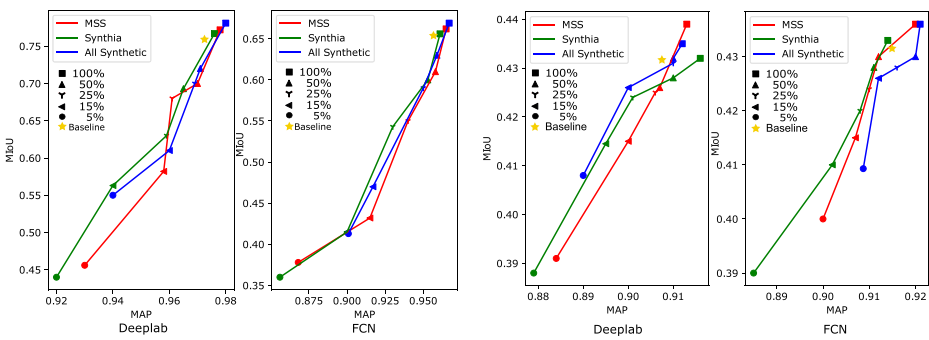
Δ refers to the difference to the baseline

For both real datasets, using an initial training on synthetic data to transfer knowledge to the real domain proves beneficial when compared to the baseline case (i.e. training only with real data). Furthermore, we observe the non-linear relationship between the percentage of real data introduced and the performance gain. With only a 5% of real data we can get up to 70% of the performance compared to training with the full dataset.

Impact of the synthetic dataset size In order to measure the impact of synthetic dataset size, Fig. 9 agglutinates the impact in the performance on real-data validation of the number of dataset samples used together with real data for training two models, one for each of the explored synthetic datasets. We can see how consistently employing larger datasets provides a better performance, hence, motivating the usage of our proposed dataset.

5.5 Curriculum learning

We explore the application of a new strategy based on Curriculum Learning (see Section 4.3) to train two architectures: Deeplab V3 with a ResNet101 backbone [10] and a FCN [40].



(a) Target set: *Cityscapes*

(b) Target set: *Mapillary*

Fig. 8 Fine-tuning with portions of target train set using a Deeplab V3 [10] and Fully Convolutional Network [40] (FCN), tested on the target validation set. The baseline corresponds to the model trained with the full target set

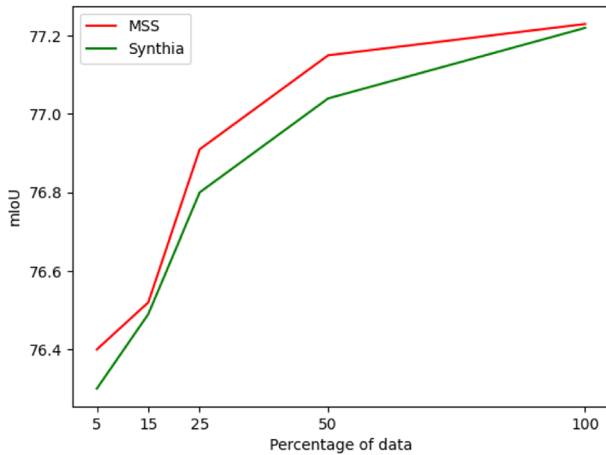


Fig. 9 Impact of synthetic dataset size for finetuning training using a DeeplabV3, tested on the target validation set *Cityscapes*

We employ all synthetic datasets in a sequential manner, periodically including new sources according to the complexity criteria defined in Section 3.2. The defined order of the datasets is (in increasing complexity, see Table 3): MSS_{50} , MSS_{100} , MSS_{250} , MSS_{750} , *Synthia* and *GTAV*.

Two hyper-parameters are used as defined in Section 4.3. γ regulates the learning rate and β regulates the amount of images which are kept from the previous training set. Table 11 presents the hyper-parameter study using a FCN architecture trained and validated with *Cityscapes* as the target dataset, from this analysis we set $\gamma = 0.9$ and $\beta = 0.75$, due to the common patterns and tendencies in performance found in previous experiments (see Sections 5.4 and 5.3), we set the same parameters for both architectures.

Tables 12 and 13 show the evolution of the results using a FCN and a Deeplab V3 respectively, we can see a mIoU increase as each new dataset is included, which affects to all classes. For comparison with the state of the art, only *Cityscapes* is included, however results extrapolate to *Mapillary*. We observe lower improvements for less representative semantic classes (e.g., *sign*, *pedestrian* and *pole*) until increasingly complex synthetic sources are added to the training set. We believe this issue to be because of the broad appearance gap between synthetic and real images. However, as more synthetic sources are added, the model is forced to look for shape similarities rather than color and texture. Hence, it produces producing big jumps in performance once a new synthetic dataset is included. For more representative semantic classes (e.g., *road*, *sidewalk*, *building* and *vegetation*), new synthetic sets reinforce performance in two ways. Regarding the first factor, note that these

Table 11 Hyper-parameter study for curriculum learning using a FCN [40] model with *Cityscapes* as the target set

γ	mIoU	β	mIoU
1	67.9	1	69.3
0.9	69.3	0.75	69.3
0.8	68.1	0.5	67.8
0.7	66.6	0.25	66.9

Table 12 Evolution in the training performance when each dataset is included in the curriculum learning scheme with a FCN

