Confidence-and-Refinement Adaptation Model for Cross-Domain Semantic Segmentation

Xiaohong Zhang, *Graduate Student Member, IEEE*, Yi Chen, *Member, IEEE*, Ziyi Shen, Yuming Shen, Haofeng Zhang, *Member, IEEE*, and Yudong Zhang, *Senior Member, IEEE*

***Abstract*— With the rapid development of convolutional neural networks (CNNs), significant progress has been achieved in semantic segmentation. Despite the great success, such deep learning approaches require large scale real-world datasets with pixel-level annotations. However, considering that pixel-level labeling of semantics is extremely laborious, many researchers turn to utilize synthetic data with free annotations. But due to the clear domain gap, the segmentation model trained with the synthetic images tends to perform poorly on the real-world datasets. Unsupervised domain adaptation (UDA) for semantic segmentation recently gains an increasing research attention, which aims at alleviating the domain discrepancy. Existing methods in this scope either simply align features or the outputs across the source and target domains or have to deal with the complex image processing and post-processing problems. In this work, we propose a novel multi-level UDA model named Confidence-and-Refinement Adaptation Model (CRAM), which contains a confidence-aware entropy alignment (CEA) module and a style feature alignment (SFA) module. Through CEA, the adaptation is done locally via adversarial learning in the output space, making the segmentation model pay attention to the high-confident predictions. Furthermore, to enhance the model transfer in the shallow feature space, the SFA module is applied to minimize the appearance gap across domains. Experiments on two challenging UDA benchmarks "GTA5-to-Cityscapes" and "SYNTHIA-to-Cityscapes" demonstrate the effectiveness of CRAM. We achieve comparable performance with the existing state-of-the-art works with advantages in simplicity and convergence speed.**

***Index Terms*— Semantic segmentation, unsupervised domain adaptation, style feature alignment, confidence-aware entropy alignment.**

Manuscript received June 20, 2021; revised October 17, 2021 and November 14, 2021; accepted December 31, 2021. This work was supported in part by the National Natural Science Foundation of China under Grant 61872187, Grant 62072246, and Grant 62077023; in part by the Natural Science Foundation of Jiangsu Province under Grant BK20201306; and in part by the 111 Program under Grant B13022. The Associate Editor for this article was S. Wan. *(Xiaohong Zhang and Yi Chen contributed equally to this work.) (Corresponding authors: Haofeng Zhang; Yudong Zhang.)*

Xiaohong Zhang and Haofeng Zhang are with the School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing 210094, China (e-mail: zhangxiaohong00@njust.edu.cn; zhanghf@njust.edu.cn).

Yi Chen is with the School of Computer Science and Technology, Nanjing Normal University, Nanjing 210023, China (e-mail: cystory@gmail.com).

Ziyi Shen is with the Department of Medical Physics and Biomedical Engineering, University College London, London WC1E 6BT, U.K. (e-mail: joanshen0508@gmail.com).

Yuming Shen is with the Department of Engineering Science, University of Oxford, Oxford OX1 2JD, U.K. (e-mail: ym_zmxncbv@hotmail.com).

Yudong Zhang is with the School of Informatics, University of Leicester, Leicester LE1 7RH, U.K. (e-mail: yudongzhang@ieee.org).

Digital Object Identifier 10.1109/TITS.2022.3140481

I. Introduction

SEMANTIC segmentation, a fundamental task of autonomous-driving, aims to assign each pixel a class label, e.g., building, road, vegetation or pedestrian. Recently, the development of convolutional neural networks (CNNs) has pushed the state-of-art in the field of semantic segmentation [1]. However, most supervised deep learning approaches [2]–[6] crucially rely on sufficient real-world images with fine segmentation annotations, which are usually expensive and labor intensive. Considering the trivial data collection/annotation procedure, some researchers [7]–[11] treat large-scale synthetic datasets [12]–[14] with annotations as alternative training signals to train a segmentation model. Despite the availability of high-quality semantic labelings for synthetic datasets, the clear domain discrepancy between synthetic (source) and real (target) images always brings about a sharp drop in the performance of an excellent segmentation model. Thus, to deal with such serious problem, researchers have strived to study the methods of unsupervised domain adaptation (UDA) for semantic segmentation.

Several recent works [15]–[18] employ image-to-image translation, narrowing the appearance gap between domains. One approach yields style transfer, which always goes through the tedious post-processing and makes the segmentation model unable to perform one-stage training. To reduce the appearance gap, adversarial mechanism is also applied. However, the training of adversarial networks is complex and unstable. Despite the effectiveness in bridging the domain gap in the appearance level, the above methods ignore rich structural information in the output space, thus neglecting the strong semantic similarities between the source and target domains.

Among some other approaches, [19]–[21] address the adaptation of semantic segmentation networks by making the distribution of high-dimensional features or final outputs close to each other between source and target images via adversarial mechanism. However, these methods perform adaptation in the high-dimensional feature/output level. As a consequence, those shallower features, which contain abundant texture information, can not be adapted well.

