POLITECNICO DI TORINO
Repository ISTITUZIONALE

Reimagine BiSeNet for Real-Time Domain Adaptation in Semantic Segmentation

Original
Reimagine BiSeNet for Real-Time Domain Adaptation in Semantic Segmentation / Tavera, A., Masone, C., Caputo, B.. - ELETTRONICO. - (2021), pp. 33-36. (Italian Institute of Robotics and Intelligent Machines (2021 I-RIM Conference) Rome 08/10/2021 - 10/10/2021) [10.5281/zenodo.6367918].

Availability:
This version is available at: 11583/2924952 since: 2022-07-19T15:35:50Z

Publisher:
Zenodo

Published
DOI:10.5281/zenodo.6367918

Terms of use:

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

Publisher copyright

(Article begins on next page)

12 September 2026

2021 I-RIM Conference
October 8-10, Rome, Italy
ISBN: 9788894580525
DOI:10.5281/zenodo.6367918

Reimagine BiSeNet for Real-Time Domain Adaptation in Semantic Segmentation

Antonio Tavera
DAUIN Politecnico di Torino
antonio.tavera@polito.it

Carlo Masone
CINI, Politecnico di Torino

Barbara Caputo
DAUIN Politecnico di Torino
barbara.caputo@polito.it

***Abstract*—Semantic segmentation models have reached remarkable performance across various tasks. However, this performance is achieved with extremely large models, using powerful computational resources and without considering training and inference time. Real-world applications, on the other hand, necessitate models with minimal memory demands, efficient inference speed, and executable with low-resources embedded devices, such as self-driving vehicles.**

In this paper, we look at the challenge of real-time semantic segmentation across domains, and we train a model to act appropriately on real-world data even though it was trained on a synthetic realm. We employ a new lightweight and shallow discriminator that was specifically created for this purpose. To the best of our knowledge, we are the first to present a real-time adversarial approach for assessing the domain adaption problem in semantic segmentation. We tested our framework in the two standard protocol: GTA5→Cityscapes and SYNTHIA→Cityscapes. Code is available at: https://github.com/taveraantonio/RTDA

***Index Terms*—Real-Time, Semantic Segmentation, Unsupervised Domain Adaptation, Autonomous Driving**

I. Introduction

Semantic segmentation, *i.e.*, assigning a semantic class to each pixel of an image, is a critical task for scene comprehension. It is fraught with challenges and the state-of-the-art models proposed to tackle them usually have a huge number of parameters. The complexity of these models not only translates to long training and inference times but it also makes it impractical to deploy them in a real-world scenario due to the large amount of resources demanded. Moreover, semantic segmentation id often required to work in real-time, particularly for robotics applications such as geo sensing, precision agriculture, and, most notably, autonomous driving.

Besides the complexity of the models, the process of collecting and annotating real-world data [1] is time-consuming and costly. A successful solution to tackle this issue is to use synthetic data generated from virtual world simulators [2], [3], [4]. Despite the much lower cost of collecting and annotating synthetic data, this technique has one major drawback: the domain shift between virtual and real world is substantial. Several unsupervised domain adaptation techniques have been proposed to address the domain gap between the synthetic (*source*) and real (*target*) domains; however, because they are not designed to be used in a real-world scenario and rely on a huge number of parameters, they are still vulnerable to resource and training time limits.

To fully solve the real-time domain adaptation problem in semantic segmentation, we require a complete lightweight model with few parameters and that can be deployed in a practical situation with limited resources. To do this, we redesigned the BiSeNet [5] model, tailoring it to the Domain Adaptation challenge and including a novel lighter and thinner fully convolutional domain discriminator (Light&Thin). To summarize:

- we propose a network for real-time domain adaptation in semantic segmentation, using a new lightweight and thin domain discriminator.
- we propose an ablation study to compare our Light&Thin discriminator to a standard domain discriminator and its lightweight variant.
- we test our architecture against two synthetic-to-real situations, GTA→Cityscapes and Synthia→Cityscapes, proving the efficacy of our solution.