Order	Training Dataset	MIoU per class													MIoU								
		Synthetic	road	sidewalk	building	wall	fence	pole	light	sign	vegetation	terrain	sky	pedestrian		rider	car	truck	bus	train	motorcycle	bicycle	
1	MSS ₅₀	✓	45.0	11.4	45.8	0	0	0	0	0.9	44.8	0	28.9	0	0	3.0	0	0	0	0	0	0	9.6
2	MSS ₁₅₀	✓	50.3	16.7	50.5	0	0	0.2	0.2	1.2	56.5	0	30.9	0	0	13.7	0	0	0	0	0	0	11.6
3	MSS ₂₅₀	✓	51.3	20.4	50.8	0	0	0.4	0.1	1.8	65.4	0	37.3	2.7	0	32.8	0	0.3	0	0	0	0	13.9
4	MSS ₇₅₀	✓	76.9	29.7	59.9	0	0	0.3	0	2.3	66.0	0	46.7	4.7	0	45.3	0	0.3	0	0	0	0	17.5
5	Synthia	✓	89.1	46.4	70.6	1.3	0	3.8	0	14.2	73.9	7.6	65.7	17.2	0	61.31	0	2.1	0	0	0	0.2	23.9
6	GTAV	✓	91.2	55.3	74.3	2.7	0.2	20.2	0	27.7	78.0	12.1	71.9	23.9	0	66.6	0.1	0.8	0	0	0	12.6	28.3
7	Cityscapes	×	97.4	78.4	89.2	34.9	44.2	47.4	60.1	65.0	91.4	69.3	93.9	77.1	51.4	92.6	35.3	48.6	46.5	51.6	66.8	69.3	

MIoU is computed using the *Cityscapes* validation set

Table 13 Evolution in the training performance when each dataset is included in the curriculum learning scheme with a DeepLabv3

Order	Dataset	MIoU per class														MIoU						
		Synthetic	road	sidewalk	building	wall	fence	pole	light	sign	vegetation	terrain	sky	pedestrian	rider		car	truck	bus	train	motorcycle	bicycle
1	MSS ₅₀	✓	50.2	7.2	53.4	0	0	0	2.3	4.8	50.9	0	60.3	16.7	0	42.6	0	0	0	0	0	15.2
2	MSS ₁₅₀	✓	78.6	8.3	67.9	0	0	0.4	0.3	0.9	79.4	0	76.1	29.2	0	60.6	0	0	0	0	0	21.2
3	MSS ₂₅₀	✓	76.3	8.5	68.7	0	0	0.3	0.4	1.1	77.9	0	73.1	53.7	0	54.2	0	0	0	0	0	21.3
4	MSS ₇₅₀	✓	76.9	15.7	70.9	0	0	9.0	12.6	5.7	80.0	0	75.2	55.1	0	45.3	0	1.1	0	0	0	23.5
5	Synthia	✓	77.7	29.1	55.3	1.7	0	16.5	12.9	11.9	80.1	0	81.1	34.1	5.1	68.6	0	6.7	0	0.1	5.1	24.16
6	GTAV	✓	90.7	20.1	74.3	15.7	3.2	18.1	19.0	11.7	83.2	26.1	72.3	40.1	12.1	70.6	5.1	13.1	0	5.3	12.6	31.2
7	Cityscapes	✓	99.2	91.3	94.1	81.2	80.6	62.1	56.2	76.6	93.9	76.9	90.9	73.7	64.5	92.3	78.9	87.3	82.7	64.3	70.1	78.8

MIoU is computed using the Cityscapes validation set

semantic instances are heavily location biased (see Fig. 10) and that bias is common to all datasets. Therefore, including new synthetic datasets seemingly reinforces the model to rely on this location patterns rather than appearance. Regarding the second factor, note that as new semantic labels are added, less pixels are wrongly labeled as those broader classes.

Finally, experimental results validate the curriculum hypothesis, as the DNN trained using our curriculum over-perform standard-paced random training on synthetic data alone by 36.62% and 20% MIOU when employing only the *MSS* dataset and both *Synthia* and *MSS* datasets respectively, see Tables 8 and 13. This also applies for the scenario where only real data is employed and in conjunction with synthetic data for training, as the baselines are surpassed by a 6.1% and a 7.9% respectively for FCN (see Table 14) and 3.4% and 11.9% respectively for DeeplabV3 (see Tables 8, 9 and 13).

5.6 Comparison with state-of-the-art methods

Table 14 compares the proposed strategies with related works employing two widely popular architectures in semantic segmentation (Deeplab and FCN). Authors of NAE [57] performed a experiment similar to our combined training (CT) and fine-tuning (FT) without the *MSS* dataset. We can see how our CT leads to a worse performance than NAE due to having a greater amount of synthetic images, hence, decreasing real images ratio in the complete dataset. However, the inclusion of the *MSS* dataset is advantageous in the FT strategy, providing richer initial weights with a 0.4 MIOU gain in performance after fine-tuning. Our curriculum strategy (CL) achieves competitive performance to state-of-the-art baselines for both Deeplab V3 and FCN architectures, with a 2.6 gain in MIOU with the baseline, and a 0.6 gain to the state of the art [5] without relying on the training of a discriminator network, making our method a more stable and reliable approach when compared to the other alternatives.