The third method is self-training(ST) [15], [22], which uses pseudo-labels to further fine-tune the segmentation model. This method usually focuses on how to choose the output of the target domain with higher confidence as the pseudo label. Therefore, it cannot guarantee one-step end-to-end training, and the process of generating pseudo-labels is relatively cumbersome. It can be exploited as an aid for other methods to further improve the performance.

Considering the above problems of existing UDA methods, in this paper, we propose a multi-level adaptation strategy named Confidence-and-Refinement Adaptation Model (CRAM), in which the adaptation is done in both structured output and shallower feature space. As shown in Figure 1, CRAM includes two components: 1) the confidence-aware entropy alignment (CEA) module; 2) the style feature alignment (SFA) module. With the two components, the domain gap between the two domains is not only reduced at the appearance level, but also at the deep output level.

With the CEA module, the gap between synthetic (source) and real-world (target) data can be narrowed in the output space. Explicitly, we align the distribution of entropy of predictions between source and target images for two reasons. First, this module adapts the pixel-wise structured outputs across domains, which are rich in the spatial and local information. Secondly, with the assistance of entropy minimization, the segmentation model will avoid generating low-confident predictions for the target domains. The entropy-based algorithm is essentially relevant with self-training (ST) method for the model fine-tuning. Unlike the prior works [9], [23], a novel confidence-aware entropy (CE) is proposed, so the segmentation model is forced to pay more attention to the high-confident (low-entropy) predictions. As a result, the UDA method for segmentation based on the modified entropy, when optimized with adversarial learning, converges faster and achieves a better performance on the target domain.

Our SFA module mitigates the appearance gap between synthetic and real images in the shallower feature level, inspired by [24]. In detail, a style feature (SF) is introduced, which is the Gram matrix of the immediate feature of the segmentation model. To reduce the domain gap in the appearance level, our method simply minimizes the distance of the style feature between source and target domains. Therefore, through simply aligning the style features across two domains, we further close the appearance gap between source and target images in the feature level. Compared to the general UDA methods based on style transfer, our SFA module requires less memory overhead and achieves competitive performance in an one-stage end-to-end way.

Thus, compared with other UDA methods for semantic segmentation, our UDA approach CRAM enjoys two main advantages from a practical perspective. The first is its simplicity. It is precisely to address the adaptation of semantic segmentation networks that cumbersome pipelines, *i.e.*, involving image-to-image adaptation, feature alignment, model fine-tuning with pseudo labels, have evolved. In our work, neither CEA nor SFA adds significant overhead to the segmentation network. The second is the fast convergence speed. With the multi-level adaptation mechanism, our network requires fewer training iterations to achieve the best results than the baseline model, which will be shown in the experiments.

The main contributions of this work are as follows:

(1) We propose a novel multi-level UDA model for semantic segmentation named Confidence-and-Refinement Adaptation Model (CRAM). Different from the common UDA methods, we conduct the adaptation in both structured output and shallower feature space. With the help of the multi-level adaptation strategy, the segmentation network trained with the annotated source data can yield promising generalization on the target domain.

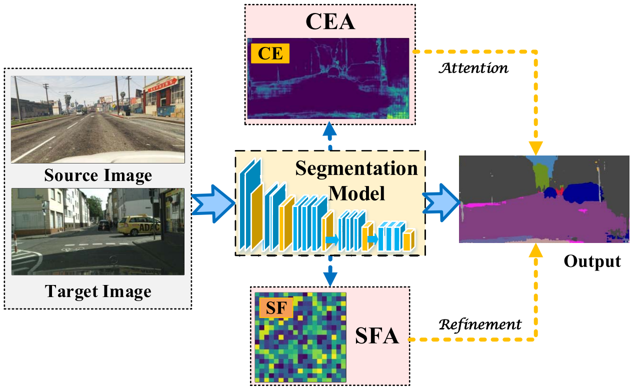


Fig. 1. Schematic of the proposed CRAM model. It contains two UDA modules. First, with the adaptation in the output space, the confidence-aware entropy alignment (CEA) module guides the segmentation model to generate high-confident model. Second, the style feature alignment (SFA) module aims to minimize the appearance gap across domains.

(2) To realize the multi-level adaptation mechanism, our model is comprised of two key adaptation modules: a confidence-aware entropy alignment (CEA) module and a style feature alignment (SFA) module. First, with the adversarial training, the CEA module aligns confidence-aware entropy (CE) of predictions between source and target images. Different from the common entropy, the proposed CE forces the segmentation network to put emphasis on the high-confidence predictions. Second, the SFA module directly minimizes the distance of the style feature between two domains. In this module, the introduced style feature represents the style of input images. It is a simple yet efficient approach to reduce cross-domain discrepancy in the appearance level.

(3) With advantages in simplicity and convergence speed, our methods can achieve comparable performance with other state-of-the-art methods on major cross-domain benchmark datasets such as “GTA5-to-Cityscapes” and “SYNTHIA-to-Cityscapes”.

In the rest of the paper, we begin by discussing different methods for UDA in semantic segmentation (section II). We then describe the architecture of our domain adaptation model CRAM (section III). Finally, we analyze the results of experiments and ablation studies (section IV).