II. Related Works

A. Semantic Segmentation and real-time application

Thanks to the use of deep learning techniques, Semantic Segmentation has exploded in popularity in recent years. The current state-of-the-art methods are determined by the approach employed to exploit semantic information, such as fully convolutional networks [6], encoder-decoder architectures [7], [8], dilated convolutions [9], [10], [11] or multi-scale and pyramid networks [12]. Because the number of parameters in semantic segmentation networks is in the order of 10^9 and their real-world application is rising in popularity, several researchers have investigated the feasibility of more lightweight architectures. The majority of architectures can be divided into two macro categories: (i) encoder-decoder architectures [13], [14], [15], which cost less at inference time than dilated convolution methods, (ii) two-pathway architectures, which address the loss of semantic information during the encoder-decoder mechanism's downsampling and upsampling operations. The BiSeNet family [5], [16] is an example of this type of architecture.

B. Domain Adaptation

The task of bridging the gap between two different distributions is referred to as Domain Adaptation. The original answer to this problem is to employ a distance minimization algorithm, such as the MMD [17], although alternative methods

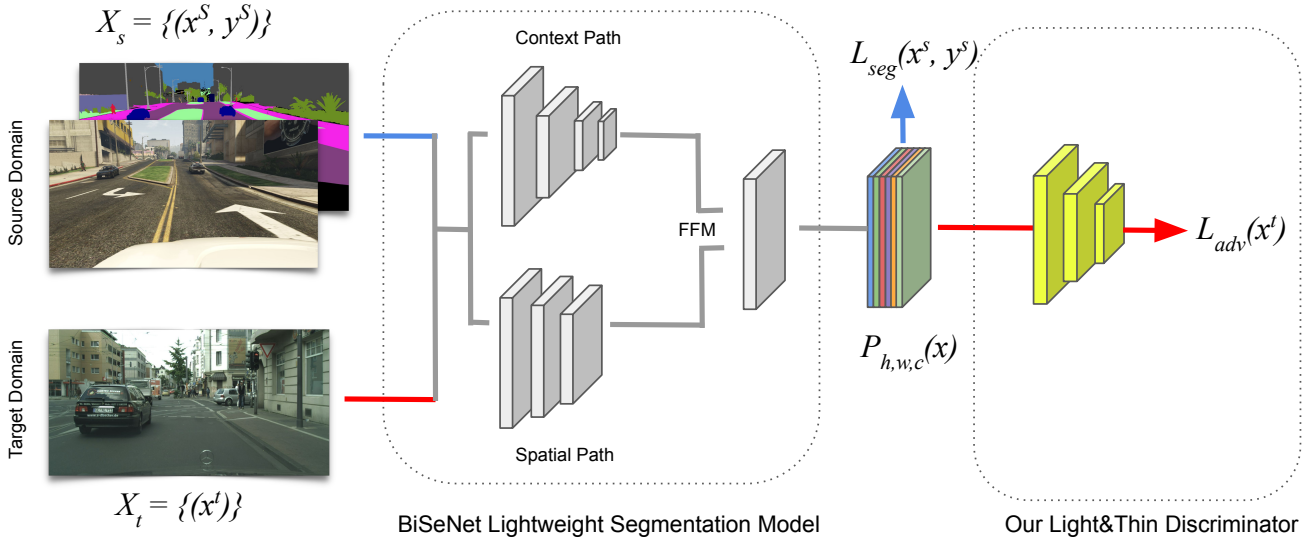

Fig. 1. Illustration of the real-time adversarial training of our framework. The adversarial loss required to align the source and target distributions is computed by our novel lightweight and shallow discriminator.

that use generative models [18], [19] to condition one domain into the other have also been used. The most noteworthy solution is the adversarial training technique [20], [21], which consists in a min-max game between the segmentation network and a discriminator in which the former attempts to trick the latter by making the distributions of the two domains identical. In any case, none of the prior solutions were applicable to real-world scenario.