Table 15 compares the best result of the analyzed strategies (i.e. Curriculum learning, CL) against related work in semantic segmentation. Only convolutional-based architectures are considered for this comparison to grant fairness. Results are provided for the *Cityscapes* validation set. Compared to the other models, our proposal main improvements are achieved on static classes. While [12, 13, 58, 79] present close performances to ours, our model has less than half the amount of parameters. Finally, [68] is attention-based, hence, we have employed their provided code to train a Deeplab + ResNet 101 which is publicly available on their selected framework [73].

We believe the main advantage of the proposed CL strategy is a better *learning* of the urban scenes topology. Our proposal achieves state of the art performance on semantic classes which are persistently located in similar areas of the image like: *road*, *sidewalk*, *building*, *wall* and *fence*. We attribute this improvement to the repeated training on different

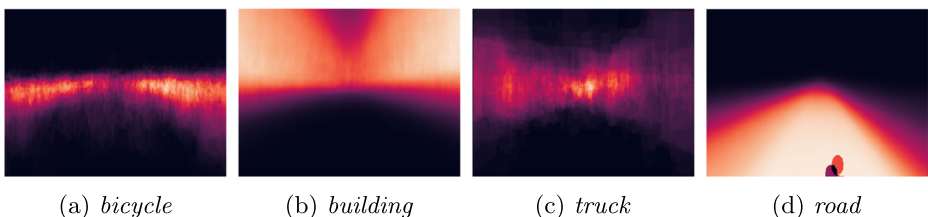


Fig. 10 Heat-map of semantic classes location probability of the *Cityscapes* dataset

Table 14 Comparison of state of the art weakly-supervised domain adaptation approaches with selected architectures on the *Cityscapes* validation set

Net	Method	Backbone	#Synthetic sets	#Real sets	Input image size	MIoU
	Proposed CT	VGG	3	1	400	64.2
	NAE [57] (CT)	VGG	2	2	512	64.6
	FCN [40]	VGG	0	3	500	65.3
	NAE [57] (FT)	VGG	2	2	512	66.0
FCN-8s	PixelDA [4, 57] ¹	VGG	2	2	512	66.1
	Focal Loss [38, 57] ¹	VGG	2	2	512	66.2
	Proposed FT	VGG	3	1	400	66.4
	NAE [57]	VGG	2	2	512	68.1
	Proposed CL	VGG	3	1	400	69.3
	Adv [29]	ResNet101	0	2	321	67.7
	RL [81]	ResNet101	2	2	512	69.2
	Proposed CT	ResNet101	3	1	400	70.4
	MME [49]	ResNet101	2	2	–	72.7
	SSDA [8]	ResNet101	1	2	–	74.7
	SSG [69]	ResNet101	2	2	512	75.2
Deeplab	Proposed baseline	ResNet101	0	2	400	76.2
	ASS [67]	ResNet101	2	2	–	77.1
	Dual level [5]	ResNet101	1	2	–	77.2
	Proposed FT	ResNet101	3	1	400	77.5
	ContrastiveSeg [68] ²	ResNet101	0	3	–	78.8
	Proposed CL	ResNet101	3	1	400	78.8

(KEY. CT: combined training, FT: Fine-tuning, CL: Curriculum learning). ¹ Results reported from [57]. ² Our results with the publicly available code. Bold indicates best results for each other class

sources of urban scenes images, hence, reinforcing through every iteration of the curriculum strategy the spatial configuration of the scene. As depicted in Fig. 10, some classes present a high location bias.

5.7 Discussion

Results One of our major findings is how much scheduling the training can impact the final performance, see Table 14. Using the same DNN, training time budget and data, we are able

Table 15 Comparison with different state of the art supervised methods for semantic segmentation on the *Cityscapes* validation set

