Real-time Domain Adaptation in Semantic Segmentation

Antonio Ferrigno
Politecnico di Torino

Giulia Di Fede
Politecnico di Torino

Vittorio Di Giorgio
Politecnico di Torino

Abstract

In the rapidly evolving landscape of computer vision, the demand for real-time semantic segmentation models is increasing quickly. This project addresses two main challenges posed by this demand: firstly, the extreme high cost in terms of time of collecting pixel-wise annotations and, secondly, the trade-off between performance and efficiency in the case of real-time scenarios, where performance is necessarily degraded in favour of efficiency. By employing the Short-Term Dense Concatenate network, we tackle the challenge posed by the high demand of training data in the field of Real-time Semantic Segmentation by evaluating the domain shift between source and target domains and by performing Adversarial Domain Adaptation via adversarial learning in the output space. In order to further test the opportunities in the field we also propose to explore three expansions: (1) Depthwise Separable Convolution, to exploit a more efficient discriminator network; (2) Fourier Domain Adaptation, to test a different kind of Domain Adaptation technique; (3) Self-Supervised Training with Multi-Band Transfer, to compare an unsupervised domain adaptation method with a self-supervised approach. Eventually, we proved Unsupervised Domain Adaptation to be capable of further reducing the domain gap between Source and Target Domain.

1. Introduction

Semantic Segmentation is a fundamental yet challenging task in computer vision for which the objective is to assign a specific label to each pixel in an image. For this reason, it is useful in a wide range of real-world applications such as autonomous driving, computerised medical diagnosis, defect detection and so on [9].

In the recent years, the field of Semantic Segmentation has been influenced by remarkable advancements, fuelled by the high demand for more high-performance models in continuous growth. With the advent of Fully Convolutional Networks (FCNs), semantic segmentation has experienced an important phase in computer vision, in which the segmentation accuracy was dramatically increased [16].

With the advancements in interactive applications, Semantic Segmentation has also recently been applied to the real-time fields of augmented reality and video surveillance which, along with autonomous driving, nowadays require an efficient inference speed for rapid interaction response. To meet these demands, many researchers have focused their efforts on developing Convolutional Neural Networks (CNNs) models that demonstrate the ability to obtain a good segmentation accuracy while maintaining a low latency inference. These real-time segmentation approaches have demonstrated a promising performance [24].

Another important challenge that often arises in the field of deep learning is the need for extensive datasets. In fact, the availability and quality of annotated data can directly impact the performance of deep learning models. However, the manual retrieval and annotation of the needed data is a strenuous and time-consuming activity. An alternative solution is given by Domain Adaptation techniques, by which a model is trained on another large-scale simulated "source" domain (e.g. computer-generated datasets) and applied in an unlabelled real-world "target" domain [26].

In this work, we specifically explore the field of Real-Time Domain Adaption in Semantic Segmentation by exploiting the Short-Term Dense Concatenate (STDC) network proposed by M. Fan et al. [7], a novel hand-crafted network built to obtain a competitive performance to that of existing methods with faster inference speed through an explainable structure.

Since source and target domain can show significant differences in terms of light conditions, weather variability and so on, the challenge of effective domain adaptation persists. Therefore, in this work we propose to first asses the domain gap between the source and target domain by evaluating the performance of the STDC network trained on the simulated source domain and subsequently tested on a real-world target domain.

After trying to reduce the domain shift by employing Augmentation techniques we propose to implement an unsupervised adversarial domain adaptation algorithm like in [21] in order to reduce the domain gap. This technique,

based on a Generative Adversarial Network (GAN), involves training a model in a way that the feature distributions of the source and target domains are made similar. The proposed model consists of a segmentation model (1) with the task to predict output results, and a discriminator with the goal to distinguish whether the input is from the source or target segmentation output (2). With an adversarial loss, the proposed segmentation model aims to fool the discriminator.

To further experiment on this field we decided to investigate the following extensions:

- **Depthwise Separable Convolution**. A variation of the discriminator which aims to reduce the computational cost and the number of parameters, making it suitable for real-time segmentation. [4]
- **Fourier Domain Adaptation**. A simple method for unsupervised domain adaptation, by which the difference between the source and target distributions is reduced by swapping the low frequency spectrum of one with the other. [23]
- **Self-supervised Training with Multi-band Transfer**. Self-supervised technique which uses pseudo-labels generated for the target domain to try improve the performance. [23]

Code available at https://github.com/GiuDF1102/Real-time-Domain-Adaptation-in-Semantic-Segmentation.git

2. Related Work

2.1. Semantic Segmentation

Given an image, Semantic Segmentation represents the task involving the assignment of a category label to its every individual pixel. Therefore, semantic segmentation aims to partition an image into meaningful mutually exclusive regions. The training of a semantic segmentation model involves the training of a deep neural network on a sufficient amount of finely annotated data. The learning process helps the network build awareness on the connection between high-level semantic concepts and low-level features [9].