III. Method

The proposed algorithm re-imagines BiSeNet tailored to the Unsupervised Domain Adaptation (UDA) task (III-A). We introduce a novel real-time adversarial domain adaption framework (III-B) comprised of the BiSeNet semantic segmentation model and an unique lightweight and thin discriminator (Fig. 1) that increases domain alignment and adaptation performance.

A. Setting

The set of RGB images composed by $\mathcal{I}$ pixels is denoted by $\mathcal{X}$, and the set of semantic labels linking each pixel I with a class c from a set of semantic classes $\mathcal{C}$ is denoted by $\mathcal{Y}$. We have two datasets to work with during training: the source $X_s = \{(x^s, y^s)\}$, which consists of $|X_s|$ semantically annotated images, and the target $X_t = \{(x^t)\}$, which consists of $|X_t|$ unlabeled images. The source and target annotation mask belonging to the set of semantic labels $\mathcal{Y}$ are defined as y^s and y^t. The goal of UDA is to use both the source and target dataset X_s and X_t to learn a function f that takes as input an image x and outputs a C-dimensional segmentation map $P_{h,w,c}(x)$.

B. Training

Due to the lack of semantic information for the target distribution, we proceed to align the features derived from the source and target domains in an adversarial fashion. To do this, as well as to meet our goal of making a network smaller, portable, and deployable on limited resource devices, we require a different domain discriminator. This is why we developed and tested two different types of lighter discriminators: a less expensive version (D_{Light}) of the widely used Fully Convolutional discriminator [22] and a shallow version ($D_{Light\&Thin}$) of the latter. Both discriminators D employ depthwise separable convolution instead of the conventional convolution and are trained to discriminate between source and target domains using the following loss:

$$L_D(x^s, x^t) = -\sum_{h,w} \log\ D(P^s_{h,w,c}) + \log(1 - D(P^t_{h,w,c})). \quad (1)$$

More details on these two lightweight discriminators are presented in Sec. IV.

The adversarial training is carried out using the features extracted by the semantic segmentation model and the domain prediction coming from a discriminator model. Both models engage in a min-max game in which the discriminator guesses the domain to which a feature belongs to and the segmentation network attempts to mislead the discriminator by making features from both domains similar. To accomplish this effect an adversarial loss L_{adv} is used as follow:

$$L_{\text{adv}}(x^t) = -\frac{1}{|X_t|} \sum_{h,w} \log D(P^t_{h,w,c}). \quad (2)$$

We jointly optimize the supervised segmentation loss L_{seg} on source samples and the unsupervised entropy loss L_{adv} on target samples while training the BiSeNet semantic segmentation model. The following is the definition of the total loss function:

$$\frac{1}{|X_s|} \sum_{(x^s, y^s) \in X_s} L_{\text{seg}}(x^s, y^s) + \frac{1}{|X_t|} \sum_{x^t \in X_t} + \lambda L_{adv}(x^t), \quad (3)$$

Experiment	Road	Sidewalk	Building	Wall	Fence	Pole	TLight	TSign	Vegetation	Terrain	Sky	Person	Rider	Car	Truck	Bus	Train	Motorcycle	Bicycle	mIoU^{19}
Target Only	97.11	77.88	88.67	48,31	48.31	41.33	39.56	48.36	58.27	89.07	57.03	91.86	66.45	40.47	90.63	60.23	67.11	50.32	44.93	64.17
FCD	74.21	28.66	72.47	12.57	16.31	12.72	28.03	17.32	80.17	14.64	77.31	41.97	8.88	65.58	18.2	6.85	18.33	11.71	0.0	31.89
FCD-Light	83.17	33.53	68.9	11.37	7.59	13.46	25.12	14.51	79.49	30.09	74.97	41.47	13.61	67.73	19.84	7.05	4.92	14.63	0.0	32.18
FCD-LightThin	83.92	37.21	74.23	14.19	15.63	17.61	29.93	19.16	79.85	24.91	72.14	43.24	11.15	61.0	17.41	14.28	7.16	8.22	0.0	**33.22**

