

# Exploiting Semantics in Adversarial Training for Image-Level Domain Adaptation

1<sup>st</sup> Pierluigi Zama Ramirez  
University of Bologna  
pierluigi.zama@unibo.it

2<sup>nd</sup> Alessio Tonioni  
University of Bologna  
alessio.tonioni@unibo.it

3<sup>rd</sup> Luigi Di Stefano  
University of Bologna  
luigi.distefano@unibo.it

**Abstract**—Performance achievable by modern deep learning approaches are directly related to the amount of data used at training time. Unfortunately, the annotation process is notoriously tedious and expensive, especially for pixel-wise tasks like semantic segmentation. Recent works have proposed to rely on synthetically generated imagery to ease the training set creation. However, models trained on these kind of data usually under-perform on real images due to the well known issue of domain shift. We address this problem by learning a domain-to-domain image translation GAN to shrink the gap between real and synthetic images. Peculiarly to our method, we introduce semantic constraints into the generation process to both avoid artifacts and guide the synthesis. To prove the effectiveness of our proposal, we show how a semantic segmentation CNN trained on images from the synthetic GTA dataset adapted by our method can improve performance by more than 16% mIoU with respect to the same model trained on synthetic images.

**Index Terms**—domain adaptation, semantic segmentation, GAN

## I. INTRODUCTION

Recent advancements in computer vision are characterized by a widespread adoption of deep learning, either as end-to-end complete solutions or as components of more complex pipelines. A common trait across all the different flavours of deep learning models is the strong correlation between the size of the accurately annotated training set and the achievable performance. As for research work, the need of a large corpus of annotated data may not be an issue thanks to availability of many curated dataset. Yet, the data issue is limiting a more widespread adoption of deep learning in many practical applications. Even assuming availability of training images for the target environment, manually producing the annotations is a tedious and expensive operation, that quickly become hard to scale for more complex tasks. For example, annotating a single image with a global label usually requires few seconds while annotating the same image for pixel-wise prediction tasks, such as depth estimation or semantic segmentation, requires many minutes or hours, even with the help of professional tools [1].

Many recent works [2]–[5] have proposed to deploy synthetic training images generated by state-of-the-art computer graphics techniques to obtain for free, during the rendering process, different kinds of annotations. Yet, such synthetic training samples turn out significantly different from the real images processed at test time, which implies a well-known issue, referred to in the machine learning literature as *domain*

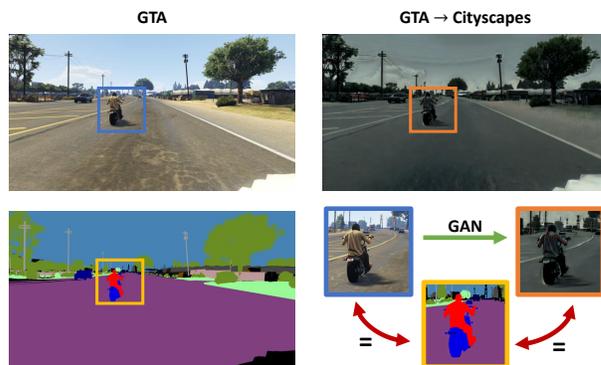


Fig. 1: On the right an image generated applying our semantically aware GAN on a synthetic image from the GTA dataset [2] (left) to make the latter look more realistic. Lower right corner: zoomed crops to highlight how our semantically aware GAN can transform images across domains preserving the semantic structure of the scene.