Plenty of advancements in the field of Semantic Segmentation were enabled by the implementation of CNNs. The VGG network introduced by Simonyan et. al [20] introduces deeper structures for improved performance. The Deep Residual Network (ResNet) by He et al. [11] gives the key contribution by incorporating its residual representation, solving the problem of training very deep neural networks. DenseNet [14] made another breakthrough, by introducing the use of densely connected blocks, promoting the key idea based on the reuse of features across the network.

Other main advancements reached by scholars brought to many promising solutions by using techniques such as the enlargement of the receptive field [3, 5], integrating encoder-decoder architectures [2], taking advantage of the hierarchical structure of the world [15] and so on. Most recently, a new family of architectures based of the mechanism of attention named "Transformer" [22] is being applied on semantic segmentation challenges [8, 10], yielding impressive state-of-the-art performance.

While the cited architectures achieve high accuracy, they present two main challenges which are represented by the requirement for a huge amount of finely annotated data essential for training and by the incompatibility with the current real-time applications due to their elevated computational demands.

In this work we propose to tackle these challenges by exploiting Domain Adaptation and Real-Time Semantic Segmentation solutions.

2.2. Real-Time Semantic Segmentation

Although the above cited architectures achieve high accuracy, they pose another problem which is represented by their incompatibility with the current real-time applications caused by their excessive computational demands. As shown by Fan et al. [7], two main paths have been explored by the academy to reduce the lack of efficiency of current semantic segmentation models: lightweight structures which incorporate the attention mechanism like MobileNets [13] and ShuffleNets [25]; and multi-branch architectures such as BiSeNet [24]. Fast-SCNN [17] is suited to efficient computation on embedded devices with low memory.

While BiSeNet utilizes a two-stream structure to enhance the accuracy of the segmentation results and take advantage of the joint information given by the fusion of low-level details and high-level context information, this strategy is affected by redundancy. Instead, the main breakthrough of the STDC network is given by the exploitation of a single-stream architecture guided by details aggregation.

Therefore, in this work we employ the Short-Term Dense Concatenate (STDC) network [7] to achieve a good trade-off between performance and efficiency.

2.3. Domain Adaptation

Academic research is aware of the important challenges posed by the retrieval of data essential for the training of these massive and extremely powerful architectures. In fact, the current state-of-the-art performance can be reached with a substantial amount of pixel-wise annotations. Hence, the challenge of gathering and labelling extensive sets of images covering an ample range of conditions has urged the development of synthetic datasets such as SYNTHIA [19], IDDA [1] and GTAV [18]. To overcome such limitations,

in this project we propose to tackle this challenge by leveraging Domain Adaptation techniques on the video-realistic GTAV dataset.

In the realm of semantic segmentation, recent advancements were made in terms of innovative approaches to address the challenges of domain adaptation. Most of these methods are based on the concept of Unsupervised Domain Adaption (UDA), which tries to solve the domain shift problem by using a generator model which aims to trick a discriminator model tasked with telling apart real images from those produced by the generator. For example, CyCADA [12] employs cycle consistency inspired by cycleGANs [27] to further reduce the domain gap. It involves mapping images from one domain to another and then back again, ensuring that the original and reconstructed images maintain their semantic content.

In this work, to tackle the Domain Shift problem we propose to follow the method introduced by Tsai et al. [21], that consists into the generator model that tries to fool the discriminator whose goal is to distinguish real images from those generated by the generator. Even if this method involves the construction of a multi-level adversarial network to perform domain adaptation at different levels, our decision was to perform such adaptation with one single discriminator.