Method	# Parameters	MIoU per class											bicycle	MIoU							
		road	sidewalk	building	wall	fence	pole	light	sign	vegetation	terrain	sky			pedestrian	rider	car	truck	bus	train	motorcycle
FCN-8s [40]	19M	97.4	78.4	89.2	34.9	44.2	47.4	60.1	65.0	91.4	69.3	93.9	77.1	51.4	92.6	35.3	48.6	46.5	51.6	66.8	65.3
Ours (FCN)	19M	96.4	86.4	94.0	64.8	59.9	69.0	56.1	79.1	93.7	57.5	90.3	71.9	59.8	92.8	55.9	59.6	44.9	46.0	68.9	69.3
RefineNet [39]	44M	98.2	83.2	91.3	47.8	50.4	56.1	66.9	71.3	92.3	70.3	94.8	80.9	63.3	94.5	64.6	76.1	64.3	62.2	69.9	73.6
DeepLabV2	44M	97.8	81.3	90.3	48.7	47.4	49.6	57.9	67.3	91.9	69.4	94.2	79.8	59.8	93.7	56.5	67.5	57.5	57.6	68.8	70.4
RecurrentParsing [31]	44M	98.5	85.4	92.5	54.4	60.9	60.2	72.3	76.8	93.1	71.6	94.8	85.2	68.9	95.7	70.1	86.5	75.5	68.3	75.5	78.2
ScaleNet [34]	44M	98.3	84.8	92.4	50.1	59.6	62.8	71.8	76.8	93.2	71.4	94.6	83.6	65.2	95.1	56.0	71.6	59.9	66.3	73.6	75.1
DLv3 [10]	61M	98.1	84.3	92.1	48.4	56.8	62.9	68.9	77.8	92.2	64.1	95.1	81.2	62.2	94.8	79.5	82.9	67.3	63.3	76.1	76.2
ESA [52]	44M	98.7	87.1	93.3	49.8	60.2	69.1	76.1	79.4	93.6	72.7	95.9	87.2	71.3	96.1	66.2	78.4	71.5	67.1	76.2	78.4
Panoptic-DeepLab [13]	68M	98.7	87.2	93.6	57.7	60.8	70.8	78.0	81.2	93.8	74.1	95.7	88.2	76.4	96.0	55.3	75.1	79.6	72.1	74.0	78.7
PSP-Net [79]	68M	98.2	85.8	92.8	57.5	65.9	62.6	71.8	80.7	92.4	64.5	94.8	82.1	61.5	95.1	78.6	88.3	77.9	68.1	78.0	78.8
DLv3+ [12]	71M	98.2	84.9	92.7	57.3	62.1	65.2	68.6	78.9	92.7	63.5	95.3	82.3	62.8	95.4	85.3	89.1	80.9	64.6	77.3	78.8
ContrastiveSeg [68]*	94M	98.3	85.8	92.9	55.2	62.9	66.7	72.0	80.1	92.7	66.4	94.6	83.5	65.8	95.5	80.4	88.5	70.9	67.2	78.7	78.8
Ours (DLv3)	61M	99.2	91.3	94.1	81.2	80.6	62.1	56.2	76.6	93.9	76.9	90.9	73.7	64.5	92.3	78.9	87.3	82.7	64.3	70.1	78.8
ProtoSeg [82]	69M	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	79.1
GSCNN [58]	137M	98.3	86.3	93.3	55.8	64.0	70.8	75.9	83.1	93.0	65.1	95.2	85.3	67.9	96.0	80.8	91.2	83.3	69.6	80.4	80.8

*Replicated results with their code on Deeplab and RN-101 backbone. Bold indicates best results for each other class

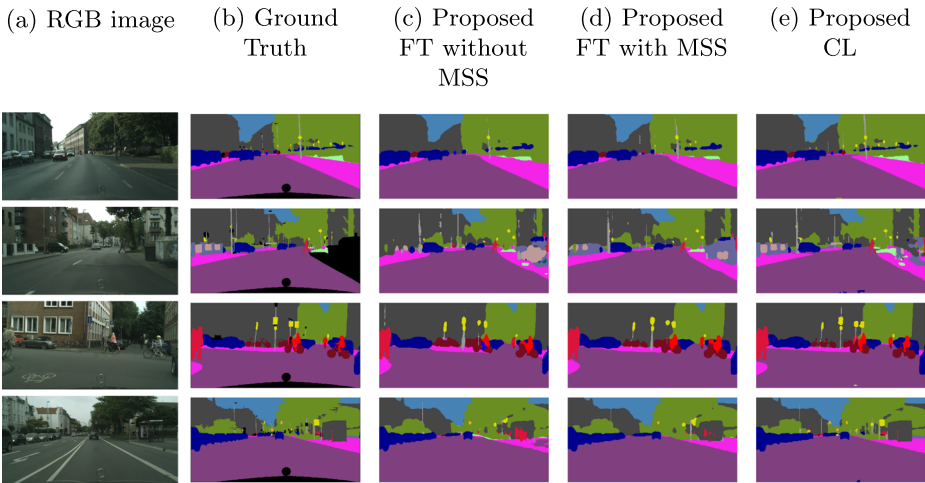


Fig. 11 Qualitative results of semantic segmentation results on the *Cityscapes* validation set. For each target image in the first column we present the GT (a). For the second to forth column we retrieve the results of the proposed finetuning without pretraining on the *MSS* dataset (b), proposed finetuning with pretraining on the *MSS* dataset (c), Proposed curriculum (d)

to improve the performance up to an 8%. Secondly, for the fine-tuning strategy, the pre-training on synthetic data leads to small fluctuations between different synthetic sets when the full real dataset is used. However, when there is little target data available, the gap in performance between synthetic datasets grows up to a 23%. Finally, following the proposed protocol for synthetic data generation (see Section 3), we managed to generate a dataset which has proven consistently useful for training compared to the most similar publicly available synthetic dataset: *Synthia* (see Tables 8, 9, 10 and Fig. 8a, b).

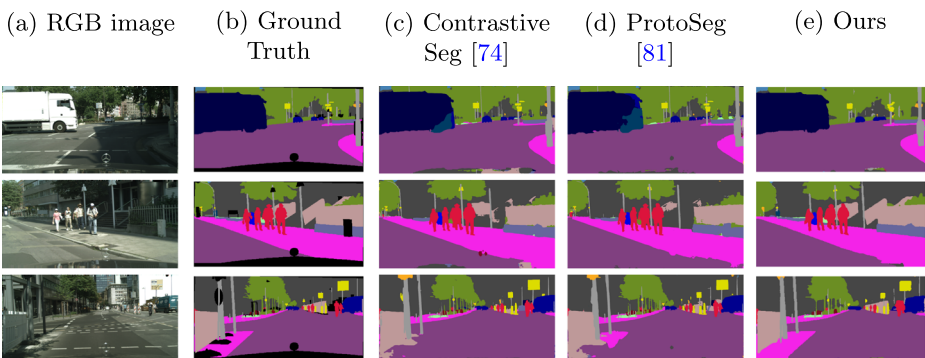


Fig. 12 Qualitative results of semantic segmentation results on the *Cityscapes* validation set against the state-of-the-art semantic segmentation proposals. For each target image in the first and second column we present the RGB image and its respective Ground truth map (a,b). For the third and fourth column we retrieve the results of two state of the art segmentation DNN, ContrastiveSeg and ProtoSeg (c,d) respectively. The last column is a DeeplabV3 model trained employing our CL proposal