*shift*. As a consequence, models trained only on synthetic data severely under-perform in the real deployment scenario. Therefore, the commonly used training protocol mandates the use of (potentially few) annotated training samples from the target domain to perform fine-tuning and recover good performance. The assumption of having annotated images from the target domain at hand, unfortunately, can not always be fulfilled for complex tasks like 3D reconstruction, semantic segmentation or pose estimation where data acquisition and, especially, labeling is often a challenging and costly task per se. Promising works like [6]–[8] try to learn models which extract the same kind of features across the two domain. While this strategy seems successful for tasks like classification, it is still unclear how well it can scale to dense *structured domain adaptation* [9] where the improvement gained by feature alignment is still modest. Alternatively, [10], [11] work directly on the training data trying to shrink the gap between synthetic and real images by transforming the first to make them look real using image-to-image generative adversarial networks. However, since they do not enforce any kind of constraint on the geometric consistency between input and output, these approaches can easily produce artifacts and distortions. Beside harming the realism of the generated images, artifacts could easily render annotations created for the synthetic images

useless, especially for pixel-level labeling task where even a few pixels shift can invalidate the annotation.

Building on these observations we propose a novel approach based on image-to-image domain translation by GANs while explicitly training the system to keep the semantic structure of the scene. The intuition behind our formulation is that forcing the *generator* network to keep the semantic structure of the image act as a regularizer enforcing overall consistency of image appearance and producing images that look more realistic and exhibit less artifacts. For example, according to our formulation a "tree" can change its appearance but it should still be recognizable as a "tree" across domains. To enforce the semantic constraint we train a *discriminator* network not only to classify the domain (real/fake) but also to solve the task of semantic segmentation on the synthetic domain (*i.e.*, we do not need labels on the real target domain). Moreover, we introduce an appearance reconstruction loss to further regularize the generation process and help preserving small details across the domain adaptation. To asses the effectiveness of our proposal we transform synthetic images obtained from the synthetic GTA datasets [2] to look similar to the real images of the Cityscapes [1] dataset. Fig. 1 shows on the right column a qualitative example generated by our method using as input the corresponding synthetic images depicted on the left. We will show how those images can be used to train a model to solve the problem of semantic segmentation yielding promising result with respect to using synthetic images.

## II. RELATED WORKS

The main topics relevant to this article are Semantic Segmentation and Domain Adaptation.

### A. Semantic Segmentation

Since the advent of deep learning, semantic segmentation is mainly performed by convolutional neural networks [8]. Several kind of architectures are employed. Multi-scale models, such as [12]–[15], take inputs at different resolution to extract context information at different abstraction levels. Encoder-decoder networks, [16]–[18], combines an encoder to extract high-level, low-resolution features which are later exploited by a decoder to reconstruct an high-resolution semantic map. Other networks employ Conditional Random Fields, [14], [19], to encode long range context information. Modern networks deploy spatial pyramid pooling and atrous-convolutions to extract information at different level of abstraction [14], [20], [21].

### B. Domain Adaptation

Many domain adaptation techniques have been proposed to address the domain-shift problem [22]–[24]. Earliest approaches such as [25], [26] try to build intermediate representations using manifolds while recent ones, tailored for deep learning, focus on adversarial training. Deep domain adaptation can be mainly divided in two branches: Pixel-level and Feature-Level. Feature-level approaches, such as

[6]–[8], [27] seek to find a domain invariant representation, obtaining networks able to perform well on both the source and target domains. On the other hand, pixel-level approaches, such as [10], work on image data and try to directly convert the source image into a target style image relying on recent image-to-image translation generative networks [28], [29]. Few works have explicitly studied domain adaptation for semantic segmentation. [30] performs two kind of alignments: a global alignment through adversarial back-propagation, as in [31], and a local one, which aligns class specific statistics by a multiple instance learning formulation. [32] proposes curriculum-style learning where a teacher network solve the easier task of learning global label distributions over images and local distributions over landmark superpixels, then a student segmentation network is trained so that the target label distribution follow these inferred label properties. Similarly to our proposal, [33] combines the cycle consistency loss proposed by [29] with a semantic consistency loss. While they combine different networks trained sequentially, in our work we proposed a simpler end-to-end architecture with a semantic discriminator that obtains comparable or even better results.

## III. PROPOSED METHOD

In this section we present our proposal for domain adaptation exploiting semantic information. We consider the problem of unsupervised and unpaired pixel-level domain adaptation from a source to a target domain. We define as  $X_s$ ,  $Y_s$  the provided source data and associated semantic labels whilst as  $X_t$  the provided target data, but without any available target labels. Our goal is to transform source images so to resemble target images while maintaining the semantic content of the scene during the generation process. A schematic representation of our method is shown in Fig. 2.