II. Related Work

A. Semantic Segmentation

As a fundamental component of a powerful computer vision system, semantic segmentation has received extensive research attention. Over the past few years, with the significant progress of deep learning and CNNs, great advances have been reported for semantic segmentation. FCN [4], the first segmentation network that successfully generated pixel-level predictions, invents the decoder-encoder architecture. Later, SegNet [2] and DeepLab [25]–[27] are proposed based on CNNs, mostly following the pipeline similar to FCN. However, the encoder-decoder based FCN architecture fails to obtain full-image contextual information, which limits the model's ability to yield more precise predictions.

In order to tackle the above drawback, the idea of exploiting the global context has been considered several times already in the literature. For instance, PSPNet [28] utilizes global features to capture long-range dependencies. Afterwards, [29] invents an efficient context scheme which focuses on the necessary object information. CCNet [30] proposes a novel criss-cross attention module that can better capture contextual information. Different from the dominating encoder-decoder based FCN architecture, alternative approaches also exploit the transformer framework for a powerful segmentation model, such as SETR [31] and SegFormer [32]. Additionally, to further boost the segmentation accuracy, it is also a more efficient and effective way to embed a novel CNN building block in a mature architecture, one example being the Asymmetric Convolution Block [33]

However, to match the capacity of these advanced networks, large scale datasets with pixel-level annotations are required during training phase. Meanwhile, the pixel-wise labeling of semantics is labor intensive and cost expensive [34]–[36]. The difficulty in segmentation data collection shapes two main families of solutions. One is weakly-supervised semantic segmentation [27], [37]–[39], [39]–[41] which utilizes image-level labels instead of the pixel-level annotations. The other one trains the segmentation model with synthetic data, of which ground-truth semantics can be obtained freely. But due to domain mismatch between synthetic and real images, the latter method always leads to a decrease in the performance of the model on the real-world dataset. Therefore, this has intensified the interest in unsupervised domain adaptation (UDA) for semantic segmentation [42]–[44], which will be discussed below.

B. Unsupervised Domain Adaptation

The goal of unsupervised domain adaptation is to enable deep stereo methods generalize well to new domains. UDA is of particular significance since it addresses the problem of expensive labeling efforts in collecting annotation of the real-world (target) datasets. Therefore, it sets an innovative and promising research direction for many tasks, such as classification, stereo matching and semantic segmentation.

In this work, we show particular interest in the task of UDA for semantic segmentation. Therefore, we only focus on the UDA methods for segmentation here. UDA for semantic segmentation aims to adapt the segmentation model trained on synthetic (source) data with free annotations to the unlabeled real-world (target) datasets. With the UDA methods, the performance of trained segmentation model will not drop dramatically in the target domain despite the presence of domain shift.

In the context of semantic segmentation, current UDA techniques [43], [45]–[47] have pursued three main directions to bridge the domain gap.

The first employs style transfer [16] or adversarial mechanism [15], [48] to transfer images across domains, thus narrowing the appearance gap between domains. However, when using style transfer, the approach has to deal with the troublesome image processing and post-processing problems. With this method alone, the trained model has not yet guaranteed a good generalization to target data without auxiliary operations. In addition, training of adversarial networks is complicated. By contrast, to bridge the appearance gap, the style feature alignment module proposed in our paper simply minimizes the distance of the style feature between source and target domains, with advantages in terms of the ease of training

The second family considers matching the distributions of representations [49]–[52] or of the final outputs [21], [53] for either source or target domains. These methods aim at globally aligning deep representations across two domains. This algorithm towards domain adaptation is realized by virtue of adversarial training. In our work, we choose to match the entropy of predictions across domains in that the structured outputs contain rich semantic information which is shared across domains.

Others resort to self-training (ST) [15], [22], [54], aiming to finetune the model with the pseudo labels, which is chosen from high-confident predictions of target data. However, the selection of pseudo labels is a challenging task. In our work, the performance of entropy minimization applied in the CEA module is equal to that of ST [9]. Moreover, without generating pseudo-label, our method is both speed and memory efficient, adding no extra workload.

The three mainstream methods mentioned above prefer to adapt domains at either the image level, the intermediate feature level or the output level. Differently, a recent work [55] performs domain adaptation in affinity space, which benefits from the affinity relationship between adjacent pixels.

In addition, some researchers study the methods of adapting the semantic segmentation model learnt on the labeled daytime dataset to unlabeled nighttime dataset. Based on generative adversarial networks, [56] performs an image-to-image translation to minimize the appearance gap between domains. DANNet [57] is the first adaptation framework that can perform one-stage training for nighttime semantic segmentation.

Recently, the centroid-aware methods [47], [54], [58], [59] have been garnering additional interest in the area of domain adaptive Semantic Segmentation. For the fine-grained feature alignment, these methods focus on reducing the distance between the corresponding classes of two domains. In addition, there are also many effective solutions for domain adaptation, e.g., Knowledge Distillation [54], [60] and Mixing [59], [61]. The proposal of these methods leads to considerable progress on major open benchmark datasets. One example is the recent work BAPA-Net [59]. By integrating both prototype alignment [11] and mixed sampling [61], BAPA-Net sets a new global state-of-the-art over all existing UDA methods for semantic segmentation. However, the training process of these novel methods is prone to be time-consuming and requires much computational resources.

