Consistency regularization for unsupervised domain adaptation in semantic segmentation

Sebastian Scherer, Stephan Brehm, Rainer Lienhart

Angaben zur Veröffentlichung / Publication details:

Scherer, Sebastian, Stephan Brehm, and Rainer Lienhart. 2022. "Consistency regularization for unsupervised domain adaptation in semantic segmentation." *Lecture Notes in Computer Science* 13231: 500–511. https://doi.org/10.1007/978-3-031-06427-2_42.

Consistency Regularization for Unsupervised Domain Adaptation in Semantic Segmentation

Sebastian Scherer Stephan Brehm Rainer Lienhart

Machine Learning and Computer Vision Lab, University of Augsburg, Germany
{sebastian1.scherer,stephan.brehm, rainer.lienhart}@uni-a.de

Abstract. Unsupervised domain adaptation is a promising technique for computer vision tasks, especially when annotating large amounts of data is very costly and time-consuming, as in semantic segmentation. Here it is attractive to train neural networks on simulated data and fit them to real data on which the models are to be used. In this paper, we propose a consistency regularization method for domain adaptation in semantic segmentation that combines pseudo-labels and strong perturbations. We analyse the impact of two simple perturbations, dropout and image mixing, and show how they contribute enormously to the final performance. Experiments and ablation studies demonstrate that our simple approach achieves strong results on relevant synthetic-to-real domain adaptation benchmarks.

Keywords: Domain Adaptation · Semi-Supervised Learning · Unsupervised Learning · Semantic Segmentation · Synthetic Data.

1 Introduction

Semantic segmentation has accomplished amazing performance on annotated data and has become one of the most important tasks in computer vision. However, labelling data for semantic segmentation requires assigning a class label to each pixel in an image, which is an extremely tedious and expensive task. For example, annotating a single image of the Cityscapes dataset [6], which consists of images of urban scenes, takes up to 90 minutes [6]. As a result, datasets for semantic segmentation of urban scenes are generally much smaller than datasets for image classification. Synthetic images from computer games are a powerful alternative to real images because they can be labelled automatically, since the geometric 3D scene and the objects it contains, that are projected into the image, are known. This results in high-resolution datasets with precise object boundaries that are inexpensive to obtain and offer almost infinite possibilities for the automatic creation of synthetic data. The problem here is, however, that computer simulations are not perfectly realistic. In general, convolutional neural networks (CNNs) learn features only from the domain on which they were trained. For this reason, CNNs trained on synthetic data tend to perform poorly on real images, even when the synthetic data consists of significantly more images. For example, in our experiments we noticed that as few as 30 images from

the Cityscapes dataset were enough to get similar performance to 24000 images from a dataset that consists of images from the GTA5 game [16].
Unsupervised domain adaptation (UDA) tries to bridge this domain gap. It aims to transfer the knowledge of a label-rich source domain to an unlabelled target domain with similar class information. When synthetic data is used as source domain, the problem of *synthetic-to-real* domain adaptation arises. Recently, researchers utilised self-training (ST) techniques for UDA [1, 5, 15, 25, 30], that allows the usage of images from the target domain directly for the training of the segmentation network via pseudo-labels. They do so by adding a new loss term to the training objective that encourage the model to make consistent prediction of images from the target domain under different perturbations of the image. These approaches achieve great results and represent the current state of the art. In such a framework, the used perturbations are the key for the success of those approaches [9]. Current approaches however use perturbations on the image level, that are sometimes not realistic, such as heavy noise [30], Fourier Mixing [15, 28] or Style Mixing [15, 20]. Even though these perturbations are non-realistic, they improve the general performance when applied. We argue that this improvement also comes from the fact, that the perturbations lead to a much worse prediction, making the pseudo labels, which may not be perfectly correct as well, still better and therefore leading to an improvement of the model. In our experiments, we found that perturbations that do not degrade the prediction of the model do not improve the UDA task, while perturbations that degrade the prediction of the model do. This comes close to the bootstrapping idea in its idiomatic meaning, which refers to a self-starting process that is supposed to continue or improve itself without external input. Based on this situation, the question arises as to how the prediction of the model can be deteriorated the most. While perturbations on the image level are effective, we found that perturbation within the network itself are powerful as well. Surprisingly, we found that a simple baseline model, which uses heavily dropout as perturbation, achieves strong results on current benchmarks for UDA. Combined with a perturbation on the image, we achieve state-of-the-art results. As image perturbation, we utilise a recent perturbation that has originally been proposed for the image classification task: the CowMask [8] image mixing method. It mixes two images and their predictions by a network with a specific mask looking similar to the typical black and white skin pattern of a cow. We argue that this perturbation is perfect for segmentation tasks, as it simulates the occlusion of objects and introduces an additional segmentation task.