### A. Architecture

Inspired by [29], we adopt a cycle architecture consisting of two generators and two semantic discriminators. The first generator,  $G_{S \rightarrow T}$ , introduce a mapping from source to target domain and produces target samples which should deceive the discriminator  $D_T$ . The discriminator  $D_T$ , instead, learns to distinguish between adapted source and true target samples. On the other hand, the second generator,  $G_{T \rightarrow S}$ , learns the opposite mapping from source to target data, while the second discriminator  $D_s$  distinguish between adapted target and true source samples. Furthermore, peculiarly to our work, both *semantic* discriminators act not only as classifiers but also as semantic segmentation networks. Thus, we add a second decoder to  $D_T$  and  $D_S$  obtaining  $D_{S_{sem}}$  and  $D_{T_{sem}}$ . The features extracted by the last encoder layer of the discriminators are used to generate both the semantic map and the domain classification score.

### B. Training

We train our system to minimize multiple losses:

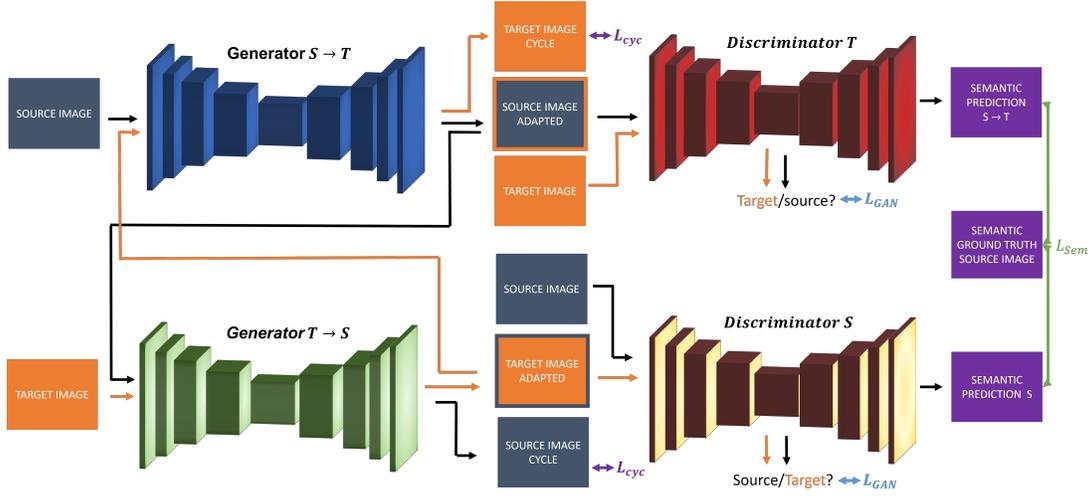


Fig. 2: Schematic representations of the proposed network architecture. In dark blue and orange images from the source domain and target domain respectively. Dual color framed images are obtained by our adaptation method. In purple the semantic maps.

### Adversarial Loss

We apply adversarial losses [34] to both mapping functions  $S \rightarrow T$  and  $T \rightarrow S$ . For the sake of space, we define here only source to target adversarial loss, being equivalent to its inverse.

$$\mathcal{L}_{adv} = \mathbb{E}_{x_t \sim X_T} [\log(D_t(x_t))] + \mathbb{E}_{x_s \sim X_s} [\log(1 - D_t(G_{s \rightarrow t}(x_s)))] \quad (1)$$

$G_{S \rightarrow T}$  tries to generate images that look similar to images from domain  $T$  while  $D_T$  tries to distinguish between adapted source samples  $G_{S \rightarrow T}(x_s)$  and real target samples  $X_T$ .  $G_{S \rightarrow T}$  seek to minimize this objective against  $D_T$  which instead tries to maximize it.

