Unsupervised Domain Extension for Nighttime Semantic Segmentation in Urban Scenes

Sebastian Scherer*, Robin Schön*, Katja Ludwig and Rainer Lienhart

University of Augsburg, Universitatsstraße 1, 86159 Augsburg, Germany

Keywords: Self-supervised Learning, Image-to-Image Translation, Semantic Segmentation, Nighttime.

Abstract: This paper deals with the problem of semantic image segmentation of street scenes at night, as the recent advances in semantic image segmentation are mainly related to daytime images. We propose a method to extend the learned domain of daytime images to nighttime images based on an extended version of the CycleGAN framework and its integration into a self-supervised learning framework. The aim of the method is to reduce the cost of human annotation of night images by robustly transferring images from day to night and training the segmentation network to make consistent predictions in both domains, allowing the usage of completely unlabelled images in training. Experiments show that our approach significantly improves the performance on nighttime images while keeping the performance on daytime images stable. Furthermore, our method can be applied to many other problem formulations and is not specifically designed for semantic segmentation.

1 INTRODUCTION

Semantic segmentation has accomplished amazing performance on annotated data. However, the datasets are limited to some cities and also often do not include scenes that are outside the usual distribution, such as situations with snow, rain or images taken at night. If the trained neural network is guided into such a new environment, it is of great interest that it also works there without having to create a new dataset. In general, one wants to adjust the training of the neural network so that it is able to respond to the new domain without degrading to the known domain. We formulate this problem as *Unsupervised Domain Extension*, i.e. given a dataset on a particular domain, our goal is to make the neural network robust to other known domains for the problem under consideration, without annotated data being available. To tackle this problem, this paper proposes a method for semantic segmentation that combines a semantic consistent image-to-image translation framework with a self-supervised learning framework. Given a source dataset with labels and a dataset from a new domain without labels, the goal is to adapt the network so that it performs similarly in both domains while training only with the labels of on one domain. This problem formulation is quite similar to the domain adaption formulation, where a network is usually trained on a synthetic dataset and shall perform in the real world. The difference is here, that we still care about the source domain.

In this work, we consider this problem for the case of semantic segmentation on daytime and nighttime images. More specific, we want to achieve that while we only have annotated data from the daytime domain, we also become more robust to nighttime images for which we do not have labelled data.

One of the biggest obstacles for a segmentation network trained only with annotated data showing a landscape during the day is the visual difference from the landscape at night. To mitigate this problem, we transform the annotated daytime images to incorporate the visual details of the nighttime images without changing the content, so that we can continue to use the existing annotations. A possible tool for such a transformation is provided by the CycleGAN framework (Zhu et al., 2017). However, this technique is prone to hallucinate semantic inconsistencies into the transformed image. Even the smallest hallucinations can cause the annotations to be incorrect and the network to receive negative, noisy feedback. In order to suppress such inconsistencies, we will use the feature loss from style transfer (Gatys et al., 2015) provided by a pretrained network to keep the content of the image.

In addition to such a transformations, we will in-

*indicates equal contribution.

Scherer, S., Schön, R., Ludwig, K. and Lienhart, R.
Unsupervised Domain Extension for Nighttime Semantic Segmentation in Urban Scenes.
DOI: 10.5220/0010551500380047
In *Proceedings of the 2nd International Conference on Deep Learning Theory and Applications (DeLTA 2021)*, pages 38-47
ISBN: 978-989-758-526-5

centivize additional robustness when training the segmentation network itself. Inspired by (Tarvainen and Valpola, 2017), we will use a student network and a teacher network of identical architecture with the teachers parameters being a exponential moving average of the student parameters. From this, we formulate two consistency losses that motivate the network to respond to night images consistently, with the goal of learning domain invariant features related to day and night.

This paper will be structured as follows. Section 2 will show previous related work and explain the basic functionality of the CycleGAN framework. Section 3 gives a detailed explanation of our proposed solution. Section 4 provides results of our experiments and section 5 contains our conclusion.

2 RELATED WORKS

Self-supervised Learning. The objective of self-supervised learning (SSL) is to include unlabelled data into the training and therefore perform better than a supervised learning technique using labelled data only. The dominant approaches for SSL are pseudo-labelling and consistency regularization. A complete review can be found in this survey (van Engelen and Hoos, 2020).

Pseudo-Labeling is probably the simplest approach for SSL. It was first proposed by (Lee, 2013). (Xie et al., 2019) recently showed that pseudo labels can indeed improve overall performance. In their approach, a model is first trained on the labelled dataset until convergence. It is then used to make predictions for the unlabelled data, so-called pseudo labels. From these pseudo labels, the samples for which the prediction is certain (above a predefined threshold) are added to the pseudo label dataset and a new model is then trained on the extended labelled dataset.

Recent SSL methods are based on consistency regularization. They employ unlabelled data to produce consistent predictions under different perturbations (Tarvainen and Valpola, 2017). Possible perturbations can be data augmentation, dropout (Srivastava et al., 2014) or simple noise on the input data. The trained model should be robust against such perturbations. Therefore, these approaches leverage the idea that a classifier should output the same distribution for different augmented versions of an unlabelled sample. This is typically achieved by minimizing the difference between the prediction of a model with weights θ of different perturbed versions $\hat{x}_1^i, \hat{x}_2^i$ of an input x^i:

$$||(f(\theta,\hat{x}_1^i), f(\theta,\hat{x}_2^i)||_2^2. \qquad (1)$$

Such a loss can be calculated on labelled and unlabelled data. The recent approach *Mean Teacher* further uses ensemble predictions during training, because an ensemble model generally gives better predictions compared to a single model (Tarvainen and Valpola, 2017). Instead of comparing model predictions for different versions of the image directly, the predictions of the trained model are compared with the predictions of a weighted average model from the previous epochs. One term of Eq. 1 is therefore replaced with the output of the ensemble model. In this setup, the teacher model is an exponential moving average (EMA) of the student model and is intended to transfer the learned knowledge to the student. Typically, a Mean Square Error is used to ensure consistency (Tarvainen and Valpola, 2017). Recently, the usage of the Mixup (Zhang et al., 2018) augmentation as perturbation has been used by (Verma et al., 2019) and showed state-of-the art results, showing that different perturbations yield different results.