In contrast, our method enables one-step end-to-end learning and enjoys the advantages of speed and efficiency. In the future work, we will strive to seek the solutions to maintain the balance between accuracy and efficiency in the area of cross-domain semantic segmentation.

C. Entropy Minimization

In [62], entropy minimization (EM) was first proposed for the task of semi-supervised learning. In the field of clustering, it has been shown to be an effective method in [63], [64].

In recent years, EM has been widely applied in many UDA tasks.

In the field of classification, [65] introduces EM for domain adaptation. Reference [66] invents a hyper-parameter validation approach such that the minimization of the domain gap and the supervised classification based on the source domain can reach an optimal balance. In [67], for the classification task, the UDA method proposes a novel domain alignment layer with which the domain gap is bridged. EM is also applied in [67] to promote classification models with high confident predictions of target domain. In order to promote the transferability of representations, Adversarial Entropy Optimization (AEO) [68] not only minimizes the entropy of the distribution from the source or target domain, but also maximizes the entropy of the combined distribution of source domain and target domain.

For semantic segmentation task, entropy is a form of weighted self-information, it contains rich structural information. AdvEnt [9] is the first UDA approach that successfully applied EM for model transfer. To realize EM, [9] proposes two different UDA approach. The first is direct entropy minimization. It calculates the entropy loss of the input image by summing the entropy of each pixel-level prediction. Then the segmentation model is optimized with the entropy loss through gradient descent. However, through direct EM, the structural dependence between local semantics is ignored. To tackle this issue, the second UDA method adopts adversarial mechanism to conduct adaptation in the self-information space. To conclude, AdvEnt introduces a combination of generative and adversarial techniques through multiple losses for the task of UDA. Similar to AdvEnt, [23] also conducts adversarial learning in the entropy space. Inspired by Smoothness training, [69] presents a novel neutral cross-entropy loss to tackle the over-sharpness of EM and the bias toward easy samples.

Motivated by [9], [23], for the goal of alleviating the domain shift, our CEA module applies adversarial technique to generate similar distributions in the weighted self-information space for either source or target images. Differently, we propose a novel confidence-aware entropy based on a new target distribution, which can better help the segmentation model take advantage of high-confident predictions.

The entropy applied in the previous work is the normalized Shannon entropy, and the value of each entropy is only related to the distribution of the associated pixel. However, directly perform adaptation in the entropy space may bring about bias toward easy samples. Thus we choose to modify the normal entropy via contrastively strengthening the distribution, *i.e.*, enhancing the contrast among the pixel-wise entropy to emphasize the most salient area. Moreover, when minimizing the adversarial loss during training, the gradient is allowed to diffuse to the whole image.

III. Confidence-and-Refinement Adaptation Model

Let $\{\mathcal{I}_S\}$ denote a set of images $\subset \mathbb{R}^{H\times W\times 3}$ from source domain $\mathcal{S}$ with the corresponding pixel-level C-class labels $\mathcal{Y}_S \subset (1, C)^{H\times W}$; in the meantime, let $\{\mathcal{I}_T\}$ denote a set of unlabeled images from target domain $\mathcal{T}$. Our work focuses on addressing the challenging UDA task, aiming at forcing the segmentation model $\mathcal{F}$ trained on $\{\mathcal{I}_S\}$ to perform well on $\{\mathcal{I}_T\}$.

For the purpose, we propose a multi-level cross-domain semantic segmentation model CRAM containing two domain adaptation modules: (1) confidence-aware entropy alignment (CEA) for structured output adaptation; (2) style feature alignment (SFA) for global style adaptation. In order to construct our model, we modify and extend the existing semantic segmentation architecture, such as DeepLab-v2. Figure 2 depicts our architecture. In the remainder of this section, the illustration of each component will be given in detail.

A. Confidence-Aware Entropy Alignment

Through the CEA module, the domain shift between synthetic and real images is reducing, thereby promoting pixel-wise adaptation for semantic segmentation in the output space.

1) Preliminary: For the source domain, take a sample I_S from $\{\mathcal{I}_S\}$ as an input, the segmentation model $\mathcal{F}$ produces a "soft-segmentation map" $P_S = \mathbf{F}(I_S) \in \mathbb{R}^{H\times W\times C}$. Note that $P_S^{(h,w)} = \left[P_S^{(h,w,c)}\right]_c$ provides the prediction of pixel (h, w) as a discrete distribution over C classes. Then the entropy map E_S can be computed through the soft-segmentation prediction (See [4] for a detailed definition):

$$E_S^{(h,w)} = -\sum_c P_S^{(h,w,c)} \log P_S^{(h,w,c)}. \tag{1}$$

Similarly, the C-dimensional "soft-segmentation map" of the target sample I_T can be written as: $P_T = \mathbf{F}(I_T) \in \mathbb{R}^{H\times W\times C}$. The corresponding entropy E_T can also be derived according to [4]:

$$E_T^{(h,w)} = -\sum_c P_T^{(h,w,c)} \log P_T^{(h,w,c)}. \tag{2}$$

As to the source domain, given the semantic annotation $Y_S \in \mathbb{R}^{H\times W\times C}$, model $\mathcal{F}$ can be trained directly with the cross-entropy loss:

$$\mathcal{L}_{seg}\left(I_S, Y_S\right) = -\sum_{h,w}\sum_c Y_S^{(h,w,c)} \log \boldsymbol{P}_S^{(h,w,c)}, \tag{3}$$

where $\left[Y_S^{(h,w,c)}\right]_c$ represents the semantic label at pixel (h, w) in the form of a one hot C-dimensional vector.