### Semantic Discriminator Loss

We train both discriminators,  $D_{S_{sem}}$  and  $D_{T_{sem}}$ , to perform semantic segmentation employing source labels.  $D_{T_{sem}}$  will be trained on adapted source images, while  $D_{S_{sem}}$  will be trained directly on source images. We used a pixel-wise cross entropy loss  $H(p, q)$  as in standard segmentation networks:

$$\mathcal{L}_{sem} = H(D_{S_{sem}}(G_{S \rightarrow T}(X_S)), Y_S) + H(D_{S_{sem}}(X_S), Y_S) \quad (2)$$

### Weighted Reconstruction Loss

We exploit the cyclic L1 reconstruction loss proposed in [29] for target samples where we do not have any label. Regarding source samples, we weight each pixel proportionally to the probability of not belonging to its semantic class. Our weighting term acts as a regularization where the network usually fail adaptation introducing artifacts, forcing the least frequent classes to be reconstructed preserving input appearance:

$$\mathcal{L}_{rec} = \|G_{S \rightarrow T}(G_{T \rightarrow S}(x_T)) - x_T\|_1 + (1 - w) \|G_{T \rightarrow S}(G_{S \rightarrow T}(x_S)) - x_S\|_1 \quad (3)$$

$w$  is a weight mask with the same resolution of the source image. Defined  $C$  as the set of possible classes, each weight  $w_{i,j}$  represents the likelihood of a class among all synthetic dataset:

$$w_{i,j} = \frac{n_{pixel \in c}}{n_{pixel}}, c \in C \quad (4)$$

### Final Loss

We train our discriminators and generators to minimize the following losses:

$$\mathcal{L}_D = -\mathcal{L}_{adv} + \lambda_{sem} \mathcal{L}_{sem} \quad (5)$$

$$\mathcal{L}_G = \mathcal{L}_{adv} + \lambda_{sem} \mathcal{L}_{sem} + \lambda_{rec} \mathcal{L}_{rec}$$

$\lambda_{sem}$  and  $\lambda_{rec}$  are hyper-parameters that control the relative importance of domain classification, weighted reconstruction and semantic segmentation. Across all our experiments we will use  $\lambda_{sem} = 1$  and  $\lambda_{rec} = 3$ .

## IV. EXPERIMENTAL RESULTS

We conduct a series of tests to assess the effectiveness of our method in producing realistic images and verify if they are suitable for training deep learning models.

### A. Datasets Creation

We have used as synthetic source domain the GTA dataset [2], that features 22K realistic synthetic images obtained from the Grand Theft Auto videogame enriched with perfect pixel level annotations for semantic segmentation. As target real images we have used the Cityscapes dataset [1] featuring 5000 images acquired during real driving sessions around Germany and annotated with precise pixel level labels for semantic segmentation. Among all available images we have used the *training* split as our target samples during training, while we have kept the *validation* split to measure performance of different semantic segmentation networks. We did not use the



Fig. 3: Image generated by CycleGAN [29](b) and our semantics-aware GAN (c) for the GTA to Cityscapes domain alignment task

*test* split since the labels are not publicly available. We chose these two datasets as they provide annotations for the same set of semantic classes and feature domains where the biggest difference concern the shift from synthetic to real images. We used ResNet as our generator networks and U-Net [18] as our discriminator. Using the loss formulation described in Sec. III-B we have trained our GAN to transform images from the GTA [2] to the Cityscapes [1] domain for 300k steps using Adam as optimizer, 0.0001 for learning rate and batch size 2. We cropped our input images to 512x512. During the training process we have used images and labels from GTA and only images from Cityscapes, *i.e.*, our method does not require annotations from the real/target domain but only from the source one. Once trained, we used the generator to transform synthetic images from the training dataset to produce an *aligned* GTA dataset that should resemble images from the real Cityscapes domain. On Fig. 3 we depict some qualitative example of images produced by our GAN (column (c)) together with the corresponding input from the GTA dataset (column (a)) and some exemplar images from the Cityscapes, target, dataset (column (d)). To better show the effectiveness of our semantic aware GAN, we also report images obtained by training a CycleGAN network [29] that does not use any semantic clues at training time (column (b)). By comparing our images (column (c)) with those produced by CycleGAN (column (b)) it turns out clearly that, unlike previous approaches (*i.e.*, column (b)), our novel formulation can preserve the semantic content and avoid introduction of artifacts. Moreover, the introduction of semantic constraints during the training process helps to produce sharper edges in the final image, which increases the quality of the images compared to CycleGAN. We have also applied our GANs to entire video sequences from the GTA domain and verified that the network can easily maintain temporal consistency even if it has only been trained on single frames.<sup>1</sup>