3. Method

3.1. Short-Term Dense Concatenate Network

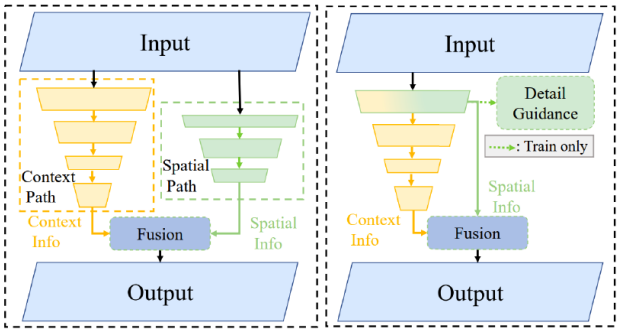


Figure 1. Standard ResNet architecture on the left and STDC architecture on the right, as shown in [7]

The baseline model adopted in this project is based on the BiSeNet architecture by Yu et al. [24] which has demonstrated great results. However, the additional path which helps in the extraction of detailed low-level features is time-consuming. To provide better inference speed Fan et al. [7] propose the following contributions:

- **Short-Term Dense Concatenate Module**. A novel structure which is able to obtain adaptable receptive fields with the use of a low number of parameters. The STDC modules are then integrated into the U-net architecture to form the STDC Network. A STDC module is seperated into several blocks, each of which comprises one convolutional layer, one batch normalization layer and a ReLU activation layer. The final output of the STDC module is calculated using a fusion by concatenation of the features maps produced by each block, gathering multi-scale information from all blocks.

- **Detail Guidance**. The Decoding Network still takes advantage of the Attention Refine Module (ARM) for the refinement of the combination features and the Features Fusion Module (FFM) adopted in BiSeNet.

3.2. Unsupervised Adversarial Domain Adaptation with Single Level Adversarial Learning

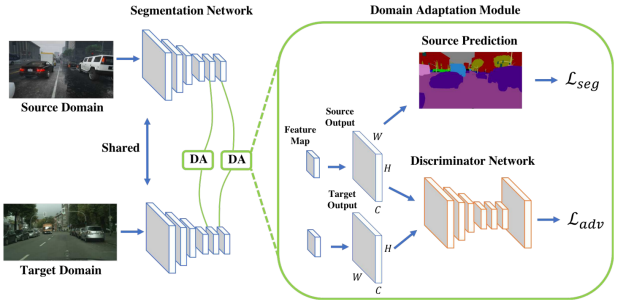


Figure 2. Algorithmic view of the Adversarial Domain Adaptation technique implemented in [21]

The architecture we used [21] consists of a segmentation network G that predicts pixel-wise labels and a discriminator network D that distinguishes between source and target domain segmentation. We denote the two set of images from the target and source domain in the $R^{H\times W\times C}$ space respectively $\{I_t\}$ and $\{I_s\}$. The source image I_s is forwarded to the segmentation network G for optimization, P_s is obtained, the target image I_t is then forwarded as well to obtain a segmentation softmax output P_t. The two obtained predictions are used as the input to the discriminator D which must recognize if the input is from the source or the target domain. By means of an adversarial loss on the target prediction the gradients are propagated from D to G, this way G is pushed towards the generation of similar segmentation distributions in the target domain to the source prediction.

Discriminator Training. The adversarial discriminator loss $\mathcal{L}_d(P)$ is computed using a cross-entropy function, where z indicates the domain source (1) or target (0), aiming to correctly classify the domain of each input.

$$\mathcal{L}_d(P) = -\sum_{h,w} (1-z)\log D(P)^{(h,w,0)} + z \log D(P)^{(h,w,1)}$$

Segmentation Training. The segmentation loss for the source domain is a cross-entropy loss between the ground truth Y_s and the predicted probabilities P_s, summed over all pixels (h, w) and channels c in C.

$$\mathcal{L}_{seg}(I_s) = \sum_{h,w} \sum_{c \in C} Y_s^{(h,w,c)} \log P_s^{(h,w,c)}$$

The adversarial loss is designed to fool the discriminator into classifying the target predictions P_t as if they were from the source, thereby aligning the target predictions with the source domain distribution.

$$\mathcal{L}_{adv}(I_t) = -\sum_{h,w} \log D(P_t)^{(h,w,1)}$$

The overall objective is a combination of segmentation loss $\mathcal{L}_{seg}(I_s)$ for the source domain and an adversarial loss, $\mathcal{L}_{adv}(I_t)$, regulated by a trade-off parameter λ_{adv}.

$$\mathcal{L}(I_s, I_t, Y_s) = \mathcal{L}_{seg}(I_s) + \lambda_{adv}\mathcal{L}_{adv}(I_t)$$