Since the entropy, one type of weighted self-information, indicates the confidence of predictions, one effective approach

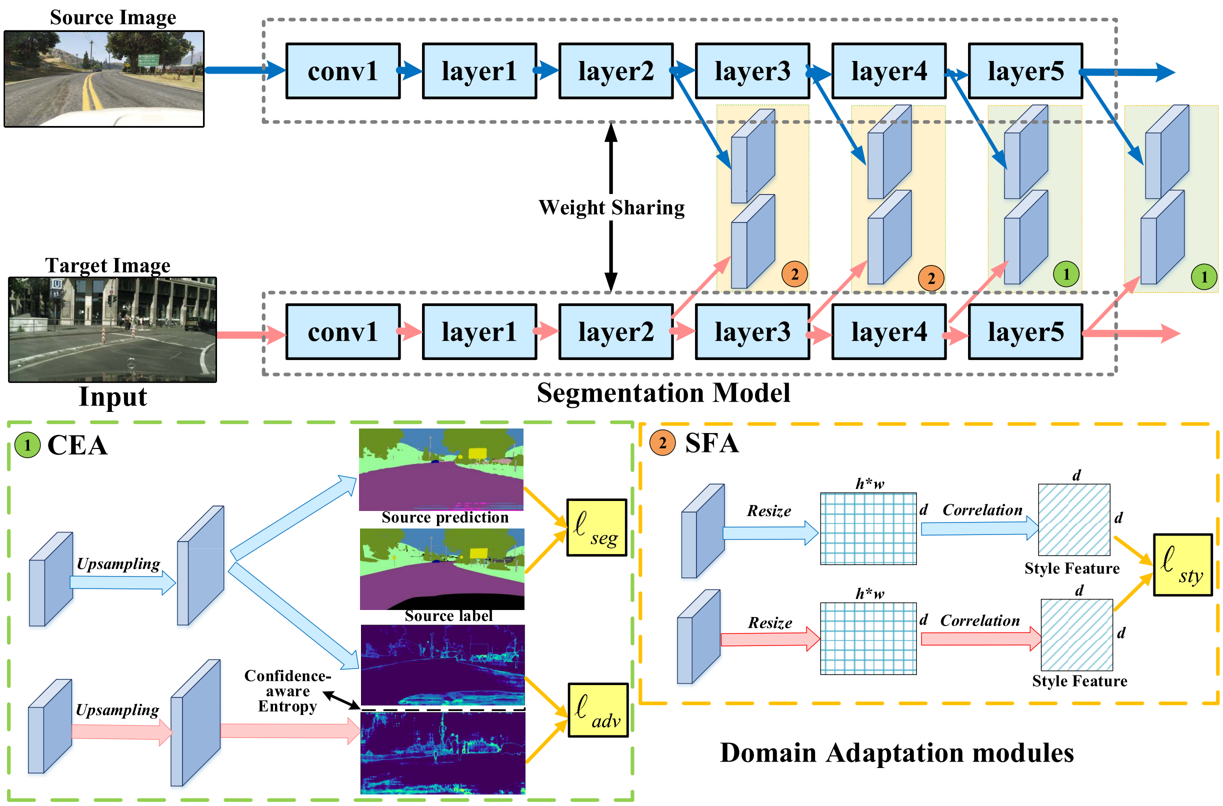


Fig. 2. Overview of our domain adaptation model CRAM for semantic segmentation. The blue arrow indicates the source domain, while the pink arrow represents the target domain. It contains two critic modules: the confidence-aware entropy alignment module(CEA) and the style feature alignment module(SFA). CEA addresses the domain gap in the structured output space while SFA further minimizes the appearance gap across domains in the shallower feature space. The blue arrow indicates the forward path of the source domain, and the pink arrow indicates the path of the target domain.

to alleviate the domain discrepancy is entropy minimization, through which the segmentation model $\mathcal{F}$ is forced to produce high-confidence (low-entropy) predictions on the target domain. In practice, the entropy minimization (EM) has been successfully applied in the UDA methods. For the particular task of semantic segmentation, [9] and [23] tackle with the pixel-level domain adaptation task in the output space with the EM technique. In detail, both methods develop an adversarial training framework, for the propose of forcing the entropy distribution of the target image similar to that of the source image.

Thus, it inspires us to perform entropy minimization via adversarial mechanism to address the UDA task for two reasons. First, it is observed that the pixel-level output of the segmentation model is rich in spatial and local information. Therefore, by aligning the distributions of entropy of target and source domains via adversarial learning, the domain gap is bridged in the output space under the constraint of structural consistency. Second, through adaptation in the weighted self-information space, the segmentation model is more likely to produce high-confident predictions on target images.

2) Ours: In the context of semantic segmentation, the normalized Shannon entropy E is widely used. For clustering, another form of entropy is proposed by [70] to improve cluster purity. With a novel target distribution introduced, the model can pay more attention to high-confident data points.