Image-to-Image Translation. During the last years, multiple methods provide the possibility to transform an image from a certain source domain $\mathcal{S}$ to adopt the style of a target domain $\mathcal{T}$ without changing the content of the original image. Publications like (Zhu et al., 2017), (Liu et al., 2017) and (Huang et al., 2018) try to solve this problem without having paired image data at hand (no exact correspondences between images) by employing a pixel-wise reconstruction constraint. Other publications, such as (Johnson et al., 2016), (Atapattu and Rekabdar, 2019), (Zhao et al., 2020) and (Nizan and Tal, 2020), try to avoid the utilization of the pixel-wise reconstruction constraint.

One of the most well-known methods to do so is the CycleGAN framework, originating from (Zhu et al., 2017). This method provides a way of transforming images from domain $\mathcal{S}$ to domain $\mathcal{T}$ and backwards by training two image transformation networks $G_{\mathcal{S}\to\mathcal{T}}$ and $G_{\mathcal{T}\to\mathcal{S}}$ simultaneously. The adoption of the other domains style will be enforced by an adversarial training scheme with discriminators $D_{\mathcal{T}}$ and $D_{\mathcal{S}}$, which learn to distinguish between real and generated images from the domains $\mathcal{T}$ and $\mathcal{S}$, respectively.

First, we draw one pair $(\mathbf{x}_s, \mathbf{x}_t) \in X_{\mathcal{S}} \times X_{\mathcal{T}}$ of training images. We assume that $X_{\mathcal{S}}$ and $X_{\mathcal{T}}$ constitute an adequate representation of the data distributions $p_{\text{data}}(\mathbf{x}_s)$ and $p_{\text{data}}(\mathbf{x}_t)$. The two images are fed to the transformation networks to obtain the transformed images

$$\hat{\mathbf{x}}_s = G_{\mathcal{S}\to\mathcal{T}}(\mathbf{x}_s) \text{ and} \tag{2}$$
$$\hat{\mathbf{x}}_t = G_{\mathcal{T}\to\mathcal{S}}(\mathbf{x}_t). \tag{3}$$

These two images are going to be judged by the respective discriminators. The transformation networks have the role of the generators, being forced to create images in a style that the discriminators find increasingly appealing. The adversarial loss for $G_{\mathcal{S}\to\mathcal{T}}$ and $D_{\mathcal{T}}$ can be formulated as

$$\begin{aligned}\mathcal{L}_{\text{GAN}}(G_{\mathcal{S}\to\mathcal{T}}, D_{\mathcal{T}}, X_{\mathcal{S}}, X_{\mathcal{T}}) = \mathbb{E}_{\mathbf{x}_t\sim p_{\text{data}}(\mathbf{x}_t)}[\log D_{\mathcal{T}}(\mathbf{x}_t)] \\ +\mathbb{E}_{\mathbf{x}_s\sim p_{\text{data}}(\mathbf{x}_s)}[\log(1 - D_{\mathcal{T}}(G_{\mathcal{S}\to\mathcal{T}}(\mathbf{x}_s))]\end{aligned} \tag{4}$$

The adversarial loss for $G_{\mathcal{T}\to\mathcal{S}}$ and $D_{\mathcal{S}}$ is conceptually equivalent.

The following steps motivate $G_{\mathcal{S}\to\mathcal{T}}$ and $G_{\mathcal{T}\to\mathcal{S}}$ to preserve the semantic content of the image during transformation. The translated images $\hat{\mathbf{x}}_s$ and $\hat{\mathbf{x}}_t$ from Equations 2 and 3, are translated backwards into

$$\mathbf{x}_s^{\text{cyc}} = G_{\mathcal{T}\to\mathcal{S}}(\hat{\mathbf{x}}_s) = G_{\mathcal{T}\to\mathcal{S}}(G_{\mathcal{S}\to\mathcal{T}}(\mathbf{x}_s)) \text{ and} \tag{5}$$
$$\mathbf{x}_t^{\text{cyc}} = G_{\mathcal{S}\to\mathcal{T}}(\hat{\mathbf{x}}_t) = G_{\mathcal{S}\to\mathcal{T}}(G_{\mathcal{T}\to\mathcal{S}}(\mathbf{x}_t)) \tag{6}$$

with the aim of reconstructing the original image. The corresponding loss will be provided by the L1-distance between the original and its cyclical reconstruction:

$$\begin{aligned}\mathcal{L}_{\text{cyc}}(G_{\mathcal{S}\to\mathcal{T}}, G_{\mathcal{T}\to\mathcal{S}}) = \\ \mathbb{E}_{\mathbf{x}_s\sim p_{\text{data}}(\mathbf{x}_s)}[||\mathbf{x}_s - G_{\mathcal{T}\to\mathcal{S}}(G_{\mathcal{S}\to\mathcal{T}}(\mathbf{x}_s))||_1] \\ +\mathbb{E}_{\mathbf{x}_t\sim p_{\text{data}}(\mathbf{x}_t)}[||\mathbf{x}_t - G_{\mathcal{S}\to\mathcal{T}}(G_{\mathcal{T}\to\mathcal{S}}(\mathbf{x}_t))||_1]\end{aligned} \tag{7}$$