### B. Semantic Segmentation

Fig. 3 shows how our network can produce visually appealing images. In the following we demonstrate that our

adapted images can be used to train a neural network to obtain much better performance on the target domain w.r.t the corresponding synthetic ones.

Focusing on semantic segmentation, we have trained a standard FCN8s [16] on the original GTA synthetic images and on our *aligned* dataset. We tested both on the validation set of Cityscapes and reported the result in Table I. For all our tests, we have initialized the feature extractor of the FCN8s with the publicly available VGG16 weights trained on the Imagenet dataset, then performed 100000 training iterations using batch 4, Adam optimizer and 0.0001 as learning rate. We trained networks on 1024x1024 cropped images.

To compare the networks we report two different metrics: the mean instance-level intersection-over-union (from now on shortened *mIoU*) computed following the guidelines of the Cityscapes benchmark [1] and the overall pixel accuracy (shortened *acc*), *i.e.*, the percentage of correctly predicted pixel labels. We also report detailed scores for each semantic class to highlight for which categories our image augmentation schema is more effective. We compare the results obtained by our domain adaptation method with alternatives recently proposed in literature: the feature-level alignment method of [30], the curriculum style domain adaptation approach of [32] and the pixel level alignment introduced in [33]. In Table I for all methods we report the performance achieved by training the very same FCN8s network [16] both before and after domain alignment, the former marked using *Source* in the method column. For each row we report per class *mIoU* and aggregated performance across the whole dataset (last two columns). Concerning aggregated *mIoU* score, we can see how our proposal can outperform both [30], [32] while being comparable with [33]. Moreover, considering pixel accuracy, our proposal compares favourably even to [33]. Considering the performance achieved before and after domain adaptation, our proposed pixel level alignment can provide an impressive +16.9 gain in *mIoU* and a +24 in *Acc.*, that, once again, compares favourably to [30], [32] and is comparable to [33]. Looking at class scores, we observe how our proposal can achieve the best absolute performances on 10 classes out of 19, including some key ones for autonomous driving like *road*

<sup>1</sup><https://youtu.be/wIpFcKLviYQ>

Method	road	sidewalk	building	wall	fence	pole	traffic light	traffic sign	vegetation	terrain	sky	person	rider	car	truck	bus	train	motorcycle	bicycle	mIoU	Acc.
Source [30]	31.9	18.9	47.7	7.40	3.10	16.0	10.4	1.00	76.5	13.0	58.9	36.0	1.00	67.1	9.50	3.70	0.00	0.00	0.00	21.2	-
[30]	70.4	32.4	62.1	14.9	5.40	10.9	14.2	2.70	79.2	21.3	64.6	44.1	4.20	70.4	8.00	7.30	0.00	3.50	0.00	27.1	-
Source [32]	18.1	6.80	64.1	7.30	8.70	21.0	14.9	16.8	45.9	2.40	64.4	41.6	17.5	55.3	8.40	5.0	6.90	4.30	13.8	22.3	-
[32]	74.9	22.0	71.7	6.00	11.9	8.40	16.3	11.1	75.7	13.3	66.5	38.0	9.30	55.2	<b>18.8</b>	18.9	0.00	<b>16.8</b>	<b>16.6</b>	28.9	-
Source [33]	26.0	14.9	65.1	5.50	12.9	8.90	6.00	2.50	70.0	2.90	47.0	24.5	0.0	40.0	12.1	1.50	0.0	0.0	0.0	17.9	54.0
[33]	83.5	<b>38.3</b>	76.4	20.6	<b>16.5</b>	22.2	<b>26.2</b>	<b>21.9</b>	80.4	<b>28.7</b>	65.7	49.4	4.2	74.6	16.0	<b>26.6</b>	2.0	8.0	0.0	<b>34.8</b>	82.8
Source Ours	43.3	11.9	54.3	3.42	11.96	9.63	10.74	5.23	68.3	6.39	46.84	30.02	2.07	33.1	7.72	0.00	0.00	0.00	0.00	18.2	60.4
Ours	<b>85.4</b>	<b>32.8</b>	<b>78.0</b>	<b>21.0</b>	9.35	<b>26.1</b>	18.0	8.71	<b>82.2</b>	22.1	<b>71.2</b>	<b>51.4</b>	<b>13.4</b>	<b>79.5</b>	16.0	13.5	<b>7.83</b>	10.1	0.03	34.2	<b>84.4</b>