3.3. Depthwise Separable Convolution

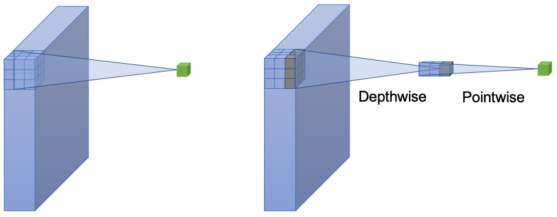


Figure 3. Standard convolution and depthwise separable convolution. Figure from [4]

While the standard convolution used in our discriminator applies a filter across all channels simultaneously, which is computationally intensive, Depthwise separable convolution decomposes this process into two stages as shown in Figure 3:

1. **Depthwise convolution**. (1) An input tensor of 3 dimensions is split into separate channels, (2) for each channel, the input is convolved with a $2D$ Filter, then (3) the output of each channel is then stacked together to get the output on the entire $3D$ tensor.

2. **Pointwise convolution**. The three channels are combined together to form an output tensor of n channels, with n as desired.

3.4. Fourier Domain Adaptation

Fourier Domain Adaptation (FDA) is an unsupervised technique for aligning source and target domains in semantic segmentation tasks. The initial transformation of an image x into the frequency domain is expressed as:

$$\mathcal{F}(x)(m,n) = \sum_{h,w} x(h,w) e^{-j2\pi(\frac{h}{H}m + \frac{w}{W}n)}, \quad j^2 = -1$$

$\mathcal{F}$ denotes the Fourier transform, capturing both amplitude and phase spectra, where (h, w) are the spatial domain coordinates, and (m, n) are the frequency domain coordinates.

The adaptation process involves the application of a mask to blend the amplitude spectra of the source and target domains:

$$M_\beta(h,w) = \mathbb{1}_{(h,w)\in[-\beta H:\beta H, -\beta W:\beta W]},$$

The adapted source image in the target domain is obtained by combining the masked target amplitude spectrum with the source spectrum, followed by an inverse Fourier transform:

$$x^{s\to t} = \mathcal{F}^{-1}([M_\beta \circ \mathcal{F}^A(x^t) + (1-M_\beta)\circ \mathcal{F}^A(x^s), \mathcal{F}^P(x^s)])$$

This results in an image with the content of the source domain and the frequency characteristics of the target domain, facilitating domain-invariant semantic segmentation without complex model retraining.

3.5. Self-supervised Training with Multi-Band Transfer

The compound loss function $\mathcal{L}$ for training the semantic segmentation network comprises three components: the cross-entropy loss computed on the adapted source dataset, the entropy loss on the target dataset and the cross-entropy loss on the target dataset with pseudo-labels.

$$\mathcal{L}(I_s, I_t, Y_s, \widehat{Y}_t) = \mathcal{L}_{seg}(I_s, Y_s) + \lambda_{adv}\mathcal{L}_{adv}(I_t) + \mathcal{L}_{seg}(I_t, \widehat{Y}_t)$$

I_s denotes the parameters of the semantic segmentation network, λ_{adv} is a regularization parameter that balances the contribution of the entropy loss, and I_t is the target dataset augmented with pseudo-labels. This loss function synergistically leverages labeled source data and unlabeled target data, thereby enabling the network to generalize better to the target domain.

Therefore, the Self-supervised Training with Multi-band Transfer (MBT) involves training multiple models with different β values in the Fourier Domain Adaptation (FDA) process.The pseudo-labels generated for the target domain by the ensamble of models are used to train the segmentation network in a self-supervised fashion.

(a) Source domain (b) Target domain (c) Source image in target style

Figure 4. An example of FDA for domain adaptation.