The overall optimization goal is then described by the loss function

$$\begin{aligned}\mathcal{L}(G_{\mathcal{S}\to\mathcal{T}}, G_{\mathcal{T}\to\mathcal{S}}, D_{\mathcal{S}}, D_{\mathcal{T}}, X_{\mathcal{S}}, X_{\mathcal{T}}) = \\ \lambda_{\text{cyc}}\mathcal{L}_{\text{cyc}}(G_{\mathcal{S}\to\mathcal{T}}, G_{\mathcal{T}\to\mathcal{S}}) \\ +\mathcal{L}_{\text{GAN}}(G_{\mathcal{S}\to\mathcal{T}}, D_{\mathcal{T}}, X_{\mathcal{S}}, X_{\mathcal{T}}) \\ +\mathcal{L}_{\text{GAN}}(G_{\mathcal{T}\to\mathcal{S}}, D_{\mathcal{S}}, X_{\mathcal{T}}, X_{\mathcal{S}})\end{aligned} \tag{8}$$

where λ_{cyc} is a hyperparameter that denotes the relative weight of the reconstruction loss. In the original publication (Zhu et al., 2017), this weight is set to $\lambda_{\text{cyc}} = 10$. The loss can be viewed as being subject to an adversarial min-max game, similar to the one for GANs:

$$\min_{G_{\mathcal{S}\to\mathcal{T}}, G_{\mathcal{T}\to\mathcal{S}}} \max_{D_{\mathcal{S}}, D_{\mathcal{T}}} \mathcal{L}(G_{\mathcal{S}\to\mathcal{T}}, G_{\mathcal{T}\to\mathcal{S}}, D_{\mathcal{S}}, D_{\mathcal{T}}, X_{\mathcal{S}}, X_{\mathcal{T}}) \tag{9}$$

Domain Adaption. Generally, convolutional neural networks (CNNs) only learn features from the domain they are trained on. This causes the networks to perform poorly on other domains, which is called a domain gap. Domain adaptation methods have been developed to overcome this problem. Many studies have been conducted by training a network on synthetic data and then evaluating it on real data. The work of (Hoffman et al., 2018) is close to ours, as they also use the CycleGAN framework and enforce image content retention over a pretrained segmentation network. In this framework, the generator also minimizes a segmentation loss of the transformed image. However, the segmentation network can only be trained on the source domain. The generator is therefore encouraged to produce images near the domain on which the segmentation network was trained to minimize this loss, i.e. the source domain. The authors of (Pizzati et al., 2020) try to bridge the domain gap with images retrieved from online videos. Recent state-of-the art approaches also make usage of SSL approaches such as pseudo labeling (Zou et al., 2018) or consistency regularization (Choi et al., 2019) to close domain gap.

3 METHOLOGY

Our approach consists of a combination of an image-to-image translation module and a SSL method. In Section 3.1 we first formalize the problem setup. Section 3.2 considers the image-to-image translation with the goal to transfer the style of nighttime images to daytime images while keeping the image content. In Section 3.3 we consider the training for semantic segmentation that combines the image-to-image translation with a SSL framework.

3.1 Problem Setup

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. We have access to N labelled segmentation masks for the source domain $\{(\mathbf{x}_s^i, \mathbf{y}_s^i)\}_{i=1}^N$ with $\mathbf{y}_s^i$ as labelled semantic segmentation masks. The target domain has no labelled samples and shares C categories of the source domain. Our task is to train a segmentation network $f_{\mathcal{S}}$ that performs well on both domains. More precisely, we want to additionally learn the new target domain without neglecting the source domain.

3.2 Stabilizing CycleGAN

The aforementioned CycleGAN method may seem as a straightforward way of obtaining additional training data when adapting domains. The problem here is, however, that CycleGAN requires the generators to only submit to a reconstruction goal. This way, the networks are allowed to learn any arbitrary transformation, as long as it can be reversed. In some cases, this leads to transformations which perturb the semantic content of an image, a phenomenon the authors of (Chen et al., 2019) describe with the term hallucinating. This is because the cyclical invertibility of the transformation does not necessarily enforce semantic correctness. Such hallucinations lead to a drastically impaired usability of generated images as training data in subsequent tasks. When adapting or extending to other domains, such images would pose the threat of the segmentation networks learning something false. It might come to mind that in order to enforce a stricter content preservation policy during training, we could simply increase the value of λ_{cyc}. However, due to being a pixel wise loss this would only lead the networks $G_{\mathcal{S}\to\mathcal{T}}$ and $G_{\mathcal{T}\to\mathcal{S}}$ to learn a transformation that is increasingly similar to the identity transformation. We thus require a loss function that puts a stricter constraint on the preservation of content without suppressing the alteration of the images style.

The feature loss from style transfer (see (Gatys et al., 2015), (Gatys et al., 2016)) would in fact be exactly what we need here. It also has been used as a loss function for the training of networks previously. In (Johnson et al., 2016) it is combined with style loss, and in (Atapattu and Rekabdar, 2019) adversarial loss is used to change the visual details of the images. (Hoffman et al., 2018) also use a network based loss, but the network they use has been trained for semantic segmentation.