TABLE I: Comparison between domain adaptation methods for semantic segmentation on the Cityscapes validation set. Middle section reports mIoU score per class, final two columns aggregated performance across the whole dataset, best results highlighted in bold.

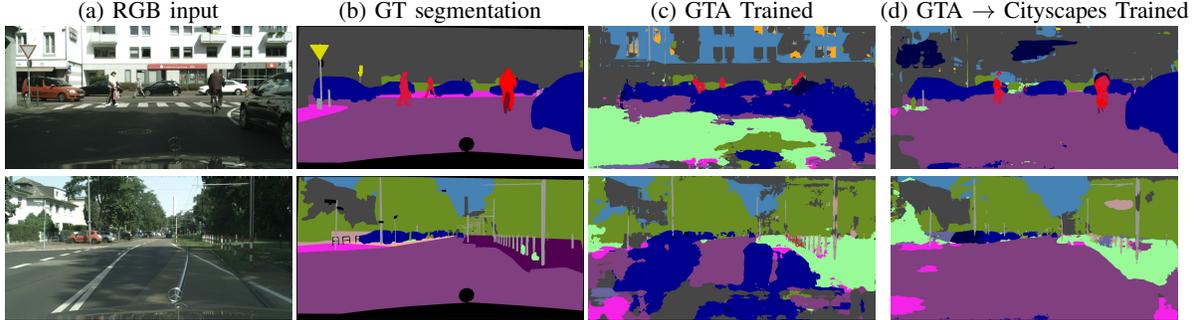


Fig. 4: Segmentation results on the Cityscapes dataset for a FCN8s network trained only on synthetic data from the GTA dataset (c) and on our GTA *adapted* dataset (d).

(+42.1 gain between before and after alignment), car (+46.4) and *person* (+21.4). We still lose something compared to other proposals on less common classes (e.g., *bus*, *motorcycle* and *bicycle*), we think that this might be due to the dataset used not having enough samples of the target classes to effectively teach to the generator how to realistically render them. Even though our proposal performs comparably to [33], we would like to stress out how our adaptation method can be trained end-to-end instead of relying on separate training steps for the different parts.

In Fig. 4 we also report some qualitative examples of the improvement in segmentation attainable by training on our *adapted* GTA images (column (d)) compared to a purely synthetic training set (column (c)). Even if the results in column (d) are still far from optimal, most of the mistakes visible in column (c) are completely gone and the overall structure of the scene is more accurately segmented. Moreover, we can notice visually how the larger improvement concerns the segmentation of *road* (colored purple), *cars* (colored blue) and *people* (colored red).