2 Related Work

UDA for semantic segmentation has been extensively studied in the last years. Adversarial training was the previously dominant approach applied either on the input space, the feature space or the output space [23] of a segmentation network. Popular input space adaptation techniques try to change the style of the source domain by performing image-to-image translation, for example by

making synthetic images look more realistic [2, 5, 11, 19]. The biggest disadvantage of adversarial training is the unstable training behaviour. Recently, a new line of methods introduced semi-supervised learning (SSL) techniques for UDA and showed remarkable results. SSL aims to include unlabelled data alongside labelled data in the training of a neural network. These approaches are either based on consistency regularization [5,15,30] or self-training [31,32]. Self-training aims to generate pseudo-labels for the unlabelled data and fine-tune the model on them iteratively [13, 27]. Zou et al. [31] and Li et al. [14] applied the pseudo-labelling approach for UDA task and achieved strong improvements. The key idea of consistency regularization is that the predictions of a model should be invariant under different perturbations. These approaches usually adopt a teacher - student framework, where the teacher model is an exponential moving average (EMA) of the student model. The teacher model transfers the learned knowledge to the student, who is additionally influenced by perturbations that are normally applied to the input image. In comparison to self-training, these approaches are typically not trained iteratively, but end-to-end. Choi et al. [5] combined this approach with a GAN-based augmentation module for image translation. Zhou et al. [30] further incorporated an additional uncertainty module that tries to approximate the uncertainty of the predictions to filter uncertain pixel predictions from the loss calculation. To do so, they perform several forward passes of an image with different Gaussian Noise applied and calculate the pixel-wise entropy based on those predictions. Melas-Kyriazi and Manrai [15] proposed a similar approach with different perturbations on the image, namely simple data augmentation, CutMix [29], style consistency and Fourier consistency. Compared to the other work, they did not apply the exponential moving average of the trained model and use the prediction of the model itself as guidance. Our work is mostly related to this line of research. Building on PixMatch [15], we show that simply by applying dropout as perturbation for the student model, we are able to improve the general results. Combined with the recent image-mixing technique using CowMask [8], we achieve strong results with a very simple and easy to implement approach. Specially, we show that the used perturbations are the key for this kind of approaches.

3 Methodology

Let $\mathcal{S}$, $\mathcal{T}$ be the source and target domain and let $X_\mathcal{S}$, $X_\mathcal{T}$ be sets of images from each domain, respectively. We denote $\mathbf{x}_s \in X_\mathcal{S}$ and $\mathbf{x}_t \in X_\mathcal{T}$ as data samples from the source and target domain. At the source domain we have access to N labelled segmentation masks, i.e., $X_\mathcal{S} = \{(\mathbf{x}_s^i, \mathbf{y}_s^i)\}_{i=1}^N$. We denote $\mathbf{y}$ as ground truth annotation from the dataset and $\hat{\mathbf{y}}$ as pseudo-label. The target domain has no labelled samples and shares C categories of the source domain. Our task is to train a segmentation network that performs well on the target domain. This problem formulation can further be extended to multiple source domains $X_\mathcal{S}^1, X_\mathcal{S}^2, ..., X_\mathcal{S}^K$ or target domains $X_\mathcal{T}^1, X_\mathcal{T}^2, ..., X_\mathcal{T}^M$, with K and M the number of source or target domains, respectively.

Figure 1 shows an overview of our proposed architecture. The overall objective function for the segmentation training is defined by a supervised part based on images from the source domain as well as a self-supervised part based on images from the target domain. The self-supervised part employs consistency regularization and pseudo-labels on the target domain. Similar to Tarvainen et al. [22], we make use of two networks of identical architecture: a student network F_S and a teacher network F_T. The predictions of the teacher network are used to produce pseudo-labels for images of the target domain, which are subsequently used to train the student network. In this framework, the teacher network is simply the exponential-moving average (EMA) of the student model and will not receive any gradient-based parameter updates. The overall objective $\mathcal{L}^{F_S}$ is defined as follows:

$$\mathcal{L}^{F_S} = \mathcal{L}_{\mathcal{S}}^{F_S} + \lambda_{\mathcal{T}} \mathcal{L}_{\mathcal{T}}^{F_S}, \tag{1}$$

where $\lambda_{\mathcal{T}}$ is a trade-off parameter. $\mathcal{L}_{\mathcal{S}}^{F_S}$ indicates the softmax cross-entropy objective $\mathcal{L}_{ce}$ between the prediction of the student network $F_S(\mathbf{x}_s)$ for an image of the source domain $\mathbf{x}_s \in X_{\mathcal{S}}$ and its pixel-level annotation map $\mathbf{y}_s$, i.e., $\mathcal{L}_{\mathcal{S}}^{F_S} = \mathcal{L}_{ce}(F_S(\mathbf{x}_s), \mathbf{y}_s)$.