Feature loss utilizes the insight that a network which has been trained for the task of image recognition, will produce feature tensors that meaningfully represent the content of the image. In our case, we will use a VGG-19 network that has been pretrained on `ImageNet`. To measure the difference between the contents of two images, we simply compare the L2 distance of their feature tensors. The computation of the feature loss itself is as follows. Let $\hat{\mathbf{x}}_s = G_{\mathcal{S}\to\mathcal{T}}(\mathbf{x}_s)$ be the transformed image. Let $k \in \mathcal{K}$ be the set of indices of the layers that are going to be used for the feature loss, and let $\mathsf{T}^k \in \mathbb{R}^{h_k \times w_k \times d_k}$ and $\hat{\mathsf{T}}^k \in \mathbb{R}^{h_k \times w_k \times d_k}$ for $k \in \mathcal{K}$ be the tensors that contain these layers' output activations for $\mathbf{x}_s$ and $\hat{\mathbf{x}}_s$, respectively. The feature loss is then defined as follows:

$$\mathcal{L}_{\text{feature}}(G_{\mathcal{S}\to\mathcal{T}}) = \sum_{k\in\mathcal{K}} \mathbb{E}_{\mathbf{x}_s\sim p_{\text{data}}(\mathbf{x}_s)}[||\mathsf{T}^k - \hat{\mathsf{T}}^k||_2^2]. \tag{10}$$

The loss for $G_{\mathcal{T}\to\mathcal{S}}$ is analogous. The overall loss of CycleGAN (see Equation 8) can be expanded to be

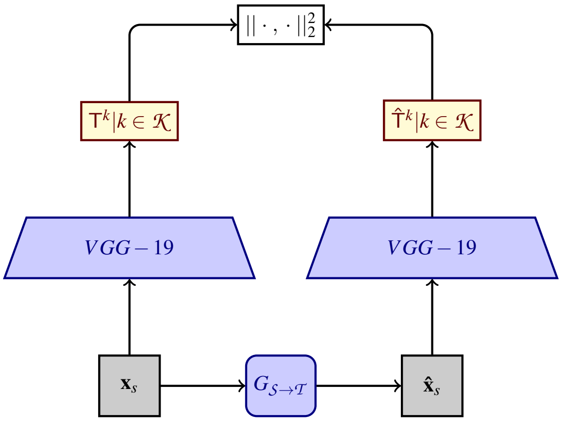

$||\cdot,\cdot||_2^2$
$\mathsf{T}^k|k\in\mathcal{K}$
$\hat{\mathsf{T}}^k|k\in\mathcal{K}$
$VGG-19$
$VGG-19$
$\mathbf{x}_s$
$G_{\mathcal{S}\to\mathcal{T}}$
$\hat{\mathbf{x}}_s$

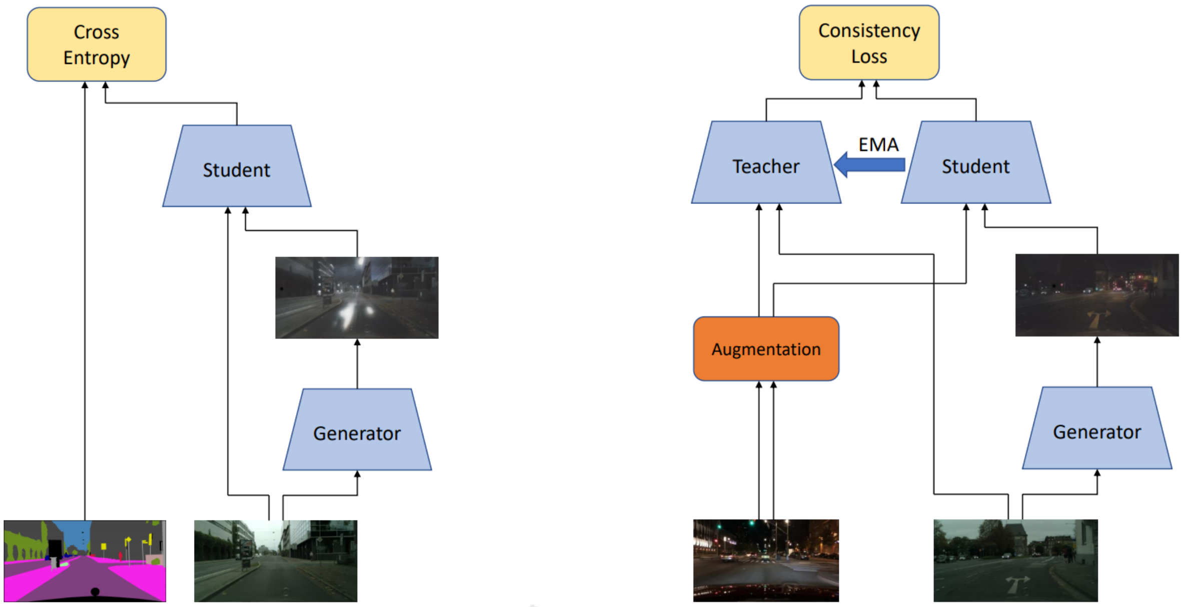
Figure 1: This picture illustrates the functionality of the feature loss for the specific case of the transformation $\hat{\mathbf{x}}_s = G_{\mathcal{S}\to\mathcal{T}}(\mathbf{x}_s)$.

$$\begin{aligned}\mathcal{L}(G_{\mathcal{S}\to\mathcal{T}}, G_{\mathcal{T}\to\mathcal{S}}, D_{\mathcal{S}}, D_{\mathcal{T}}, X_{\mathcal{S}}, X_{\mathcal{T}}) = \\ \lambda_{\text{cyc}}\mathcal{L}_{\text{cyc}}(G_{\mathcal{S}\to\mathcal{T}}, G_{\mathcal{T}\to\mathcal{S}}) \\ +\mathcal{L}_{\text{GAN}}(G_{\mathcal{S}\to\mathcal{T}}, D_{\mathcal{T}}, X_{\mathcal{S}}, X_{\mathcal{T}}) \\ +\mathcal{L}_{\text{GAN}}(G_{\mathcal{T}\to\mathcal{S}}, D_{\mathcal{S}}, X_{\mathcal{T}}, X_{\mathcal{S}}) \\ +\lambda_{\text{feature}}\mathcal{L}_{\text{feature}}(G_{\mathcal{S}\to\mathcal{T}}) \\ +\lambda_{\text{feature}}\mathcal{L}_{\text{feature}}(G_{\mathcal{T}\to\mathcal{S}})\end{aligned} \tag{11}$$

such that λ_{feature} poses an additional hyper parameter for the relative weight of the feature loss. The weights of the VGG-19 network will not be altered during training. Similar to style transfer, we can achieve a higher freedom in texture for the generator if we only take few layers that are occurring later in the network, due to them having developed a more high level representation of the image content. An illustration of the content loss for the special case of $G_{\mathcal{S}\to\mathcal{T}}$ can be found in Figure 1.