Qualitative results of our model Figure 11 presents a qualitative comparison between the generated semantic segmentation of the models trained with only real data, synthetic data and a combination of both. Employing the proposed *MSS* datasets seems to be aiding the model with better discrimination of smaller structures such as signs, fences and walls. Furthermore, the proposed approach appears to provide a finer-detailed segmentation when compared with the proposed FT strategies, where one can observe a fuzzier discrimination of classes such as rider (1st row), fence and wall (2nd row), bike (3rd row) and sidewalk (4th row). Figure 12 compares different state of art alternatives ([68, 82]) with our DeeplabV3 model trained employing CL. While state of the art models provide finer details on farther and smaller structures —such as gaps between traffic signs, our model predicts more reliably structures like buses, sidewalks and fences.

6 Conclusion

In this paper our contribution is three-folded. First, we propose a new synthetic data generation protocol. By using the *MSS* simulator, we generate a new synthetic dataset for semantic segmentation which is composed of four different subsets ordered in terms of complexity, defining the complexity in terms of the amount of smaller semantic instances present in the virtual scenario. Second, we analyze the impact of introducing synthetic data using different architectures for semantic segmentation in urban scenes. We explore two different strategies to abstract synthetic data knowledge to the real domain: combined training and fine-tuning. Third, we propose a new curriculum learning strategy based on a complexity analysis of the generated data with the proposed protocol. When handling domain adaptation, we find that structuring the learning of the model leads to significant boost in performance, having combined training as the least optimal approach, sometime leading to worse models than using solely real data. Pre-training with synthetic data and fine-tuning with limited real images provides better results than training with all sources jointly. Moreover, a structured learning where images are presented in an increasing complexity manner leads to better understanding of the scene. This progressively pacing leads to a better learning of broader structures such as roads and buildings first, allowing later epochs to be focused on understanding small-sized semantic instances such as pedestrians, traffic lights and poles. This approach differs from current state of the art approaches by being model agnostic, hence, can be applied to any architecture. The results of the experiments also suggest that realism is not the only key factor of synthetic data, the content of each image and the inclusion pace of the synthetic images to the model during training are also a key factors barely analyzed in the literature.

In this work we have studied and validated the benefits of structuring and arranging data in a sample-based curriculum learning paradigm. As potential improvements of this work, we envision the incorporation of an explicit domain adaption technique to further narrow the real and synthetic domains. Moreover, we have a simple, yet effective, definition of sample complexity (i.e., number of moving object instances) for a single virtual scenario. This can be extended by incorporating additional complexity factors such as multiple view-points, number of semantic labels or even the usage of several virtual scenarios. Finally, it is important to highlight the unpaired number of classes between real and synthetic datasets. This mismatch leads to a performance degradation when evaluating using real data.

Funding Open Access funding provided thanks to the CRUE-CSIC agreement with Springer Nature. This work is part of the preliminary tasks related to the SEGA-CV (TED2021-131643A-I00) and the HVD (PID2021-125051OB-I00) projects funded by the Ministerio de Ciencia e Innovacion of the Spanish Government.

Data Availability The datasets generated during and analysed during the current study are available in the WSDA_semantic repository, http://www-vpu.eps.uam.es/publications/WSDA_semantic/

Declarations

Conflict of Interests Roberto Alcover-Couso, Juan C. SanMiguel, Marcos Escudero-Viñolo and Alvaro Garcia-Martin declare that they have no conflict of interest.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