Given an image from the target domain $\mathbf{x}_t \in X_{\mathcal{T}}$ and its perturbed version $\bar{\mathbf{x}}_t$, we feed the image through the teacher network F_T to obtain the soft pseudo-label $\hat{\mathbf{y}}_t^{soft} = F_T(\mathbf{x}_t)$. We can get a hard pseudo-label by calculating the argmax at the class dimension, i.e. $\hat{\mathbf{y}}_t^{hard} = argmax(\hat{\mathbf{y}}_t^{soft})$. Both soft and hard pseudo labels may now be used as targets for the student network. By incorporating $\hat{\mathbf{y}}_t^{soft}$, we can calculate a loss by applying the Mean Squared Error (MSE):

$$\mathcal{L}_{\mathcal{T}}^{F_S} = \mathcal{L}_{MSE}(F_S(\bar{\mathbf{x}}_t), \hat{\mathbf{y}}_t^{soft}). \tag{2}$$

By incorporating $\hat{\mathbf{y}}_t^{hard}$, we can calculate a loss by applying the standard cross entropy loss:

$$\mathcal{L}_{\mathcal{T}}^{F_S} = \mathcal{L}_{ce}(F_S(\bar{\mathbf{x}}_t), \hat{\mathbf{y}}_t^{hard}). \tag{3}$$

Comparing soft and hard pseudo-labels, soft pseudo-labels are generally more robust against noisy labels (false classifications), while hard pseudo-labels bring an additional learning effect as the highest activation is reinforced across classes. We will investigate both in our ablation study.

Perturbations As stated in [9], the success of SSL techniques based on consistency regularization depends on the used perturbations. These are used only at the forward pass of the student model when calculating the self-supervised loss on images from the target domain. We experiment with two different perturbations, one at the input image and one within the model itself.

An easy way to perform a perturbation on the student side is by using dropout layers. The role of dropout is to improve generalization performance by preventing the model from overfitting. It forces the network to learn more robust features that can deal with many random subsets of neurons. In addition, dropout usually also leads to a deterioration of the prediction and thus to an increase of the error, since, in a sense, only a part of the original network is used.

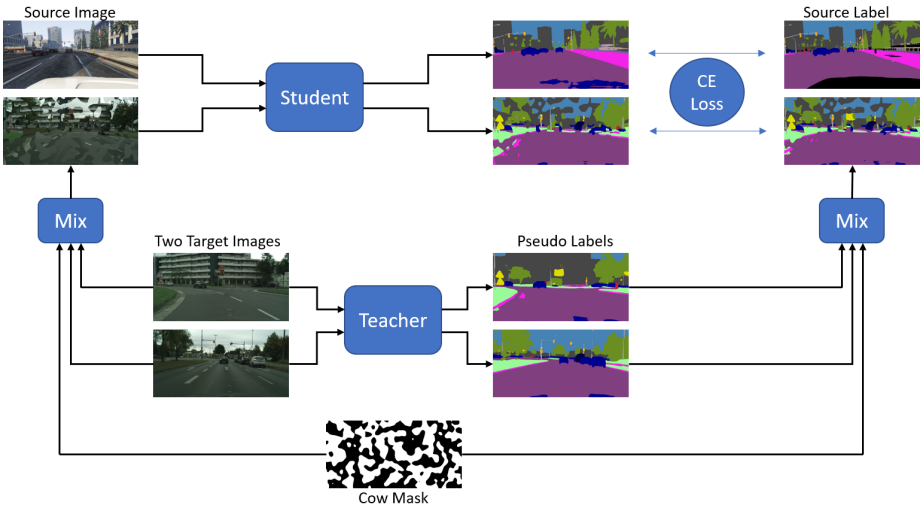


Fig. 1: Illustration of the proposed solution.

As we work with CNNs, we will make usage of SpatialDropout as proposed by Tompson et al. [24]. Given a feature tensor from a convolutional layer of size $height \times width \times depth$, where $depth$ is the amount of filters of the layer, the dropout is applied at some dimensions of the depth across the entire feature map. It therefore simulates that certain filters had no activation. The dropout within the network has the positive effect, that the resulting sub-network of the original network will be trained on clean images, with lets the network learn particular features of that domain. Other perturbations such as heavy random noise or Style change [15, 20] applications let the network learn on images that will not occur in reality.