Motivated by this, we argue that the new entropy could be useful when applied on domain adaptation for segmentation task. To this end, unlike the prior works, a novel confidence-aware entropy (CE) of the target image is proposed in the CEA module and can be defined as:

$$C_T^{(h,w)} = \frac{-1}{\log(C)} \sum_c Q_T^{(h,w,c)} \log P_T^{(h,w,c)}, \tag{4}$$

where Q_T is an auxiliary target distribution derived from P_T. Explicitly, $Q_T^{(h,w,c)}$ is obtained through normalizing the square of $P_T^{(h,w,c)}$ by frequency per segmentation and can be written as:

$$Q_T^{(h,w,c)} = \frac{\left(P_T^{(h,w,c)}\right)^2 / f_c}{\sum_{c'} \left(P_T^{(h,w,c')}\right)^2 / f_{c'}}, \tag{5}$$

where $f_c = \sum_{h,w} P_T^{(h,w,c)}$ are soft-segmentation frequencies.

Similarly, the confidence-aware entropy of the source domain is defined as:

$$C_S^{(h,w)} = \frac{-1}{\log(C)} \sum_c Q_S^{(h,w,c)} \log P_S^{(h,w,c)}, \tag{6}$$

where the calculation of $Q_S^{(h,w,c)}$ can refer to that of $Q_T^{(h,w,c)}$.

It is worth noting that the proposed CE is quite different from the entropy used in clustering. First, for clustering, the soft assignments are computed by means of the Student's t-distribution as approximate probability estimates. In contrast, we directly treat the outputs of the segmentation model as soft assignments. Furthermore, the clustering task applies the target

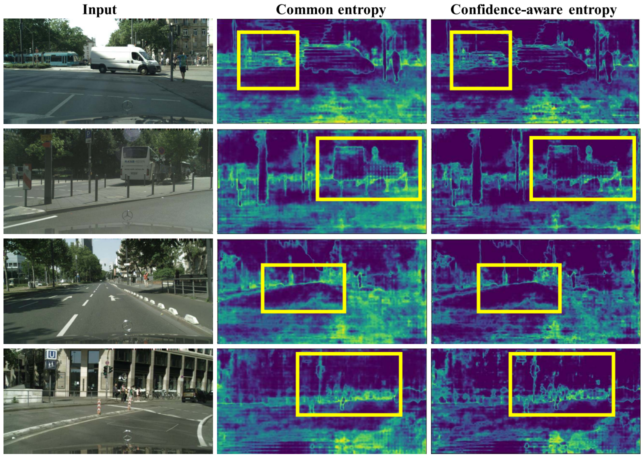


Fig. 3. Visualization of different entropy maps of semantic outputs. Compared with the common entropy applied in the existing UDA methods, that is, the normalized Shannon Entropy, the proposed CE strengthens semantic outputs. For example, for the same output, CE puts emphasis on the predictions of the main objects, such as the cars and the road. In addition, the entropy maps contain a lot of noise. In contrast, the visual result of CE is clearer.

distribution to calculate the KL divergence, and minimizes the KL divergence to make the soft assignment close to the target distribution. However, our method utilizes the target distribution to modify the normalized Shannon Entropy which is rigorously related to the cross entropy. As far as we know, we are first to apply the modified entropy in the UDA task for semantic segmentation successfully.

The visualization results of the basic entropy and the proposed confidence-aware entropy are illustrated in Figure 3. Compared with the common entropy applied in the prior UDA methods, the proposed CE really strengthens semantic outputs. For example, for the same output, CE puts emphasis on the predictions of the main objects, such as the cars and the road. In addition, the entropy maps contain a lot of noise. In contrast, the visual result of CE is clearer. To conclude, CE can better capture the main information of the semantic outputs.

In practice, when the adaptation is done in the modified entropy space, our UDA method can yield more precise segmentation on the target domain than the model with common entropy, with speed efficient. This is may due to several properties of Q. First, through raising the original distribution $P^{(h,w)}$ to the second power distribution, the segmentation predictions are strengthened, forcing the segmentation model to shift to the high-confident predictions. Furthermore, the normalized distribution can prevent larger segmentation categories deforming the hidden feature space. Also, the modified entropy adds no significant overhead to the UDA model for semantic segmentation.

To conduct adversarial adaptation, CE is fed to the discriminator $\mathcal{D}$. The cross-entropy loss $\mathcal{L}_d$ is applied to train $\mathcal{D}$ and as follows,

$$\mathcal{L}_d(C) = -(\sum_{h,w} \log\left(\mathbf{D}(C_T)^{(h,w,0)}\right) + \log\left(\mathbf{D}(C_S)^{(h,w,1)}\right)). \quad (7)$$

While for the segmentation model, the adversarial loss $\mathcal{L}_{\text{adv}}$ can be defined as:

$$\mathcal{L}_{adv}(C_T) = -\sum_{h,w} \log\left(\mathbf{D}(C_T)^{(h,w,1)}\right). \quad (8)$$

Thus the total loss for training segmentation model through the CEA module is defined as:

$$\mathcal{L}_{\text{CEA}} = \mathcal{L}_{seg}\left(I_S, Y_S\right) + w_{adv}\mathcal{L}_{adv}\left(C_T\right), \quad (9)$$

where w_{adv} is the weighting factor of the adversarial term $\mathcal{L}_{adv}$.