### C. Ablation Study

In Sec. IV-B we have proven that the images generated by our proposal can effectively be used to train a semantic segmentation network so as to nearly double its performance compared to using synthetic data only. In this section, instead, we investigate more in depth on how each component of our proposal contributes to the final result. Purposely, we trained different architectures, keeping the comparison as fair

Test	mIoU	Acc.
(a) Synthetic	18.23	60.43
(b) GAN+Sem.	29.45	78.13
(c) GAN+Sem+weight.	31.33	79.85
(d) Cycle [29]	29.43	79.20
(e) Cycle+sem+weight.	<b>34.27</b>	<b>84.48</b>

TABLE II: Ablation study on the different component of our semantic aware GAN. Best results in bold.

as possible by maintaining the same training protocol. We report the results of these tests in Table II. We first investigated the performance of training a semantic segmentation network on images adapted by a simple GAN [34]. As we obtained results even worse than our baseline network (a), we decided to not report them in Table II. We then trained a GAN framework enriched with our semantic discriminator. Comparing line (b) with (a) we can clearly see how adding our semantics-aware discriminator not only allows to successfully train the GAN system but also results in a +11.22% *mIoU*, thus testifying how semantic information can successfully regularize training. We then added our weighted L1 reconstruction loss between source and adapted image (c) slightly improving performances by a +1.88% *mIoU*. We then trained a standard CycleGAN [29] with no semantic clue demonstrating how having two couples of generator and discriminator is extremely effective to stabilize training of a GAN framework, as shown by (d) reaching comparable results to (b). Finally (e) reports the performance achievable by our full proposal that deploys

the CycleGAN network combined with the semantics-aware discriminator and our semantic weighting system, achieving remarkable performance: +16.04% *mIoU* and +24.05% *Acc.* with respect to our baseline(a).

## V. CONCLUSION AND FUTURE WORKS

In this paper we have demonstrated how a semantically aware image-to-image translation network can be successfully deployed to shrink the gap between images belonging to two drastically different domains such as synthetic and real images. Our novel network structure and loss function can successfully produce realistic images, thanks to its adversarial component, while at the same time maintaining structural coherence between input and output thanks to the enforced semantic consistency.

We have addressed the adaptation problem only at *pixel level*, however recent works [33] have shown how for the semantic segmentation task the best absolute performance can be achieved by a mix of *pixel level* and *feature level* alignment. Therefore we plan to add an additional fine tuning step of our semantic segmentation network in order to introduce feature alignment in our pipeline. Moreover, we have tested our proposal for domain adaptation from synthetic to real images in the context of image segmentation, however, the same process can be used to address different tasks, *e.g.* object detection, or different type of domain shifts, *e.g.* different seasons, different sensors or different weather conditions. We plan to carry out these tests in order to achieve a more comprehensive experimental evaluation of our proposal.