Model	road	sidewalk	building	wall	fence	pole	light	sign	vegetation	terrain	sky	person	rider	car	track	bus	train	motorcycle	bicycle	mIoU%	precision%
CS → CS (Upper Bound)	88.9	66.2	83.2	23.0	29.0	40.2	27.4	47.0	80.3	40.3	84.4	67.5	26.5	86.0	21.4	11.0	13.5	22.4	60.0	48.3	77.8
Base	75.5	0	35.9	1.2	0	8.43	0.2	1.1	27.4	2.8	41.1	2.7	0	9.8	1.5	0	0	0	0	11	45.3
Augm.	81.5	3.7	64.7	9.0	0.9	13.6	2.9	0.4	57.8	5.9	72.1	17.9	0.6	49.1	7.0	6.5	0.2	0	0	20.7	65.3
Adv. DA + Augm.	81.4	17.5	71.0	5.3	6.4	15.6	11.5	6.9	62.4	7.0	73.3	37.8	4.7	67.5	11.3	17.0	2.6	0	0	26.3	66.4
Adv. DA + Augm. + DSC	81.0	9.9	72.9	8.3	7.3	21.7	11.5	9.7	56.2	4.5	70.4	34.9	4.4	66.3	9.3	13.7	5.8	1.5	0	25.8	66.3
Adv. DA + Augm. + FDA	82.3	15.8	71.0	6.3	4.9	26.8	5.4	9.2	72.1	10.6	58.6	28.1	4.5	63.6	10.4	10.4	3.0	2.3	0	25.5	67.4
Adv. DA + Augm. + DSC + FDA	76.0	21.9	70.9	6.9	2.4	28.2	5.0	10.1	68.9	6.6	65.9	30.9	3.6	65.2	6.2	8.7	1.7	2.6	0	25.4	64.8
Adv. DA + Augm. + DSC + FDA + SSL	83.6	13.1	70.6	7.1	7.3	25.4	6.4	6.0	71.4	10.0	67.1	23.1	3.7	62.9	8.2	6.1	1.4	3.1	0.4	25.1	68.2

Table 1. Adversarial Domain Adaptation performance comparison using different techniques. The table presents mean Intersection over Union (mIoU%) and precision, compared to the results of the Cs → Cs (Upper Bound) which represents the upper limit achievable on the target domain.

4. Experimental Results

4.1. Datasets

To implement Domain Adaptation we focused on two main datasets: the source dataset and the target dataset. For both training and evaluation, only the 19 classes shared by Cityscapes and GTAV were considered in this project.

GTAV [18] was chosen as the source dataset. It consists of 2500 training frames. Each frame in the collected dataset has a resolution of 1914×1052 pixels which was resized to 1280x720 for training purposes. The dataset is highly variable in its content and layout.

Cityscapes was the chosen target dataset. It contains 1572 evaluation frames, each of them having a resolution of 2048x1024 which was resized to 1024x512. This dataset is also variable for content and layout, but, on the contrary of GTAV, also comprise different cities, which eventually add variability.

4.2. Implementation details

As mentioned in previous sections, the baseline model used for the segmentation task is the STDC Network by Fan et al. [7] pretrained on ImageNet [6]. We use stochastic gradient descent (SGD) with momentum 0.9 and weight decay $1 \cdot 10^{-4}$, and an initial learning rate of 0.01 with an applied decay procedure given by $init\_lr \cdot (1 - \frac{iter}{max\_iter}^{power})$, with power set to 0.9. We decided to maintain a batch size of 8 for all the course of our experiments and a number of epochs equal to 50. Since we encountered some hardware requirements issues, in some experiments we had to reduce the batch size to 7 in order to proceed. For the adaptation phase, a Fully Convolutional Discriminator which comprises 5 convolutional layers (kernel size 4×4, stride 2 and padding 1) was used with channels $(64, 128, 256, 512, 1)$. The optimizer employed for the discriminator was Adam, with a $1 \cdot 10^{-4}$ learning rate and the same polynomial decay employed in the segmentation network. The $\vec{\beta}$ parameters of Adam optimizer is set to (0.9, 0.99).