Following [8], we further use the Cutmix [29] augmentation using a CowMask [8] as perturbation. In this augmentation, individual parts of the image will be replaced by another image. These augmentations do not occur in reality, but they serve as an additional perturbation by suppressing areas and introducing additional object boundaries at the edge of the applied mask. The replacement of certain areas within the image further results in an occlusion of the objects. To generate a single sample for the student network, we take two images from the target domain $\mathbf{x}_t^1$ and $\mathbf{x}_t^2$. Both images will be fed to the teacher network F_T to obtain the pseudo labels $\hat{\mathbf{y}}_t^1$ and $\hat{\mathbf{y}}_t^2$. We then calculate the image and target for the training of the student network by mixing the images and the pseudo labels according to the generated mask M by:

$$\begin{aligned} \bar{\mathbf{x}}_t &= M \odot \mathbf{x}_t^1 + (1 - M) \odot \mathbf{x}_t^2, \\ \hat{y}_t &= M \odot \hat{\mathbf{y}}_t^1 + (1 - M) \odot \hat{\mathbf{y}}_t^2, \end{aligned} \tag{4}$$

where $M \in \{0, 1\}^{W \times H}$ denotes a binary mask and $\odot$ is element-wise multiplication. For the calculation of a random CowMask M, please refer to [8].

4 Experiments and Results

In this section, we detail the experiments that we conducted in order to show the benefit and performance of our proposed approach.

4.1 Datasets

We use five datasets in our experiments. The Cityscapes dataset [6] contains images from real-world urban scenes, split into 2975 images for training and 500 for validation. The GTA5 dataset [16] and the SYNTHIA dataset [17] contain 24966 and 9400 synthetic images with pixel wise annotations, respectively. The annotations of both datasets are compatible with Cityscapes. Both synthetic datasets serve as source domain, while the Cityscapes dataset serves as target domain. This results in two popular *synthetic-to-real* domain adaptation scenarios: GTA5 to Cityscapes (GTA $\rightarrow$ CS) and SYNTHIA to Cityscapes (SYN $\rightarrow$ CS). We further experiment with a third synthetic dataset Synscapes [26], which contains 25000 photo-realistic images. Besides *synthetic-to-real* domain adaptation, we also evaluate our approach at the CS $\rightarrow$ ACDC benchmark. The ACDC [18] dataset contains images of four common adverse visual conditions: fog, nighttime, rain and snow. Images from the Cityscapes dataset are taken at normal weather conditions and at daytime. This domain adaptation attempts to improve the segmentation model at different visual conditions as they occur in the labelled dataset. As it considers four different visual conditions, it can be seen as a multi-target domain adaptation problem. Each visual condition contains of 400 training images, 100 validation images and 500 test images.

4.2 Implementation details

For a fair comparison to earlier works, we adopt the VGG16 [21] and the ResNet101 [10] backbone pre-trained on the ImageNet dataset [7]. Following Deeplab-V2 [3], we incorporate Atrous Spatial Pyramid Pooling (ASPP) as the decoder and then use bilinear upsampling to get the segmentation output. We use color jittering as augmentation on images from the source domain with the same settings as used in [4]. For the dropout perturbation, we place a dropout layer before each pooling or strided convolutional layer for simple reproducibility and re-implementation. If not stated otherwise, we use a dropout rate of 0.3 and an EMA value of 0.999. For the ResNet101, the Batch Normalization layers are frozen during training. We set λ_{real} to 50 when using soft pseudo-labels, otherwise to 1. All experiments were conducted on a single NVIDIA V100 GPU with 16GB of VRAM. We perform 25.000 training steps where no loss is calculated on the target domain as warm-up phase. Afterwards, each mini-batch consist of an image from the source and target domain respectively. We train our models with early stopping.

Table 1: Ablation Study of our proposed perturbations within the consistency regularization framework using the VGG16 network as backbone. 'SL' and 'HL' stands for soft and hard pseudo-labels, respectively. 'CM' stands for the image-mixing technique using a CowMask.