B. Style Feature Alignment

To further enhance the segmentation model transfer, the SAE module is proposed to minimize the appearance gap in low-level feature level.

1) Preliminary: Let $V \subset \mathbb{R}^{H\times W\times c}$ denote the representation of the input image. Through the flatten operation, we can first transform the matrix V into $V_1 \subset \mathbb{R}^{s\times c}$, where $s = H \times W$. The Gram matrix can be obtained by calculating the inner product of V_1 and its transpose matrix.

According to [71], a somewhat obvious, yet significant observation is that the Gram matrix includes the correlations of multi-layer representations, thus helping capture texture information. In [71], the Gram matrix is utilized to transfer the style of the original image to the white noise image.

Motivated by this, we construct a style feature (SF) through computing the Gram matrix of the low-level representations, which serves as a representation of the image style. When adaptation is conducted in the style feature space, the style of source images is successfully transferred onto target images.

2) Ours: To perform the style adaptation, we directly minimize the distance between the style features of source images and that of target images. On one hand, adaptation in the style feature space contributes to minimization of the appearance gap across domains. On the other hand, this UDA strategy can be treated as a form of style transfer. Compared with the traditional UDA methods which perform image-to-image translation, aligning the style features across domains is a simple yet effective approach to reduce the data bias globally.

With reference to the form of Gram matrix, the style feature S computed by the low-level feature F is defined as:

$$S^{(h,w)} = \sum_k V^{(h,k)} V^{(w,k)} \quad (10)$$

Through inner product between the different feature maps, the style features can capture the texture information of the input image.

Let S^l_S denote the style feature in layer l with the corresponding immediate feature $V^l \in \mathcal{R}^{N_l \times M_l}$ of a source image. Similarly, let S^l_T denote the style feature in layer l on the source domain.

To carry out domain adaptation, we minimize the mean-squared distance between the style features of two domains. The style loss in layer l is:

$$\mathcal{L}^l_{sty}(S^l_S, S^l_T) = \frac{1}{4N_l^2 M_l^2} d_2(S^l_S, S^l_T)^2, \quad (11)$$

where $d_2(\mathbf{A}, \mathbf{B}) = \sqrt{\sum_{i=1}^{n} \sum_{j=1}^{n} \left(a_{ij} - b_{ij}\right)^2}$. Therefore, the total style loss is formulated as follows:

$$\mathcal{L}_{sty} = \sum_{l=0}^{L} w_l \mathcal{L}_{\text{sty}}^{l}(S_S^l, S_T^l), \tag{12}$$

with w_l as weighting factors of the influence of layer l on the total style loss.

In experiments, for style transfer, features of both layer2 and layer3 are selected to compute the style features. For the purpose of style transfer in multi-level feature space, the feature correlations of multiple layers are included in the SFA module. By including the multi-scale representation of the input image, the SFA module can match source and target distributions in terms of multi-level texture information. The selection of features applied for the style features will be explained in our ablation study detailedly (section IV).

Differ from [71] that treats the whole white noise image as an optimization object, the style loss in our work is designed to guide the training of the segmentation model. In [71], the Gram matrix is introduced to induce the style of the iterable white noise image to be consistent with that of the given picture. While in our network, style features are used to guide the training of the segmentation network.

Furthermore, these UDA methods based on style transfer always depend on adversarial mechanism. On the contrary, our method simply minimizes the distance of the style feature between source and target domains so that the segmentation model can be trained in an one-stage end-to-end way. For one thing, our method avoids the troublesome training of adversarial networks which is complex and unstable. For another, the SFA module adds no significant overhead to the already designed network.

C. Confidence-and-Refinement Adaptation for Semantic Segmentation

The architecture of CRAM is illustrated in Figure 2. It contains two UDA modules for semantic segmentation. Firstly, for adapting structured output space, the CEA module forces the segmentation model to pay more attention to high-confident predictions. Secondly, in order to further enhance the adaptation, the SFA module closes the appearance gap between synthetic and real images in the shallower feature level.

In our work, we utilize ResNet-101 [79] and VGG-16 [80] as the backbone network of semantic segmentation.

In the forward path, the source image I_S and the target image I_T are fed into the segmentation model $\mathcal{F}$. Then the output P_S and P_T are generated.

For CEA, on VGG-16, we only utilize the feature at layer5 as the output P. Similar to the settings in [9], on ResNet-101, the feature from layer4 is also chosen to perform the adaptation. With P, CE can be further calculated. This multi-level adversarial mechanism takes advantage of richer information, both spatially and locally.

At the same time, in the SFA module, the representations extracted from layer2 and layer3 from the source and target domains are utilized to compute the style features S_S^l and S_T^l.