## REFERENCES

- [1] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The cityscapes dataset for semantic urban scene understanding," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.
- [2] S. R. Richter, V. Vineet, S. Roth, and V. Koltun, "Playing for data: Ground truth from computer games," in *European Conference on Computer Vision*. Springer, 2016, pp. 102–118.
- [3] S. R. Richter, Z. Hayder, and V. Koltun, "Playing for benchmarks," in *International Conference on Computer Vision (ICCV)*, 2017.
- [4] G. Ros, L. Sellart, J. Materzynska, D. Vazquez, and A. M. Lopez, "The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 3234–3243.
- [5] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun, "CARLA: An open urban driving simulator," in *Proceedings of the 1st Annual Conference on Robot Learning*, 2017, pp. 1–16.
- [6] E. Tzeng, J. Hoffman, K. Saenko, and T. Darrell, "Adversarial discriminative domain adaptation," in *Computer Vision and Pattern Recognition (CVPR)*, vol. 1, no. 2, 2017, p. 4.
- [7] Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. Marchand, and V. Lempitsky, "Domain-adversarial training of neural networks," *The Journal of Machine Learning Research*, vol. 17, no. 1, pp. 2096–2030, 2016.
- [8] M. Long, Y. Cao, J. Wang, and M. Jordan, "Learning transferable features with deep adaptation networks," in *Proceedings of the 32nd International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, vol. 37. PMLR, 2015, pp. 97–105.
- [9] M. Yamada, L. Sigal, and Y. Chang, "Domain adaptation for structured regression," *International journal of computer vision*, vol. 109, no. 1-2, pp. 126–145, 2014.
- [10] A. Shrivastava, T. Pfister, O. Tuzel, J. Susskind, W. Wang, and R. Webb, "Learning from simulated and unsupervised images through adversarial training," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 3, no. 4, 2017, p. 6.
- [11] K. Bousmalis, A. Irpan, P. Wohlhart, Y. Bai, M. Kelcey, M. Kalakrishnan, L. Downs, J. Ibarz, P. Pastor, K. Konolige *et al.*, "Using simulation and domain adaptation to improve efficiency of deep robotic grasping," in *International Conference on Robotics and Automation*, 2018.
- [12] D. Eigen and R. Fergus, "Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture," in *ICCV*, 2015, pp. 2650–2658.
- [13] L.-C. Chen, Y. Yang, J. Wang, W. Xu, and A. L. Yuille, "Attention to scale: Scale-aware semantic image segmentation," in *CVPR*, 2016, pp. 3640–3649.
- [14] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs," *TPAMI*, vol. 40, no. 4, pp. 834–848, 2018.
- [15] C. Liang-Chieh, G. Papandreou, I. Kokkinos, K. Murphy, and A. Yuille, "Semantic image segmentation with deep convolutional nets and fully connected crfs," in *ICLR*, 2015.
- [16] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 3431–3440.
- [17] V. Badrinarayanan, A. Kendall, and R. Cipolla, "Segnet: A deep convolutional encoder-decoder architecture for image segmentation," *TPAMI*, 2017.
- [18] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *MICCAI*. Springer, 2015, pp. 234–241.
- [19] S. Zheng, S. Jayasumana, B. Romera-Paredes, V. Vineet, Z. Su, D. Du, C. Huang, and P. H. Torr, "Conditional random fields as recurrent neural networks," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1529–1537.
- [20] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 2881–2890.
- [21] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation," in *ECCV*, 2018.
- [22] V. M. Patel, R. Gopalan, R. Li, and R. Chellappa, "Visual domain adaptation: A survey of recent advances," *IEEE signal processing magazine*, vol. 32, no. 3, pp. 53–69, 2015.
- [23] M. Wang and W. Deng, "Deep visual domain adaptation: A survey," *Neurocomputing*, 2018.
- [24] G. Csurka, "A comprehensive survey on domain adaptation for visual applications," in *Domain Adaptation in Computer Vision Applications*. Springer, 2017, pp. 1–35.
- [25] B. Gong, Y. Shi, F. Sha, and K. Grauman, "Geodesic flow kernel for unsupervised domain adaptation," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*. IEEE, 2012, pp. 2066–2073.
- [26] R. Gopalan, R. Li, and R. Chellappa, "Domain adaptation for object recognition: An unsupervised approach," in *Computer Vision (ICCV), 2011 IEEE International Conference on*. IEEE, 2011, pp. 999–1006.
- [27] M.-Y. Liu and O. Tuzel, "Coupled generative adversarial networks," in *Advances in neural information processing systems*, 2016, pp. 469–477.
- [28] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.
- [29] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *The IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.
- [30] J. Hoffman, D. Wang, F. Yu, and T. Darrell, "Fcns in the wild: Pixel-level adversarial and constraint-based adaptation," *arXiv preprint arXiv:1612.02649*, 2016.
- [31] Y. Ganin and V. Lempitsky, "Unsupervised domain adaptation by backpropagation," in *International Conference on Machine Learning*, 2015, pp. 1180–1189.
- [32] Y. Zhang, P. David, and B. Gong, "Curriculum domain adaptation for semantic segmentation of urban scenes," in *The IEEE International Conference on Computer Vision (ICCV)*, vol. 2, no. 5, 2017, p. 6.
- [33] "Cycada: Cycle consistent adversarial domain adaptation," in *International Conference on Machine Learning (ICML)*, 2018.
- [34] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in neural information processing systems*, 2014, pp. 2672–2680.