Method	mIoU^{19}
Baseline	35.23
SL	40.57
SL + Dropout	49.03
SL + CM	51.29
SL + Dropout + CM	49.81
HL	40.05
HL + Dropout	48.0
HL + CM	52.26
HL + Dropout + CM	**53.4**

(a) GTA5 → Cityscapes

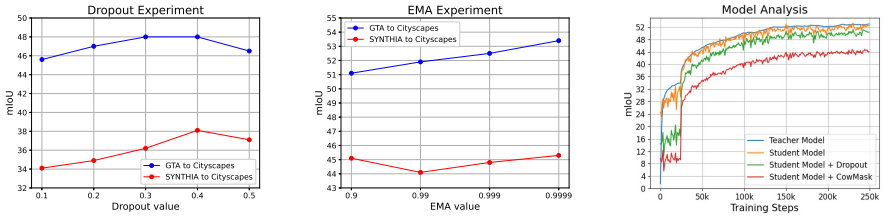
Method	mIoU^{16}	mIoU^{13}
Baseline	31.0	35.9
SL	32.05	36.86
SL + Dropout	37.6	42.90
SL + CM	38.74	43.85
SL + Dropout + CM	41.47	47.34
HL	33.17	38.01
HL + Dropout	38.12	43.53
HL + CM	41.82	47.50
HL + Dropout + CM	**44.74**	**51.21**

(b) SYNTHIA → Cityscapes

4.3 Ablation Study

In this section, we study the effectiveness of each component in our approach and investigate how they contribute to the final performance on both benchmarks when using the VGG16 as backbone. Table 1 compares the use of soft and hard pseudo-labels (SL vs. HL) as well as the perturbations alone and in combination. Comparing the results for soft and hard pseudo-labels, we can identify only minor differences. However, when both perturbations are used, hard pseudo-labels lead to better results. This is presumably because when both perturbations are used, the model improves and so do the pseudo labels, giving the additional learning effect of the pseudo labels a better impact. Comparing the approach without any perturbation to the baseline, we can observe an improvement of 5% on the GTA → CS and 1% on the SYN → CS benchmark. Using only dropout as perturbations improves the results for $8-9\%$ on the GTA → CS and 5% on the SYN → CS benchmark. The same applies for the CowMask image mixing perturbation, which improves the result even more. Thus, the use of dropout or image mixing as perturbations leads to a stronger improvement than the use of pseudo-labels in general. Combining both perturbations gives a slight improvement in comparison to both perturbations alone. In general for the GTA → CS benchmark, we improve the result by 17% compared to the baseline, while 12% come alone from the used perturbations. This shows that the used perturbations are indeed the key of the success of such methods.

We also study the effect of different dropout and EMA score rates in Figure 2. For the dropout experiment, the image mixing perturbation is not used, but it is for the EMA experiment. It can be seen, that different EMA values hardly have an effect on the final performance. Higher values achieve a slightly better result. It should also be noted that higher EMA values lead to more stable training and the results are much better reproducible. For the dropout values,



(a) (b) (c)

Fig. 2: (a), (b) Ablation study for different dropout and EMA decay values for the GTA → CS and the SYNTHIA → CS benchmark. (c) Performance of the teacher model, the student model and the student model with the proposed perturbations during training. The experiments were performed with a VGG16 as backbone. Best viewed in colour.

the performance increases up to 0.4, then decreases again slightly. This may be due to the fact that at very high dropout values too much information is lost.

To explain the effect of perturbations on the final performance, we conducted an experiment where we continuously evaluate the teacher model and the student model with and without our proposed perturbations. The result is shown in Figure 2c. It can be seen, that the teacher model is always better as the student model impaired by perturbations. That means, that the feedback the student gets from the teacher during training is always on average better as its own predictions. As it receives a positive feedback, the model may be able to continuously improve itself and learn better decision boundaries.

4.4 Comparisons to state-of-the-art

For a fair comparison, we compare our method on both benchmarks with similar methods from the last two years that are primary based on consistency regularization for SSL in any kind. Note that not all published methods report results for the VGG16 and the ResNet101 backbone. We report results using hard pseudo-labels and both perturbations. The results are shown in Table 2 and Table 3. For the VGG backbone, we compare our method with two approaches that combine SSL with additional image-to-image translation [5, 30]. For the ResNet101 backbone, we compare our method with four state-of-the art methods that utilize SSL [1, 12, 15, 25]. We can see that our much simpler approach achieves a substantial improvement on both benchmarks and achieves the best results for the GTA → CS benchmark with both backbones. At the SYN → CS benchmark we achieve the second-best results, while SAD [1] performs best. However, we believe that this comes mainly from their additional class balanced training and importance sampling, where they increase the sample frequency of certain classes during training, as they reported a drop from 49.9% to 44.5% at the GTA → CS benchmark when not using it. Specially importance sampling of

Table 2: Results on the GTA5 → Cityscapes benchmark. We compare our method using the VGG16 (A) and the ResNet-101 (B) backbone.