TABLE I
GTA5→CITYSCAPES UNSUPERVISED DOMAIN ADAPTATION EXPERIMENTS. FCD STANDS FOR FULLY CONVOLUTIONAL DISCRIMINATOR. FCD-LIGHT INDICATES OUR LIGHTWEIGHT VARIANT. FCD-LIGHT&THIN INDICATES OUR THINNER AND LIGHTWEIGHT DISCRIMINATOR.

Experiment	Road	Sidewalk	Building	Wall	Fence	Pole	TLight	TSign	Vegetation	Sky	Person	Rider	Car	Bus	Motorcycle	Bicycle	mIoU^{16}
Target Only	97.11	77.88	88.67	48,31	48.31	41.33	39.56	48.36	58.27	57.03	91.86	66.45	40.47	60.23	50.32	44.93	64.17
FCD	72.74	32.18	75.31	4.45	0.35	14.07	0.09	2.58	66.39	80.87	35.84	3.32	54.26	18.08	1.49	9.18	29.45
FCD-Light	68.02	33.17	74.76	8.69	0.32	16.41	6.25	4.77	56.92	80.67	37.33	4.64	50.61	17.7	3.49	16.63	30.02
FCD-LightThin	63.01	23.5	76.94	8.71	0.74	19.93	9.04	7.52	76.56	79.98	44.01	4.29	63.76	14.56	1.99	11.97	**31.66**

TABLE II
SYNTHIA→CITYSCAPES UNSUPERVISED DOMAIN ADAPTATION EXPERIMENTS. FCD STANDS FOR FULLY CONVOLUTIONAL DISCRIMINATOR. FCD-LIGHT INDICATES OUR LIGHTWEIGHT VARIANT. FCD-LIGHT&THIN INDICATES OUR THINNER AND LIGHTWEIGHT DISCRIMINATOR.

where L_{seg} minimize the standard cross-entropy loss defined as:

$$L_{\text{seg}}(x^s, y^s) = -\frac{1}{|X_s|} \sum_{h,w} \sum_{c} y^s_{h,w,c} \log(P^s_{h,w,c}) \qquad (4)$$

IV. EXPERIMENTS

A. Datasets

We test our model over the two standard synthetic-to-real benchmarks in Domain Adaptation for Semantic Segmentation: GTA5→Cityscapes and SYNTHIA→Cityscapes.

GTA5 [2] is made up of 24966 annotated photos from the aforementioned video-game. The standard 19-classes, which Cityscapes shares, is used for training and evaluation.

SYNTHIA [3] is made up of 9400 annotated images from a virtual world and belonging to the RAND-CITYSCAPES subset. The usual 19-classes shared by Cityscapes are utilized for training, whereas the assessment is performed on 16-classes using the [21] protocol.

Cityscapes [1] is made up of 2975 real-world pictures gathered from various German cities. To test our network, we use the entire validation set of 500 photos at the original 2048x1024 resolution.

B. Implementation details

The segmentation model of our method is BiSeNet [5] with the Context Path (see section 3.2 of [5]) initialized with a ResNet-101 [23] pretrained on ImageNet. The standard discriminator used for the comparison is a common Fully Convolutional Discriminator (FCD) with 5 convolution layers with kernel size $4x4$, channel numbers $\{64, 128, 256, 512, 1\}$, padding 2 and stride 1. Its lightweight variant (FCD-Light) is obtained by substituting each convolution operation with a depthwise-separable convolution [24], comprises of a depthwise convolution done independently over each input channels, followed by a pointwise convolution, with kernel size $1x1$. Our thinner version (FCD-Light&Thin) has only 3 depthwise separable convolution layers with channel numbers $\{64, 128, 1\}$. Each convolution or depthwise separable convolution layer is followed by a Leaky ReLU with negative slope 0.2.