3.3 Segmentation Training

As illustrated in Figure 2, our overall objective function is defined by a supervised term from labelled data and a self-supervised term from unlabelled data. As already mentioned, we assume that only the images from the source domain have annotations and that there is unlabelled data for both domains available. Following (Tarvainen and Valpola, 2017), we make use of two networks: A student network f_S and a teacher network f_T, where the architecture of the teacher network is identical to the one of the student network. The student network f_S will be subject to gradient based optimization, whereas the teacher network f_T will not experience any direct training, but only updates its weights as an exponential moving average of the student networks weights. As supervised loss we simply define the cross-entropy loss on the



(a) Supervised Learning.

(b) Self-Supervised Learning.

Figure 2: Schematic of our proposed domain extension method. It is divided into a supervised and a self-supervised training section. The supervised section has access to the ground truth annotations. We calculate a loss on the original image at daytime as well as its transformation to nighttime. The self-supervised training step has access to unlabelled images from daytime and nighttime, respectively. We enforce consistent predictions between the student and the teacher network, where the teacher gets a daytime and an augmented nighttime image, while the student gets the transformed day2night image as well as a different augmented nighttime image.

daytime images as well as their corresponding transformation to nighttime via the network $G_{\mathcal{S}\to\mathcal{T}}$. Since the transformation network has been trained to maintain the content of the image, the labelled annotations can be assumed to be correct even after the daytime to nighttime transformation, and can be propagated into the network without damage. The supervised loss is therefore defined as:

$$\begin{aligned}\mathcal{L}_{\text{sup}} = \mathcal{L}_{\text{ce}}(\sigma(f_{\mathcal{S}}(\mathbf{x}_s)), \mathbf{y}_s)\\ +\mathcal{L}_{\text{ce}}(\sigma(f_{\mathcal{S}}(G_{\mathcal{S}\to\mathcal{T}}(\mathbf{x}_s))), \mathbf{y}_s)\end{aligned} \tag{12}$$

where $\mathcal{L}_{\text{ce}}$ is the standard cross-entropy loss and σ is the softmax function.

The teacher's weights $w_{t,i}$ at time step i are updated by the student's weights $w_{s,i}$ with the formula:

$$w_{t,i} = \alpha w_{t,i-1} + (1-\alpha) w_{s,i}, \tag{13}$$

where α refers to the exponential moving average (EMA) decay that controls the updating rate. During training on unlabelled data, the teacher network guides the learning of the student network by providing its outputs as a reference.

We use a self-supervised loss for two different types of inputs that encourages the output consistency between different styles of the same image. This allows the network to learn the feature space of the target domain. In particular, given an image at nighttime $\mathbf{x}_t \in X_{\mathcal{T}}$, we construct a weak augmentation $A_w(\mathbf{x}_t)$ which we feed into the teacher and a strong augmentation $A_s(\mathbf{x}_t)$ which we feed into the student network and minimize the consistency loss defined as the difference between the outputs of the two networks. In this setup, we use Gaussian noise and color jittering as augmentations. Furthermore, given a daytime image $\mathbf{x}_s$ and its transformation into nighttime $G_{\mathcal{S}\to\mathcal{T}}(\mathbf{x}_s)$, we feed $\mathbf{x}_s$ into the teacher and $G_{\mathcal{S}\to\mathcal{T}}(\mathbf{x}_s)$ into the student. Since it is more challenging to predict the correct segmentation for nighttime images than for daytime images, we can view the transformation from day to nighttime as a form of perturbation for the student network input. This motivates our system to give consistent predictions between daytime and nighttime images, despite the visual appearance of the images. In addition to that, our segmentation network has access to real nighttime images to align the different features of both domains and to learn the underlying structure and style of nighttime images. Following (Tarvainen and Valpola, 2017), we further use dropout at the forward pass of the student as additional perturbation. No dropout is performed at the forward pass of the teacher and when the student receives an image for the supervised loss computation.

As the predictions of the teacher $f_{\mathcal{T}}$ are error-prone, we found it useful to try to remove uncertain predictions from the teacher network for the loss calculation. For each pixel, we can view the maximum output probability over all the C categories as a measurement of how confident f_T is about its prediction, and exclude pixels for which the probability is below a certain threshold. The excluded pixels will not play a role in the loss propagation in order to stabilize the training. Only pixels with high confidence will be left and the student network can learn reliable target predictions from the teacher. We can calculate a mask for each sample $\mathbf{x}$ as

$$w_{i,j}(f_T(\mathbf{x})) = \begin{cases} 1 & \text{if } \max_{c\in\{1,\ldots,C\}} \sigma(f_T(\mathbf{x}))_{i,j,c} \geq \rho \\ 0, & \text{otherwise} \end{cases} \tag{14}$$

where σ refers to the softmax activation function for pixel positions (i, j) and ρ is the previously mentioned confidence threshold. We can use this mask to mask out uncertain predictions. The final self-supervised loss is now defined as:

$$\begin{aligned} \mathcal{L}_{\text{ssl}} = \\ ||w_{i,j}(f_T(\mathbf{x}_t)) \cdot (\sigma(f_T(A_w(\mathbf{x}_t))) - \sigma(f_S(A_s(\mathbf{x}_t))))||_2^2 + \\ ||w_{i,j}(f_T(\mathbf{x}_s)) \cdot (\sigma(f_T(\mathbf{x}_s)) - \sigma(f_S(G_{\mathcal{S}\to\mathcal{T}}(\mathbf{x}_s))))||_2^2 \end{aligned} \tag{15}$$

Note that this loss is only propagated into the student network and not into the teacher network.