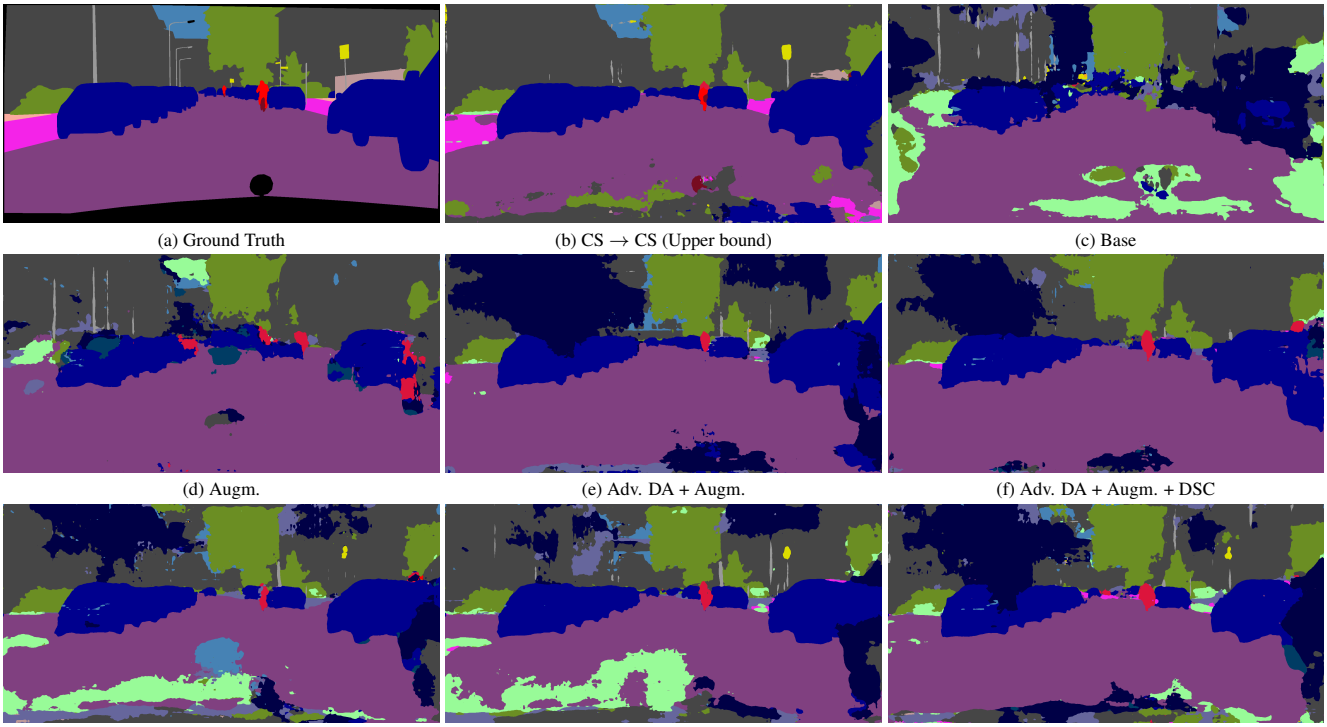
4.3. STDC

Our experiments started with the assessment of the domain gap between the source and target domain by evaluating the performance of the STDC network trained on the GTAV dataset and tested on the Cityscapes dataset.

First we defined the upper bound for the domain adaptation phase, then we trained the STDC Network on the GTAV synthetic dataset. Then we proceeded to test the STDC Network trained on the GTAV dataset on the Cityscapes dataset in order to evaluate the domain shift.

As it is visible in table 1 the STDC network provides good results when tested on the original domain, but there is a significant performance drop which highlights the domain shift when the training is executed in the synthetic source domain (GTAV).

In fact, the light and weather conditions in the two domains differ significantly. For this reason we attempted the adoption of Augmentation techniques in order to reduce the domain shift. As we reported in table 1 our best Augmenta-



(a) Ground Truth (b) CS → CS (Upper bound) (c) Base

(d) Augm. (e) Adv. DA + Augm. (f) Adv. DA + Augm. + DSC

(g) Adv. DA + Augm. + FDA (h) Adv. DA + Augm. + DSC + FDA (i) Adv. DA + Augm. + DSC + FDA ($\beta = 0.01$) + SSL

Figure 5. Example of the segmentation maps produced by our experiments.

tion technique gave us a 10% gain in mIoU.

4.4. Unsupervised Adversarial Domain Adaptation

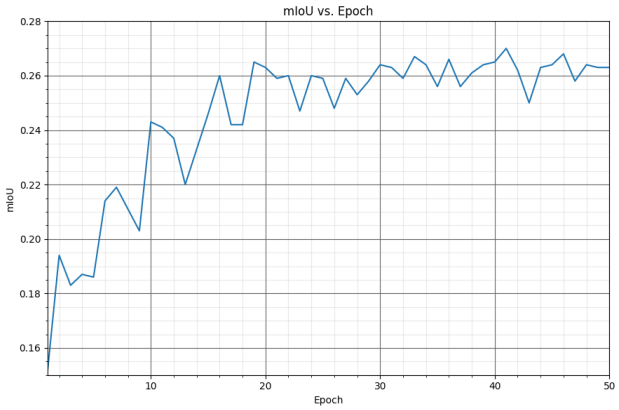


Figure 6. Mean Intersection over Union (mIoU) metric against the number of training epochs for the architecture described in [21] (Adv. DA + Data Augm.)