- Balaji Y, Chellappa R, Feizi S (2019) Normalized wasserstein for mixture distributions with applications in adversarial learning and domain adaptation. In: Proc IEEE conf Comput Vis (ICCV), pp 6499–6507
- Bengio Y, Louradour J, Collobert R, Weston J (2009) Curriculum learning. In: Proceedings of the 26th annual international conference on machine learning. ICML '09. Association for computing machinery, pp 41–48
- Biasetton M, Michieli U, Agresti G, Zanuttigh P (2019) Unsupervised domain adaptation for semantic segmentation of urban scenes. In: Proc IEEE conf comput vis pattern recognit (CVPR) workshops, vol 2019-june
- Bousmalis K, Silberman N, Research G, York N, Dohan D, Erhan D, Brain G, Francisco S, Krishnan D (2019) Unsupervised pixel-level domain adaptation with generative adversarial networks. In: Proc IEEE conf Comput Vis Pattern recognit. (CVPR)
- Chen S, Jia X, He J, Shi Y, Liu J (2021) Semi-supervised domain adaptation based on dual-level domain mixing for semantic segmentation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR), pp 11018–11027
- Chen Y, Li W, Chen X, Gool LV (2019) Learning semantic segmentation from synthetic data: a geometrically guided input-output adaptation approach. Proc IEEE Conf Comput Vis Pattern Recognit (CVPR) 2019-June:1841–1850
- Chen Y-C, Lin Y-Y, Yang M-H, Huang J-B (2019) Crdoco: pixel-level domain transfer with cross-domain consistency. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)
- Chen Y, Ouyang X, Zhu K, Agam G (2021) Semi-supervised domain adaptation for semantic segmentation. arXiv:2110.10639
- Chen L-C, Papandreou G, Kokkinos I, Murphy K, Yuille AL (2015) Semantic image segmentation with deep convolutional nets and fully connected crfs. In: ICLR. arXiv:1412.7062
- Chen L-C, Papandreou G, Schroff F, Adam H (2017) Rethinking atrous convolution for semantic image segmentation. arXiv:1706.05587
- Chen M, Xue H, Cai D (2019) Domain adaptation for semantic segmentation with maximum squares loss. In: Proc IEEE conf Comput Vis (ICCV)
- Chen L-C, Zhu Y, Papandreou G, Schroff F, Adam H (2018) Encoder-decoder with atrous separable convolution for semantic image segmentation. In: Proc eur conf comput vis (ECCV)
- Cheng B, Collins MD, Zhu Y, Liu T, Huang TS, Adam H, Chen L-C (2020) Panoptic-deeplab: a simple, strong, and fast baseline for bottom-up panoptic segmentation. In: CVPR

14. Cordts M, Omran M, Ramos S, Rehfeld T, Enzweiler M, Benenson R, Franke U, Roth S, Schiele B (2016) The cityscapes dataset for semantic urban scene understanding. In: Proc IEEE conf comput vis pattern recognit (CVPR)
15. Cui Y, Jia M, Lin T-Y, Song Y, Belongie S (2019) Class-balanced loss based on effective number of samples. In: Proc IEEE conf comput vis pattern recognit (CVPR), vol 2019-June
16. Di Mauro D, Furnari A, Patanè G, Battiato S, Farinella GM (2020) Sceneadapt: scene-based domain adaptation for semantic segmentation using adversarial learning. *Pattern Recogn Lett* 136:175–182
17. Doersch C, Gupta A, Efros AA (2015) Unsupervised visual representation learning by context prediction. In: Proc IEEE conf comput vis (ICCV), pp 1422–1430
18. Dosovitskiy A, Ros G, Codevilla F, Lopez A, Koltun V (2017) CARLA: an open urban driving simulator. In: Proc 1st annual conf on robot learning, pp 1–16
19. Ganin Y, Ustinova E, Ajakan H, Germain P, Larochelle H, Laviolette F, Marchand M, Lempitsky V (2016) Domain-adversarial training of neural networks. *J Mach Learn Res* 17(1):2096–2030
20. Gatys L, Ecker A, Bethge M (2015) A neural algorithm of artistic style. arXiv:1508.06576
21. Geiger A, Lenz P, Urtasun R (2012) Are we ready for autonomous driving? the kitti vision benchmark suite. In: Proc IEEE conf comput vis pattern recognit (CVPR)
22. Georgakis G, Mousavian A, Berg AC, Košecká J (2017) Synthesizing training data for object detection in indoor scenes. arXiv:1702.07836
23. Gong C, Tao D, Maybank SJ, Liu W, Kang G, Yang J (2016) Multi-modal curriculum learning for semi-supervised image classification. *IEEE Trans Image Process* 25(7):3249–3260
24. González M (2017) Multicamera distributed system based on unity. Bachelor Thesis, Universidad Autonoma of Madrid
25. Goodfellow I, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, Ozair S, Courville A, Bengio Y (2014) Generative adversarial nets. In: Advances in neural information processing systems, pp 2672–2680
26. Gretton A, Borgwardt KM, Rasch MJ, Schölkopf B, Smola A (2012) A kernel two-sample test. *J Mach Learn Res* 13(25):723–773
27. Hinterstoisser S, Pauly O, Heibel H, Marek M, Bokoloh M (2019) An annotation saved is an annotation earned: using fully synthetic training for object instance detection. In: Proc IEEE Conf Comput Vis Pattern Recognit (CVF)
28. Hoffman J, Wang D, Yu F, Darrell T (2016) Fcns in the wild: pixel-level adversarial and constraint-based adaptation. arXiv:1612.02649. [cs.CV]
29. Hung W-C, Tsai Y-H, Liou Y-T, Lin Y-Y, Yang M-H (2018) Adversarial learning for semi-supervised semantic segmentation. In: Proceedings of the british machine vision conference (BMVC)
30. Ionescu RT, Alexe B, Leordeanu M, Popescu M, Papadopoulos DP, Ferrari V (2016) How hard can it be? estimating the difficulty of visual search in an image. In: 2016 IEEE conference on computer vision and pattern recognition (CVPR), pp 2157–2166
31. Kong S, Fowlkes CC (2017) Recurrent scene parsing with perspective understanding in the loop. CoRR arXiv:1705.07238
32. Kumar MP, Packer B, Koller D (2010) Self-paced learning for latent variable models. In: NIPS
33. Kumar MP, Turki H, Preston D, Koller D (2011) Learning specific-class segmentation from diverse data. In: 2011 International conference on computer vision, pp 1800–1807
34. Li Y, Kuang Z, Chen Y, Zhang W (2019) Data-driven neuron allocation for scale aggregation networks. In: Proc IEEE conf comput vis pattern recognit CVPR
35. Li S, Liu CH, Lin Q, Xie B, Ding Z, Huang G, Tang J (2020) Domain conditioned adaptation network. In: Proc conf art intell (AAAI), pp 11386–11393
36. Li X, Zhou T, Li J, Zhou Y, Zhang Z (2020) Group-wise semantic mining for weakly supervised semantic segmentation. arXiv
37. Lian Q, Lv F, Duan L, Gong B (2019) Constructing self-motivated pyramid curriculums for cross-domain semantic segmentation: a non-adversarial approach. In: Proc IEEE conf comput vis (ICCV)
38. Lin T-Y, Goyal P, Girshick RB, He K, Dollár P (2017) Focal loss for dense object detection. In: Proc IEEE conf Comput Vis (ICCV), pp 2999–3007
39. Lin G, Milan A, Shen C, Reid ID (2016) Refinenet: Multi-path refinement networks for high-resolution semantic segmentation. CoRR arXiv:1611.06612
40. Long J, Evan Shelhamer* TD (2015) Fully convolutional models for semantic segmentation. In: Proc IEEE conf comput vis pattern recognit (CVPR)
41. Michieli U, Zanuttigh P (2021) Knowledge distillation for incremental learning in semantic segmentation. *Comput Vis Image Underst* 205:103167. <https://doi.org/10.1016/j.cviu.2021.103167>
42. Neuhold G, Ollmann T, Rota Bulò S, Kotschieder P (2017) The mapillary vistas dataset for semantic understanding of street scenes. In: Proc IEEE conf comput vis (ICCV)