By combining the consistency loss (Eq. 15) with the supervised loss (Eq. 12), the overall objective function is defined as:

$$\mathcal{L}_{\text{overall}} = \mathcal{L}_{\text{sup}} + \lambda_{\text{ssl}} \mathcal{L}_{\text{ssl}}, \tag{16}$$

where λ_{ssl} is a trade-off parameter.

4 EXPERIMENTS

This section describes the experimental setup and details of the proposed approach. We provide experimental results for our stabilized CycleGAN as well as the SSL learning procedure.

4.1 Datasets

Cityscapes (CS). The Cityscapse dataset (Cordts et al., 2016) contains images of urban street scenes collected from 50 cities around Germany and neighboring countries. It consists of a training set of 2975 images and a validation set of 500 images. The images were all taken at daytime and have moderate diversity in weather and lighting conditions. We use this dataset for supervised training and report results using the standard 19 training classes. The dataset further comprises 19998 coarsely annotated images.

BDD Dataset (BDD). The BDD dataset (Yu et al., 2018) is a large driving video dataset which consists of 100.000 images collected from different cities. It has a large set of images taken at nighttime. It further provides pixel-wise semantic segmentation labels for a very small subset. However, by manual inspection we observed that they contain labeling errors which makes them unsuitable for evaluation. This has also been mentioned in (Sakaridis et al., 2020). We therefore only use the images of this dataset.

Nighttime Driving (ND). The Nighttime dataset (Dai and Gool, 2018) was collected during 5 rides in multiple Swiss cities and their suburbs using a GoPro Hero 5 camera. The dataset contains 50 annotated nighttime images, referred to as *Nighttime Driving-test*. The semantic annotations follow the 19 evaluation classes of the Cityscapes dataset. It further assigned a void label to pixels which shall not be used for evaluation. In this paper, we utilize the annotated nighttime images for evaluation. At the time of this work, the unlabelled nighttime images were not available. We have noticed that only 17 of the standard 19 classes are annotated in this test set. We therefore report results only for those 17 classes.

Dark Zurich Dataset (DZ). The Dark Zurich dataset (Sakaridis et al., 2019) was recorded in Zurich using a GoPro 5 camera as well. It comprise 3041 daytime, 2920 twilight and 2416 nighttime images.

In summary, for the image-to-image translation training, we use all images of the CS dataset as daytime images and all nighttime images of the BDD and DZ dataset. The 2975 fine labelled images from CS are used for supervised training. For SSL training, we utilize all images from CS and all nighttime images of BDD and DZ. The ND dataset is used for the evaluation of the nighttime domain.

4.2 Implementation Details

The image transformation framework and the networks used for segmentation are trained separately. The generators used for image transformation have the same architecture as in (Zhu et al., 2017), with one exceptio. In order to suppress checkerboard artifacts (Odena et al., 2016), the deconvolution layers have been replaced by a concatenation of a bilinear upsampling layer and a subsequent regular con-

Table 1: Results of our methods on the CS and ND validation set in the center and right column. The leftmost column contains the method and the respective training data separated with a ”-”, while a ”+” indicates a combination of multiple methods or datasets. The ”Baseline” model was only trained on the Cityscapes training set and without SSL. ”Day2Night (CS)” denotes the Cityscapes dataset translated to nighttime prior to the segmentation training. If ”Day2Night (CS)” is used, the translation was carried out with the respective image translation method mentioned in the row.

	Cityscapes		Nighttime	
Method - Train Data	mIoU	Acc	mIoU	Acc
Baseline - Daytime (CS)	**68.8**	**94.7**	35.0	80.0
CycleGAN - Day2Night (CS)	52.9	90.9	40.34	84.8
CycleGAN + FL - Day2Night (CS)	63.7	93.5	44.7	85.4
CycleGAN + FL - Day2Night (CS) + Daytime (CS)	67.4	94.3	45.1	85.5
SSL + CycleGAN + FL - (CS + Day2Night (CS) + DZ + BDD)	68.2	94.4	**56.5**	**90.1**

volutional layer with the same filter size and channel number as the replaced deconvolution. Due to the high resolution of our images, we use the version with 9 residual blocks. Similar as in the CycleGAN framework, our discriminator is a PatchGAN discriminator (Isola et al., 2017). The Adam optimizer ($\beta_1 = 0.5, \beta_2 = 0.999$) (Kingma and Ba, 2017) is used for training the networks, with an initial learning rate of 0.0002 for the first 100.000 iterations and 0.0001 after that. The generators / discriminators are trained for a total of 700.000 iterations. Additionally, the gradient norms are clipped to 5.0 per network, in order to stabilize the training. Following common practise, we save the exponential moving average of the generator for further usage (Yazici et al., 2019), with an EMA decay of 0.9999. For the feature loss, we utilize the `block5_conv2` layer of the VGG-19 architecture. We have noticed empirically that this layer gives slightly more stable results than others.