In our implementation of the method exposed by [21], we jointly train our STDC Segmentation network with the FCD discriminator, maintaining the best Augmentation settings discovered in the previous section. We follow the single level method with $\lambda_{adv} = 1 \cdot 10^{-3}$.

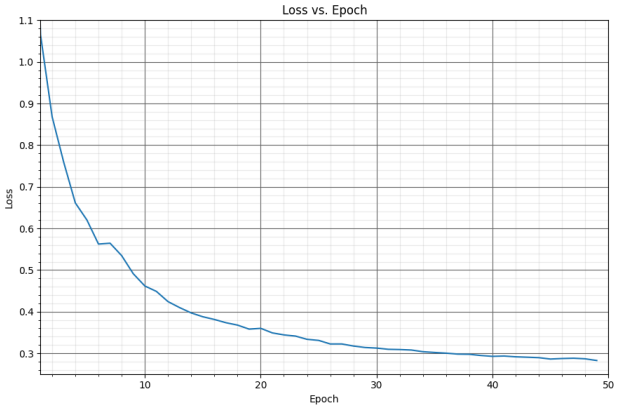


Figure 7. Loss against the number of training epochs for the architecture described in [21] (Adv. DA + Data Augm.)

As evident from table 1, the domain adaptation method proves useful, helping our network gain almost 6% in mIoU compared to the baseline model with Augmentation.

From the graph in figure 6 it is clear that the method is effective, despite the oscillations, the mIoU value becomes stable at ≈ 0.260.

Instead in figure 7 we can see the behaviour of the loss. As it is visible, the loss initially starts at a relatively high value and gradually decreases over time as the model un-

dergoes optimization, this indicates the model is learning smoothly. Beyond a certain point, the loss tends to stabilize, indicating that the model has converged to a certain extent. We suppose that the network would behave certainly better if the number of epochs was extended as we can see it doesn't reach a full plateau.

4.5. Depthwise Separable Convolution

As the first extension, we modified the discriminator employed in the Adversarial Domain Adaptation architecture by introducing a Depthwise Separable Convolution based network [4], in which each convolution layer in the FCD discriminator was replaced with a DSC layer.

We expect this extension to obtain worse performances compared to the earlier solution, but to be much more efficient because of the lower number of parameters needed.

In fact, from table 1 we can notice a slight decrease in performance (0.5%). However we consider this to be a good trade-off, since the main goal of this experiment was to also focus on Real-Time semantic segmentation.

4.6. Fourier Domain Adaptation

Given the differences observed between the target and the source domain we decide to further explore the possibilities in Domain Adaptation given by Fourier Domain Adaptation [23], a method which is based on aligning source and target domains by leveraging Fourier transformations.

The results obtained in our experiments were obtained by jointly evaluating the models obtained with the following β parameters $(0.01, 0.05, 0.09)$. The results are reported in Table 1.

We tested this extension with the FCD discriminator and then with the employment of the DSC discriminator and we noticed a slight drop in performance in both cases. We suppose this approach needs a work of fine-tuning in order to obtain better performances.

4.7. Self-supervised Training with MBT

For the Self-Supervised training, we used $\beta = (0.01, 0.05, 0.09)$ to train three STDC networks in the UDA architecture, then we produced the labels in order to proceed with a round of Self-Supervised Learning approach. To avoid self-referential issues and enhance regularization, the method applies a threshold to the confidence values of each prediction. Predictions with confidence in the top 66% or above 0.9 are accepted for each semantic class. The approach is further detailed with the observation that smaller β values lead to less variation, suggesting that adapted source datasets with smaller β are closer to the target dataset, thus imposing less bias. This method aims to align the source and target domains more closely, enhancing the effectiveness of the domain adaptation process.

5. Conclusions

In conclusion, our project represents a comprehensive exploration of real-time domain adaptation in semantic segmentation, leveraging the Short-Term Dense Concatenate network, adversarial learning, and extensions like Depthwise Separable Convolution, Fourier Domain Adaptation, and Self-supervised Training with Multi-band Transfer. Our experimentation and analysis demonstrates some interesting advancements in reducing the domain gap between synthetic and real-world datasets, thus enhancing the performance and efficiency of semantic segmentation models in real-time applications without the need to gather and label big amounts of data. The proposed methods seem to be promising for future research and practical applications, potentially leading to more robust and adaptable segmentation models. Our results could be significantly improved with a more exact hyperparameters search and more data, computational power at hand to prevent overfitting.