GTA5 → Cityscapes																					
	Model	road	side.	buil.	wall	fence	pole	t-light	t-sign	vege.	terr.	sky	pers.	rider	car	truck	bus	train	motor	bike	**mIoU**
Baseline	A	88.0	39.8	79.3	27.5	7.4	26.0	25.9	11.4	81.3	24.6	67.0	52.0	12.8	81.5	20.8	12.2	0.0	8.3	0.7	35.2
Choi et al. [5]	A	90.2	51.5	81.1	15.0	10.7	37.5	35.2	28.9	84.1	32.7	75.9	62.7	19.9	82.6	22.9	28.3	0.0	23.0	25.4	42.5
Zhou et al. [30]	A	95.1	66.5	84.7	35.1	19.8	31.2	35.0	32.1	86.2	43.4	82.5	61.0	25.1	87.1	35.3	46.1	0.0	24.6	17.5	47.8
SAD [1] (w/o CBT-IS-FL)	A	88.1	41.0	85.7	30.8	30.6	33.1	37.0	22.9	86.6	36.8	90.7	67.1	27.1	86.8	34.4	30.4	8.5	7.5	0.0	44.5
SAD [1]	A	90.0	53.1	86.2	33.8	32.7	38.2	46.0	40.3	84.2	26.4	88.4	65.8	28.0	85.6	40.6	52.9	17.3	13.7	23.8	49.9
Ours	A	93.6	58.7	88.4	41.3	40.6	33.9	47.4	59.5	85.0	37.4	86.0	57.7	33.9	86.7	38.7	53.5	24.9	42.7	4.1	**53.4**
Baseline	B	89.1	41.0	81.9	31.0	5.3	28.7	27.9	14.4	82.3	28.9	84.9	51.7	12.5	81.5	21.6	15.9	0.0	5.1	0.2	37.0
MLSL [12]	B	89.0	45.2	78.2	22.9	27.3	37.4	46.1	43.8	82.9	18.6	61.2	60.4	26.7	85.4	35.9	44.9	36.4	37.2	49.3	49.0
PixMatch [15]	B	91.6	51.2	84.7	37.3	29.1	24.6	31.3	37.2	86.5	44.3	85.3	62.8	22.6	87.6	38.9	52.3	0.65	37.2	50.0	50.3
DACS [25]	B	89.9	39.7	87.9	30.7	39.5	38.5	46.4	52.8	88.0	44.0	88.8	67.2	35.8	84.5	45.7	50.2	0.0	27.3	34.0	52.1
SAD [1]	B	90.4	53.9	86.6	42.4	27.3	45.1	48.5	42.7	87.4	40.1	86.1	67.5	29.7	88.5	49.1	54.6	9.8	26.6	45.3	53.8
Ours	B	95.1	65.1	88.3	46.6	28.2	36.5	44.6	49.0	86.9	42.0	89.0	64.1	30.9	89.7	53.0	64.6	0.0	25.4	49.0	**55.2**

rare classes is a typical approach of semantic segmentation applications in general and not specific to UDA. Compared to PixMatch [15] that is mostly related to our work, we observe a substantial improvement on both benchmarks.

We further evaluate our proposed solution for multi-source or multi-target domain adaptation problems. At these experiments, we simply merge the different domains or datasets into one. Table 4a shows that the performance increases when we simply combine different synthetic datasets. Combining GTA5 and SYNTHIA, we can improve the performance from 55.2% using GTA5 only to 59.9%. Including Synscapes as well, we can achieve a performance of 63.3%. This shows that using multiple different synthetic datasets has a positive effect on the performance, and that by exploring and combining more advanced synthetic datasets, we could achieve similar results to a model trained fully supervised on Cityscapes. Table 4b shows the result for the CS → ACDC benchmark, where our proposed method is able to improve the performance on different visual conditions. It should also be mentioned that we did not observe any deterioration in the performance of the adapted model at the Cityscapes validation set. In this context, the *Source-only* model is trained fully supervised on Cityscapes, while the *Oracle* model used the labelled training set of Cityscapes and the ACDC dataset as supervision. Compared to the *Source-only* model, we can improve the performance relative to the *Oracle* model by nearly 55%. Since the ACDC dataset contains only 400 images per visual condition at the training set, we believe that a larger unlabelled dataset can further improve the results, making the annotation process unnecessary in this case. The ability of SSL to use unlabelled data is not exploited in this experiment, as both our UDA trained model as well as the *Oracle* see the same images during training.