As segmentation network architecture we use the DeeplabV2 (Chen et al., 2018) architecture for the student and teacher network. The backbone is a VGG-16 (Simonyan and Zisserman, 2015) network, which has been pretrained for classification on `ImageNet`. For stabilization, we report results of our teacher model for evaluation. For those trainings without SSL, we still maintain an EMA version of the segmentation network. We train the segmentation network for 50.000 iterations with a batch size of 8 on random crops of resolution 256×256 from images rescaled to 512×1024. The EMA decay for the teacher network is at 0.999. The confidence threshold ρ is set to 0.5 at the beginning and linearly increased to 0.9 during training. We select a warm-up phase of 10.000 steps, where no SSL loss is used for training. After the 10.000th iteration both the SSL loss and the supervised loss are used to compute the gradient. A batch consists of daytime images, converted daytime images and nightime images. Since the total SSL loss is generally quite small during training, we set the relative weight λ_{ssl} (if used) to an initial value of 50 and linearly increase the value until it has doubled at the end of the training. The optimizer in use is Adam with a learning rate of 0.0001. The image generation networks will not be updated during the training of the segmentation network.

4.3 Results

We run several experiments and compare the results to a baseline model. The baseline version was only trained on the Cityscapes training set without SSL or transformed images. First, we compare our extended CycleGAN method with the standard CycleGAN without feature loss. The two implementations differ in the additional feature loss alone. To quantitatively evaluate whether the generated images of our extended method are better suited for segmentation, we trained a segmentation network only on the transformed images of the Cityscapes training set for both our extended method and those from the standard CycleGAN. We report standard IoU and accuracy as performance measurements. Table 1 summarizes the evaluation on the CS and the ND dataset. It can be seen that the segmentation network that receives only transformed CS images with standard CycleGAN performs worse both on CS and ND compared to our modification with an additional feature loss. The fact that the model trained with CycleGAN images is significantly worse on the CS validation set can be explained by the fact that standard CycleGAN does not preserve semantic content well. Compared to the baseline, the segmentation network trained on CycleGAN + FL images performs better on the nighttime dataset and slightly worse at the daytime dataset. This can be explained by the fact that the network only receives transformed images and therefore no images from the source domain directly. To analyse this more precisely, we also trained our segmentation network on 50% transformed CS images as well as

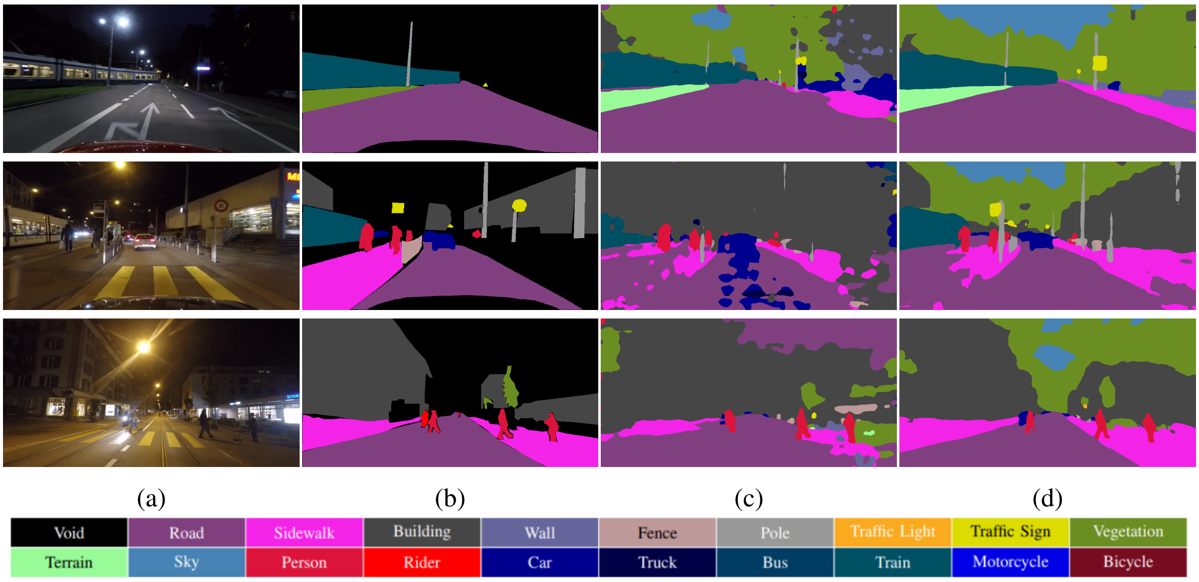


Figure 3: **[Best viewed in color]** (a) Input image. (b) Ground-truth segmentation mask. (c) Prediction of our baseline model trained only on (CS). (d) Prediction of our extended method.

50% non-transformed images per batch. This way the trained model has access to both domains. Compared to the baseline we can see that we improved mIoU by 10% at the nighttime domain while slightly dropping performance on the daytime domain. Table 1 also reports results of our segmentation network trained with our SSL approach. It can be seen that it further improves the performance at the nighttime dataset. Compared to the baseline, we improved the mIoU on the nighttime dataset by 21.5%, while keeping the performance on the daytime dataset stable.

Two reasons may explain the slight drop in performance in the daytime validation set in our trained models with CycleGAN+FL images on the one hand and with additional SSL loss on the other hand. Firstly compared to the baseline, the network now learns features of an additional domain. It either does so by learning domain invariant features or by different modes within the network. Furthermore, the network is no longer strongly adapted to a single domain, which means that we no longer overfit this domain strongly. Secondly, even though we enforce to keep the content of the image after the day to night transformation, it may still be possible that the borders of the segmentation masks slightly change. As we use a lower dimensional feature map for the loss calculation, we can not guarantee that this is avoided.

We also show the improvement on nighttime images qualitatively in Figure 3. It compares the predictions of our proposed method to the baseline. With our extended training, the predicted labels are more precise. In all rows, the examples show a better estimation of the road and also smaller objects like people and vehicles are detected more finely.