PyTorch is used to implement our technique. The segmentation model is trained with batch size 4 and SGD with an initial learning rate of $2.5 \times 10 - 4$, which is then changed at each iteration with a "poly" learning rate decay with power 0.9, momentum 0.9, and weight decay 0.0005. Adam is used to train all of the discriminators, with momentum $(0.9, 0.99)$, learning rate $10 - 5$, and the same segmentation model scheduler. The model has undergone $30k$ iterations of training. The value of λ_{adv} is set to 0.01. The training images are shrunk to $(1024, 512)$, whereas the evaluation is done on the $(2048, 1024)$ original image dimension.

We use the standard Intersection over Union metric to measure the performance of our experiments.

C. Results

Table I and Table II show the result on the GTA→Cityscapes and SYNTHIA→Cityscapes, respectively. By looking at Table I, it is clear that using a typical Fully Convolutional discriminator (FCD) we get performances that are approximately half of what we would achieve if we trained directly on the target. When each convolution in this discriminator is replaced with its lightweight counterpart (FCD-Light), we get comparable results with just a $+0,29\%$ gain in accuracy. However, as seen in Table III, the number of parameters and FLOPS decreases significantly, as does training and inference time. When the input resolution is 1024x512, the difference in parameters is 2.59 million, while the FLOPS move from 30.883G to barely 2.14G. When we use our light and shallow discriminator (Light&Thin), the reduction in parameters and FLOPS is proportionate to an enhancement in accuracy; indeed, our solution improves performance by $+1,33\%$ over the typical FCD. Since the task is classification, using a shallow domain discriminator like ours takes less epochs to attain a local optima than a conventional DCGAN discriminator, which would require more epochs and longer training time to converge. We would want to emphasize that all of this results were collected while training on two TESLA v100 GPUs rather than on commercial hardware such as a Jetson Javier. The SYNTHYA→Cityscapes experiment described in Table II

Fig. 2. Qualitative results for the GTA→Cityscapes experiment. Starting from the left: RGB, FCD, FCD-Light, FCD-Light&Thin, Ground Truth.

	FCD	FCD-Light	FCD-Light&Thin
Parameters	2.781M	0.191M	13.316K
FLOPS	30.883G	2.14G	1.038G
Training Time	7h:04m	6h:39m	6h:32m

TABLE III
COMPARISON BETWEEN THE NUMBER OF PARAMETERS, FLOPS AND TRAINING TIME AMONG THE THREE DISTINCT DISCRIMINATORS USED.

shows a similar pattern. Replacing common convolutions with depthwise separable convolutions results in a small $+0.57\%$ improvement, but when utilizing our Light&Thin discriminator, an average boost of $+2,21\%$ is attained. Figure 2 confirms this tendency; as you can see, our Light&Thin model allows for better segmentation, even for small classes like pedestrians, poles or traffic signs. It should be noted that these results come from models that were trained on synthetic data with a distribution that is substantially different from the real-world test set. There is still work to be done to improve performance and bridge the gap between the two domains and the existing state-of-the-art but non-real-time domain adaptation models.

V. CONCLUSION

In this paper, we look at Real Time Domain Adaptation in Semantic Segmentation. The primary goal is to minimize model parameters as well as training and inference time in order to make the model feasible for real-world applications. We present a whole lightweight framework that includes a unique light and shallow discriminator. We evaluated our approach using the two common synthetic-to-real protocols. The results indicate that there is still work to be done in this task; future research will focus on applying our discriminator to more complex and powerful lightweight semantic segmentation models, as well as enhancing the entire framework.

REFERENCES

[1] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2016.