Table 3: Results on the SYNTHIA → Cityscapes benchmark. We compare our method using the VGG16 (A) and the ResNet-101 (B) backbone.

SYNTHIA → Cityscapes																			
	Model	road	side.	buil.	wall*	fence*	pole*	t-light	t-sign	vege.	sky	pers.	rider	car	bus	motor	bike	**mIoU^{16}**	**mIoU^{13}**
Baseline	A	44.8	19.6	64.7	3.1	0.1	26.0	4.8	12.7	75.9	75.5	46.4	12.3	65.1	15.7	10.0	18.5	31.0	35.9
Choi et al. [5]	A	90.1	48.6	80.7	2.2	0.2	27.2	3.2	14.3	82.1	78.4	54.4	16.4	82.5	12.3	1.7	21.8	38.5	46.6
Zhou et al. [30]	A	93.1	53.2	81.1	2.6	0.6	29.1	7.8	15.7	81.7	81.6	53.6	20.1	82.7	22.9	7.7	31.3	41.5	48.6
SAD [1]	A	77.9	38.6	83.5	15.8	1.5	38.2	41.3	27.9	80.8	83.0	64.3	21.2	78.3	38.5	32.6	62.1	**49.1**	**56.2**
Ours	A	65.0	25.6	81.9	19.1	0.0	31.1	1.3	40.9	79.1	82.4	61.5	27.9	86.5	61.4	14.6	37.7	44.7	51.2
Baseline	B	52.5	20.6	72.8	3.0	0.0	27.6	0.0	6.8	78.8	78.7	42.7	15.8	67.7	18.5	8.6	18.6	32.1	37.2
MLSL [12]	B	59.2	30.2	68.5	22.9	1.0	36.2	32.7	28.3	86.2	75.4	68.6	27.7	82.7	26.3	24.3	52.7	45.2	51.0
PixMatch [15]	B	92.5	54.6	79.8	4.8	0.1	24.1	22.8	17.8	79.4	76.5	60.8	24.7	85.7	33.5	26.4	54.4	46.1	54.5
DACS [25]	B	80.6	25.1	81.9	21.5	2.9	37.2	22.7	24.0	83.7	90.8	67.6	38.3	82.9	38.9	28.5	47.6	48.3	54.8
SAD [1]	B	89.3	47.2	85.5	26.5	1.3	43.0	45.5	32.0	87.1	89.3	63.6	25.4	86.9	35.6	30.4	53.0	**52.6**	**59.3**
Ours	B	89.0	53.6	85.0	23.7	3.2	34.4	6.0	41.3	82.2	80.6	54.1	39.0	86.2	68.1	25.6	45.1	51.1	58.1

Table 4: Results for multi-source and multi-target domain adaptation.

	Sources	mIoU^{19}
Single Source DA	G	55.2
	S	44.3
	C	55.4
Multi Source DA	G+S	59.9
	G+S+C	63.3

(a) Results on the Cityscapes validation set combining different source domains. G: GTA5, S: SYNTHIA, C:Synscapes.

Method	mIoU^{19}
Source-only	50.6
Ours	59.1
Oracle	66.3

(b) Results on the ACDC test set for the Cityscapes → ACDC benchmark using the ResNet101 backbone.

5 Conclusion

In this work, we investigated the problem of unsupervised domain adaptation for semantic segmentation. To address this problem, we presented the use of an approach for consistency regularization combined with perturbations on the input image as well as the model itself. Through a comprehensive series of ablation studies, we have sought to understand which aspects of this approach are most important to the final performance of the model. We were able to show that the type of perturbation is the key to success. Even a simple perturbation such as dropout is able to improve the performance of the model by a large margin. Combined with an image mixing method, the approach is able to achieve state-of-the-art results. Future work may explore the combination of other existing perturbation functions.