43. Nowruzi FE, Kapoor P, Kolhatkar D, Hassanat FA, Laganieri R, Rebut J (2019) How much real data do we actually need: Analyzing object detection performance using synthetic and real data. arXiv:1907.07061
44. Pemasiri A, Nguyen K, Sridharan S, Fookes C (2021) Multi-modal semantic image segmentation. *Comput Vis Image Underst* 202:103085
45. Pentina A, Sharmanska V, Lampert CH (2015) Curriculum learning of multiple tasks. In: 2015 IEEE conference on computer vision and pattern recognition (CVPR), pp 5492–5500
46. Richter SR, Vineet V, Roth S, Koltun V (2016) Playing for data: ground truth from computer games. In: *Proc eur conf comput vis (ECCV)*, pp 102–118. Springer, Cham
47. Ros G, Sellart L, Materzynska J, Vazquez D, Lopez AM (2016) The synthia dataset: a large collection of synthetic images for semantic segmentation of urban scenes. *Proc IEEE Conf Comput Vis Pattern Recognit (CVPR)* 2016:3234–3243
48. Russo P, Tommasi T, Caputo B (2019) Towards multi-source adaptive semantic segmentation. In: *Image analysis and processing – ICIAP 2019: 20th international conference, Trento, Italy, 9–13 Sept 2019, proceedings, Part I*. Springer, pp 292–301
49. Saito K, Kim D, Sclaroff S, Darrell T, Saenko K (2019) Semi-supervised domain adaptation via minimax entropy. In: *Proc IEEE Conf Comput Vis (ICCV)*
50. Sankaranarayanan S, Balaji Y, Jain A, Lim SN, Chellappa R (2018) Learning from synthetic data : addressing domain shift for semantic segmentation. *Proc IEEE Conf Comput Vis Pattern Recognit (CVPR)* 2018-June:3–5
51. Saporta A, Vu T-H, Cord M, Pérez P (2020) ESL: entropy-guided self-supervised learning for domain adaptation in semantic segmentation
52. Seichter D, Köhler M, Lewandowski B, Wengefeld T, Gross H-M (2021) Efficient rgb-d semantic segmentation for indoor scene analysis. In: *IEEE international conference on robotics and automation (ICRA)*, pp 13525–13531
53. Shoebi A, Khodatars M, Alizadehsani R, Ghassemi N, Jafari M, Moridian P, Khadem A, Sadeghi D, Hussain S, Zare A, Sani ZA, Bazeli J, Khozeimeh F, Khosravi A, Nahavandi S, Acharya UR, Shi P (2020) Automated detection and forecasting of COVID-19 using deep learning techniques: a review. *CoRR arXiv:2007.10785*
54. Shoebi A, Khodatars M, Jafari M, Moridian P, Rezaei M, Alizadehsani R, Khozeimeh F, Gorriz JM, Heras J, Panahiazar M, Nahavandi S, Acharya UR (2021) Applications of deep learning techniques for automated multiple sclerosis detection using magnetic resonance imaging: a review. *Comput Bio Med* 136:104697
55. Soviany P, Ionescu RT, Rota P, Sebe N (2021) Curriculum learning: a survey. *CoRR arXiv:2101.10382*
56. Sun B, Feng J, Saenko K (2016) Return of frustratingly easy domain adaptation. In: *AAAI*
57. Sun R, Zhu X, Wu C, Huang C, Shi J, Ma L (2019) Not all areas are equal: transfer learning for semantic segmentation via hierarchical region selection. In: *Proc IEEE conf comput vis pattern recognit (CVPR)*
58. Takikawa T, Acuna D, Jampani V, Fidler S (2019) Gated-scnn: gated shape cnns for semantic segmentation. *ICCV*
59. Tobin J, Fong R, Ray A, Schneider J, Zaremba W, Abbeel P (2017) Domain randomization for transferring deep neural networks from simulation to the real world. *Proc IEEE/RSJ Conf Intell Rob Sys (IROS)*:23–30
60. Toldo M, Michieli U, Zanuttigh P (2021) Unsupervised domain adaptation in semantic segmentation via orthogonal and clustered embeddings. In: *Proc IEEE conf appl comp vis (WACV)*
61. Tremblay J, Prakash A, Acuna D, Brophy M, Jampani V, Anil C, To T, Cameracci E, Boochoon S, Birchfield S (2018) Training deep networks with synthetic data: bridging the reality gap by domain randomization. *Proc IEEE Conf Comput Vis Pattern Recognit (CVPR) Workshops 2018-June:1082–1090*
62. Tsai Y-H, Hung W-C, Schuster S, Sohn K, Yang M-H, Chandraker M (2018) Learning to adapt structured output space for semantic segmentation. In: *Proc IEEE conf comput vis pattern recognit (CVF)*, pp 7472–7481
63. Vu T-H, Jain H, Bucher M, Cord M, Pérez P (2019) Advent: adversarial entropy minimization for domain adaptation in semantic segmentation. In: *Proc IEEE conf comput vis pattern recognit (CVPR)*
64. Wang Q, Gao J, Li X (2019) Weakly supervised adversarial domain adaptation for semantic segmentation in urban scenes. *Proc IEEE Trans Image Process* 28:4376–4386
65. Wang Y, Mo L, Ma H, Yuan J (2020) Occgan: semantic image augmentation for driving scenes. *Pattern Recogn Lett* 136:257–263
66. Wang H, Shen T, Zhang W, Duan L, Mei T (2020) Classes matter: a fine-grained adversarial approach to cross-domain semantic segmentation. In: *Proc eur conf comput vis (ECCV)*