[2] Stephan R. Richter, Vibhav Vineet, Stefan Roth, and Vladlen Koltun. Playing for data: Ground truth from computer games. In Bastian Leibe, Jiri Matas, Nicu Sebe, and Max Welling, editors, *Eur. Conf. Comput. Vis.*, volume 9906 of *LNCS*, pages 102–118. Springer International Publishing, 2016.

[3] German Ros, Laura Sellart, Joanna Materzynska, David Vazquez, and Antonio M. Lopez. The SYNTHIA dataset: A large collection of synthetic images for semantic segmentation of urban scenes. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 3234–3243, June 2016.

[4] Emanuele Alberti, Antonio Tavera, Carlo Masone, and Barbara Caputo. IDDA: A large-scale multi-domain dataset for autonomous driving. *IEEE Robot. and Autom. Lett.*, 5(4):5526–5533, 2020.

[5] Changqian Yu, Jingbo Wang, Chao Peng, Changxin Gao, Gang Yu, and Nong Sang. Bisenet: Bilateral segmentation network for real-time semantic segmentation. In *ECCV*, pages 334–349, 2018.

[6] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 3431–3440, 2015.

[7] Hyeonwoo Noh, Seunghoon Hong, and Bohyung Han. Learning deconvolution network for semantic segmentation. *arXiv preprint arXiv:1505.04366*, 2015.

[8] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. volume 9351, pages 234–241, 10 2015.

[9] L. 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. *IEEE Trans. Pattern Anal. Mach. Intell.*, 40(4):834–848, 2018.

[10] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Eur. Conf. Comput. Vis.*, pages 801–818, 2018.

[11] Liang-Chieh Chen, G. Papandreou, Florian Schroff, and H. Adam. Rethinking atrous convolution for semantic image segmentation. *ArXiv*, abs/1706.05587, 2017.

[12] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 2881–2890, 2017.

[13] Sachin Mehta, Mohammad Rastegari, Anat Caspi, Linda Shapiro, and Hannaneh Hajishirzi. Espnet: Efficient spatial pyramid of dilated convolutions for semantic segmentation. In *ECCV*, 2018.

[14] Eduardo Romera, José M. Álvarez, Luis M. Bergasa, and Roberto Arroyo. Erfnet: Efficient residual factorized convnet for real-time semantic segmentation. *IEEE Transactions on Intelligent Transportation Systems*, 19(1):263–272, 2018.

[15] X. Zhang, X. Zhou, Mengxiao Lin, and Jian Sun. Shufflenet: An extremely efficient convolutional neural network for mobile devices. *CVPR*, pages 6848–6856, 2018.

[16] Changqian Yu, Changxin Gao, Jingbo Wang, Gang Yu, Chunhua Shen, and Nong Sang. Bisenet v2: Bilateral network with guided aggregation for real-time semantic segmentation, 2020.

[17] Bo Geng, Dacheng Tao, and Chao Xu. DAML: domain adaptation metric learning. *IEEE Trans. Image Process.*, 20(10):2980—2989, October 2011.

[18] Yunsheng Li, Lu Yuan, and Nuno Vasconcelos. Bidirectional learning for domain adaptation of semantic segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 6936–6945, 2019.

[19] Myeongjin Kim and Hyeran Byun. Learning texture invariant representation for domain adaptation of semantic segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 12975–12984, 2020.

[20] Wei-Lun Chang, Hui-Po Wang, Wen-Hsiao Peng, and Wei-Chen Chiu. All about structure: Adapting structural information across domains for boosting semantic segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 1900–1909, 2019.

[21] Tuan-Hung Vu, Himalaya Jain, Maxime Bucher, Matthieu Cord, and Patrick Pérez. ADVENT: Adversarial entropy minimization for domain adaptation in semantic segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 2517–2526, 2019.

[22] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks, 2016.

[23] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2016.

[24] Francois Chollet. Xception: Deep learning with depthwise separable convolutions. pages 1800–1807, 07 2017.