References

1. Araslanov, N., Roth, S.: Self-supervised augmentation consistency for adapting semantic segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 15384–15394 (2021)
2. Brehm, S., Scherer, S., Lienhart, R.: Semantically consistent image-to-image translation for unsupervised domain adaptation. In: 2022 International Conference on Agents and Artificial Intelligence (2022)
3. Chen, L.C., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.L.: Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. IEEE transactions on pattern analysis and machine intelligence **40**(4), 834–848 (2017)
4. Chen, T., Kornblith, S., Norouzi, M., Hinton, G.: A simple framework for contrastive learning of visual representations. In: International conference on machine learning. pp. 1597–1607. PMLR (2020)
5. Choi, J., Kim, T., Kim, C.: Self-ensembling with gan-based data augmentation for domain adaptation in semantic segmentation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 6830–6840 (2019)
6. Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., Schiele, B.: The cityscapes dataset for semantic urban scene understanding. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 3213–3223 (2016)
7. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: 2009 IEEE conference on computer vision and pattern recognition. pp. 248–255. Ieee (2009)
8. French, G., Oliver, A., Salimans, T.: Milking cowmask for semi-supervised image classification. arXiv preprint arXiv:2003.12022 (2020)
9. French, G., Laine, S., Aila, T., Mackiewicz, M., Finlayson, G.: Semi-supervised semantic segmentation needs strong, varied perturbations. In: British Machine Vision Conference (2020)
10. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016)
11. Hoffman, J., Tzeng, E., Park, T., Zhu, J.Y., Isola, P., Saenko, K., Efros, A., Darrell, T.: Cycada: Cycle-consistent adversarial domain adaptation. In: International conference on machine learning. pp. 1989–1998. PMLR (2018)
12. Iqbal, J., Ali, M.: Mlsl: Multi-level self-supervised learning for domain adaptation with spatially independent and semantically consistent labeling. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. pp. 1864–1873 (2020)
13. Lee, D.H.: Pseudo-label : The simple and efficient semi-supervised learning method for deep neural networks. ICML 2013 Workshop : Challenges in Representation Learning (WREPL) (07 2013)
14. Li, Y., Yuan, L., Vasconcelos, N.: Bidirectional learning for domain adaptation of semantic segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 6936–6945 (2019)
15. Melas-Kyriazi, L., Manrai, A.K.: Pixmatch: Unsupervised domain adaptation via pixelwise consistency training. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 12435–12445 (2021)

16. Richter, S.R., Vineet, V., Roth, S., Koltun, V.: Playing for data: Ground truth from computer games. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) European Conference on Computer Vision (ECCV). LNCS, vol. 9906, pp. 102–118. Springer International Publishing (2016)
17. Ros, G., Sellart, L., Materzynska, J., Vazquez, D., Lopez, A.M.: The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 3234–3243 (2016)
18. Sakaridis, C., Dai, D., Van Gool, L.: Acdc: The adverse conditions dataset with correspondences for semantic driving scene understanding. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 10765–10775 (2021)
19. Scherer, S., Schön, R., Ludwig, K., Lienhart, R.: Unsupervised domain extension for nighttime semantic segmentation in urban scenes. In: 2021 International Conference on Deep Learning Theory and Applications (2021)
20. Sheng, L., Lin, Z., Shao, J., Wang, X.: Avatar-net: Multi-scale zero-shot style transfer by feature decoration. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 8242–8250 (2018)
21. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)
22. Tarvainen, A., Valpola, H.: Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. arXiv preprint arXiv:1703.01780 (2017)
23. Toldo, M., Maracani, A., Michieli, U., Zanuttigh, P.: Unsupervised domain adaptation in semantic segmentation: a review. Technologies **8**(2), 35 (2020)
24. Tompson, J., Goroshin, R., Jain, A., LeCun, Y., Bregler, C.: Efficient object localization using convolutional networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 648–656 (2015)
25. Tranheden, W., Olsson, V., Pinto, J., Svensson, L.: Dacs: Domain adaptation via cross-domain mixed sampling. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. pp. 1379–1389 (2021)
26. Wrenninge, M., Unger, J.: Synscapes: A photorealistic synthetic dataset for street scene parsing. arXiv preprint arXiv:1810.08705 (2018)
27. Xie, Q., Hovy, E.H., Luong, M., Le, Q.V.: Self-training with noisy student improves imagenet classification. CoRR **abs/1911.04252** (2019), http://arxiv.org/abs/1911.04252
28. Yang, Y., Soatto, S.: Fda: Fourier domain adaptation for semantic segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4085–4095 (2020)
29. Yun, S., Han, D., Oh, S.J., Chun, S., Choe, J., Yoo, Y.: Cutmix: Regularization strategy to train strong classifiers with localizable features. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 6023–6032 (2019)
30. Zhou, Q., Feng, Z., Cheng, G., Tan, X., Shi, J., Ma, L.: Uncertainty-aware consistency regularization for cross-domain semantic segmentation. arXiv preprint arXiv:2004.08878 (2020)
31. Zou, Y., Yu, Z., Kumar, B., Wang, J.: Unsupervised domain adaptation for semantic segmentation via class-balanced self-training. In: Proceedings of the European conference on computer vision (ECCV). pp. 289–305 (2018)
32. Zou, Y., Yu, Z., Liu, X., Kumar, B., Wang, J.: Confidence regularized self-training. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 5982–5991 (2019)