5 CONCLUSION

In this work, we investigated the problem of unsupervised domain extension by training a network for the task of semantic segmentation on daytime images and additionally making it robust for nighttime images. We proposed two complementary approaches to solve this task. In the first step, daytime images were transformed into nighttime images and vice versa via the CycleGAN framework. We proposed a simple additional mechanism to retain more of the semantic content during the transformation for better supervised segmentation learning. In the second step, we integrated the transformation network into a state-of-the art self-supervised learning approach. Overall, we improved the performance on nighttime images compared to a baseline by a large margin, while keeping the performance on the source domain stable. These results demonstrate the effectiveness of our approach, making segmentation networks trained with only labelled daytime images perform robustly at night.

REFERENCES

Atapattu, C. and Rekabdar, B. (2019). Improving the realism of synthetic images through a combination of adversarial and perceptual losses. In *2019 International*

Joint Conference on Neural Networks (IJCNN), pages 1–7.

Chen, L., Papandreou, G., Kokkinos, I., Murphy, K., and Yuille, A. L. (2018). Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE Trans. Pattern Anal. Mach. Intell.*, 40(4):834–848.

Chen, Y., Li, W., Chen, X., and Gool, L. V. (2019). Learning semantic segmentation from synthetic data: A geometrically guided input-output adaptation approach. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Choi, J., Kim, T., and Kim, C. (2019). Self-ensembling with gan-based data augmentation for domain adaptation in semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.

Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., and Schiele, B. (2016). The cityscapes dataset for semantic urban scene understanding. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Dai, D. and Gool, L. (2018). Dark model adaptation: Semantic image segmentation from daytime to nighttime. *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, pages 3819–3824.

Gatys, L. A., Ecker, A. S., and Bethge, M. (2015). A neural algorithm of artistic style.

Gatys, L. A., Ecker, A. S., and Bethge, M. (2016). Image style transfer using convolutional neural networks. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2414–2423.

Hoffman, J., Tzeng, E., Park, T., Zhu, J.-Y., Isola, P., Saenko, K., Efros, A. A., and Darrell, T. (2018). Cycada: Cycle consistent adversarial domain adaptation. In *International Conference on Machine Learning (ICML)*.

Huang, X., Liu, M.-Y., Belongie, S., and Kautz, J. (2018). Multimodal unsupervised image-to-image translation. In *ECCV*.

Isola, P., Zhu, J.-Y., Zhou, T., and Efros, A. A. (2017). Image-to-image translation with conditional adversarial networks. *CVPR*.

Johnson, J., Alahi, A., and Fei-Fei, L. (2016). Perceptual losses for real-time style transfer and super-resolution. In *European Conference on Computer Vision*.

Kingma, D. P. and Ba, J. (2017). Adam: A method for stochastic optimization.

Lee, D.-H. (2013). Pseudo-label : The simple and efficient semi-supervised learning method for deep neural networks. *ICML 2013 Workshop : Challenges in Representation Learning (WREPL)*.

Liu, M.-Y., Breuel, T., and Kautz, J. (2017). Unsupervised image-to-image translation networks. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R., editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Nizan, O. and Tal, A. (2020). Breaking the cycle - colleagues are all you need. In *IEEE conference on computer vision and pattern recognition (CVPR)*.

Odena, A., Dumoulin, V., and Olah, C. (2016). Deconvolution and checkerboard artifacts. *Distill*.

Pizzati, F., Charette, R. d., Zaccaria, M., and Cerri, P. (2020). Domain bridge for unpaired image-to-image translation and unsupervised domain adaptation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*.

Sakaridis, C., Dai, D., and Van Gool, L. (2019). Guided curriculum model adaptation and uncertainty-aware evaluation for semantic nighttime image segmentation. In *The IEEE International Conference on Computer Vision (ICCV)*.

Sakaridis, C., Dai, D., and Van Gool, L. (2020). Map-guided curriculum domain adaptation and uncertainty-aware evaluation for semantic nighttime image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Simonyan, K. and Zisserman, A. (2015). Very deep convolutional networks for large-scale image recognition. In Bengio, Y. and LeCun, Y., editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958.

Tarvainen, A. and Valpola, H. (2017). Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R., editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

van Engelen, J. E. and Hoos, H. H. (2020). A survey on semi-supervised learning. *Machine Learning*, 109(2):373–440.

Verma, V., Lamb, A., Kannala, J., Bengio, Y., and Lopez-Paz, D. (2019). Interpolation consistency training for semi-supervised learning. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19*, pages 3635–3641. International Joint Conferences on Artificial Intelligence Organization.

Xie, Q., Hovy, E. H., Luong, M., and Le, Q. V. (2019). Self-training with noisy student improves imagenet classification. *CoRR*, abs/1911.04252.

Yazici, Y., Foo, C., Winkler, S., Yap, K., Piliouras, G., and Chandrasekhar, V. (2019). The unusual effectiveness of averaging in GAN training. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

Yu, F., Xian, W., Chen, Y., Liu, F., Liao, M., Madhavan, V., and Darrell, T. (2018). BDD100K: A diverse driving video database with scalable annotation tooling. *CoRR*, abs/1805.04687.

Zhang, H., Cisse, M., Dauphin, Y. N., and Lopez-Paz, D. (2018). mixup: Beyond empirical risk minimization. *International Conference on Learning Representations*.

Zhao, Y., Wu, R., and Dong, H. (2020). Unpaired image-to-image translation using adversarial consistency loss. In Vedaldi, A., Bischof, H., Brox, T., and Frahm, J.-M., editors, *Computer Vision – ECCV 2020*, pages 800–815, Cham. Springer International Publishing.

Zhu, J.-Y., Park, T., Isola, P., and Efros, A. A. (2017). Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Computer Vision (ICCV), 2017 IEEE International Conference on*.

Zou, Y., Yu, Z., Kumar, B. V., and Wang, J. (2018). Unsupervised domain adaptation for semantic segmentation via class-balanced self-training. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 289–305.