67. Wang Z, Wei Y, Feris R, Xiong J, Hwu W-M, Huang TS, Shi H (2020) Alleviating semantic-level shift: a semi-supervised domain adaptation method for semantic segmentation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR) workshops
68. Wang W, Zhou T, Yu F, Dai J, Konukoglu E, Van Gool L (2021) Exploring cross-image pixel contrast for semantic segmentation. arXiv
69. Wen S, Tian W, Zhang H, Fan S, Zhou N, Li X (2020) Semantic segmentation using a gan and a weakly supervised method based on deep transfer learning. *IEEE Access* 8:176480–176494
70. Wu Y, Liu L, Bae J, Chow K-H, Iyengar A, Pu C, Wei W, Yu L, Zhang Q (2019) Demystifying learning rate policies for high accuracy training of deep neural networks. In: 2019 IEEE international conference on big data (big data), pp 1971–1980
71. Yang L, Balaji Y, Lim S-N, Shrivastava A (2020) Curriculum manager for source selection in multi-source domain adaptation. In: European conference on computer vision. Springer, pp 608–624
72. Yao R, Lin G, Xia S, Zhao J, Zhou Y (2020) Video object segmentation and tracking: a survey. *ACM Trans Intell Syst Technol*, vol 11(4)
73. Yuan Y, Fu R, Huang L, Lin W, Zhang C, Chen X, Wang J (2021) Hrt: high-resolution transformer for dense prediction
74. Zhang Y, David P, Foroosh H, Gong B (2020) A curriculum domain adaptation approach to the semantic segmentation of urban scenes. *Proc IEEE Trans Pattern Anal Mach Intell* 42:1823–1841
75. Zhang Y, David P, Gong B (2017) Curriculum domain adaptation for semantic segmentation of urban scenes. *Proc IEEE Conf Comput Vis (ICCV)*:2039–2049
76. Zhang B, Zhao S, Zhang R (2021) Cross-domain semantic segmentation of urban scenes via multi-level feature alignment. In: 2020 25th International conference on pattern recognition (ICPR), pp 1912–1917
77. Zhao S, Li B, Reed C, Xu P, Keutzer K (2020) Multi-source domain adaptation in the deep learning era: a systematic survey. *CoRR arXiv:2002.12169*
78. Zhao S, Li B, Yue X, Gu Y, Xu P, Hu R, Chai H, Keutzer K (2019) Multi-source domain adaptation for semantic segmentation. In: *Advances in neural information processing systems*, vol 32
79. Zhao H, Shi J, Qi X, Wang X, Jia J (2017) Pyramid scene parsing network. arXiv:1612.01105. [cs.CV]
80. Zhao S, Xiao Y, Guo J, Yue X, Yang J, Krishna R, Xu P, Keutzer K (2021) Curriculum cyclegan for textual sentiment domain adaptation with multiple sources. In: Proceedings of the web conference 2021, pp 541–552
81. Zheng Q, Chen J, Huang P, Hu R (2019) Urban scene semantic segmentation with insufficient labeled data. *China Commun* 16(11):212–221
82. Zhou T, Wang W, Konukoglu E, Van Gool L (2022) Rethinking semantic segmentation: a prototype view. In: CVPR
83. Zhou T, Zhang M, Zhao F, Li J (2022) Regional semantic contrast and aggregation for weakly supervised semantic segmentation. arXiv

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.