References

[1] Emanuele Alberti, Antonio Tavera, Carlo Masone, and Barbara Caputo. Idda: A large-scale multi-domain dataset for autonomous driving. *IEEE Robotics and Automation Letters*, 5(4):5526–5533, Oct. 2020. 2

[2] Vijay Badrinarayanan, Alex Kendall, and Roberto Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation, 2016. 2

[3] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L. Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs, 2017. 2

[4] Francois Chollet. Xception: Deep learning with depthwise separable convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017. 2, 4, 7

[5] Jifeng Dai, Haozhi Qi, Yuwen Xiong, Yi Li, Guodong Zhang, Han Hu, and Yichen Wei. Deformable convolutional networks, 2017. 2

[6] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009. 5

[7] Mingyuan Fan, Shenqi Lai, Junshi Huang, Xiaoming Wei, Zhenhua Chai, Junfeng Luo, and Xiaolin Wei. Rethinking bisenet for real-time semantic segmentation, 2021. 1, 2, 3, 5

[8] Jun Fu, Jing Liu, Haijie Tian, Yong Li, Yongjun Bao, Zhiwei Fang, and Hanqing Lu. Dual attention network for scene segmentation, 2019. 2

[9] Shijie Hao, Yuan Zhou, and Yanrong Guo. A brief survey on semantic segmentation with deep learning. *Neurocomputing*, 406:302–321, 2020. 1, 2

[10] Adam W. Harley, Konstantinos G. Derpanis, and Iasonas Kokkinos. Segmentation-aware convolutional networks using local attention masks, 2017. 2

[11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition, 2015. 2

[12] Judy Hoffman, Eric Tzeng, Taesung Park, Jun-Yan Zhu, Phillip Isola, Kate Saenko, Alexei A. Efros, and Trevor Darrell. Cycada: Cycle-consistent adversarial domain adaptation, 2017. 3

[13] Andrew Howard, Mark Sandler, Grace Chu, Liang-Chieh Chen, Bo Chen, Mingxing Tan, Weijun Wang, Yukun Zhu, Ruoming Pang, Vijay Vasudevan, Quoc V. Le, and Hartwig Adam. Searching for mobilenetv3, 2019. 2

[14] Gao Huang, Zhuang Liu, Laurens van der Maaten, and Kilian Q. Weinberger. Densely connected convolutional networks, 2018. 2

[15] Liulei Li, Tianfei Zhou, Wenguan Wang, Jianwu Li, and Yi Yang. Deep hierarchical semantic segmentation, 2022. 2

[16] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation, 2015. 1

[17] Rudra P K Poudel, Stephan Liwicki, and Roberto Cipolla. Fast-scnn: Fast semantic segmentation network, 2019. 2

[18] Stephan R. Richter, Vibhav Vineet, Stefan Roth, and Vladlen Koltun. Playing for data: Ground truth from computer games, 2016. 2, 5

[19] 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 *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3234–3243, 2016. 2

[20] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition, 2015. 2

[21] Yi-Hsuan Tsai, Wei-Chih Hung, Samuel Schulter, Kihyuk Sohn, Ming-Hsuan Yang, and Manmohan Chandraker. Learning to adapt structured output space for semantic segmentation, 2020. 1, 3, 6

[22] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need, 2023. 2

[23] Yanchao Yang and Stefano Soatto. Fda: Fourier domain adaptation for semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 2, 7

[24] Changqian Yu, Jingbo Wang, Chao Peng, Changxin Gao, Gang Yu, and Nong Sang. Bisenet: Bilateral segmentation network for real-time semantic segmentation, 2018. 1, 2, 3

[25] Xiangyu Zhang, Xinyu Zhou, Mengxiao Lin, and Jian Sun. Shufflenet: An extremely efficient convolutional neural network for mobile devices, 2017. 2

[26] Sicheng Zhao, Xiangyu Yue, Shanghang Zhang, Bo Li, Han Zhao, Bichen Wu, Ravi Krishna, Joseph E. Gonzalez, Alberto L. Sangiovanni-Vincentelli, Sanjit A. Seshia, and Kurt Keutzer. A review of single-source deep unsupervised visual domain adaptation, 2020. 1

[27] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks, 2020. 3