Our CRAM enables one-stage end-to-end training. Considering the above two adaptation modules, the total loss for training the segmentation model $\mathcal{F}$ is:

$$\mathcal{L}_{total} = \mathcal{L}_{CEA} + \lambda \mathcal{L}_{sty}, \tag{13}$$

where λ is the weight employed to balance the losses of two domain adaptation modules. During the training phase, for each training batch, C_T is first passed to the discriminator $\mathcal{D}$. Now we can compute the total loss $\mathcal{L}_{\text{total}}$ to optimize $\mathcal{F}$ with the parameter of the discriminator fixed. Afterwards, we fix the parameter of the segmentation network, and forward C_S to optimize the discriminator $\mathcal{D}$ with Equation (7). C_T is also passed to the discriminator $\mathcal{D}$ for training. The whole training process is shown in Alg. 1.

Algorithm 1 The Training Scheme of the Proposed Method

```
Require:
    Input: training images and labels $(I_S, Y_S, I_T)$, the number of training iteration $T$;
    Initialize the parameters of the segmentation model $\mathcal{F}$ with the pretrained model on ImageNet [81];
    Randomly initialize the parameters of discriminators $\mathcal{D}$, set $t = 0$;
    for all $t < T$ do
        Randomly choose a batch of target images, a batch of source images and their corresponding labels;
        Fix the parameters of the segmentation model $\mathcal{F}$, train the two discriminator networks $\mathcal{D}$ with Eq 7. (// Training the Discriminator Networks in the CEA module);
        Fix the parameters of the two discriminator networks in CEA, train the segmentation model $\mathcal{F}$ with the segmentation loss as shown in Eq 3. (// Training the segmentation model with the supervised segmentation loss);
        Train the segmentation model $\mathcal{F}$ with the adversarial loss in Eq 8. (// Training the Segmentation model with CEA);
        Train the segmentation model $\mathcal{F}$ with the style loss in Eq 12. (// Training the Segmentation model with SFA);
        $t = t + 1$;
    end for
Ensure:
    Output: $F_T$
```

It is worth noting that we did not construct two corresponding segmentation networks for the source and target domains. For each batch, the source image and the target image are passed into the same segmentation network. During the inference phase, this segmentation network directly processes the input image and outputs the corresponding semantic prediction.

IV. Experiments

A. Experimental Setup

1) Datasets: In our experiments, we treat the synthetic datasets including GTAV [12] and SYNTHIA [13] as source domains and the real-world dataset Cityscapes [82] as the target domain.

TABLE I

RESULTS OF ADAPTING GTA5 TO CITYSCAPES. THE 19 CLASS MIOU (%) IS USED AS THE EVALUATION MATRIC OF SEMANTIC SEGMENTATION PERFORMANCE

Backbone	Method	road	sdwk	bldng	wall	fence	pole	light	sign	vgttn	trrn	sky	person	rider	car	truck	bus	train	mcycl	bcycl	mIoU
ResNet101	Without adaptation [19]	75.8	16.8	77.2	12.5	21.0	25.5	30.1	20.1	81.3	24.6	70.3	53.8	26.4	49.9	17.2	25.9	6.5	25.3	36.0	36.6
	DAKD [60]	89.1	44.8	74.2	20.0	11.4	22.1	19.2	11.8	73.6	29.6	59.9	43.6	7.6	79.1	18.1	22.1	8.2	6.5	1.4	33.8
	KDN [72]	80.6	27.6	75.2	21.8	12.0	16.2	16.1	8.1	77.3	30.9	73.9	40.6	2.6	76.8	24.1	31.4	0.0	14.0	2.4	33.2
	AdaptSegNet [21]	88.9	26.1	80.1	22.2	21.9	27.6	34.2	21.1	83.1	31.5	76.2	56.1	28.2	82.1	27.1	35.0	2.0	26.3	28.6	42.0
	CyCADA [8]	86.7	35.6	80.1	19.8	17.5	**38.0**	**39.0**	**41.5**	82.7	27.9	73.6	**64.9**	19.0	65.0	12.0	28.6	4.5	31.1	42.0	42.7
	MinEnt [9]	82.5	23.4	72.6	17.5	21.9	29.8	33.0	21.4	83.0	30.4	75.8	58.8	28.6	79.4	32.9	27.3	0.0	29.9	**43.7**	41.7
	AdvEnt [9]	87.7	20.7	80.8	20.5	24.5	31.1	33.4	18.8	82.6	27.3	74.5	58.8	27.4	83.1	28.5	36.6	7.9	30.6	29.7	42.4
	CLAN [73]	87.0	27.1	79.6	27.3	23.3	28.3	35.5	24.2	83.6	27.4	74.2	58.6	28.0	76.2	33.1	36.7	6.7	**31.9**	31.4	43.2
	SWD [74]	**92.0**	46.4	82.4	24.8	24.0	35.1	33.4	34.2	83.6	30.4	80.9	56.9	21.9	82.0	24.4	28.7	6.1	25.0	33.6	44.5
	FDA [16]	90.0	40.5	79.4	25.3	26.7	30.6	31.9	**29.3**	79.4	28.8	76.5	56.4	27.5	81.7	27.7	45.1	**17.0**	23.8	29.6	44.6
	Intra [23]	90.2	35.8	82.2	25.3	24.1	28.6	30.3	21.8	84.8	37.3	80.5	57.9	27.2	85.7	**35.5**	49.2	0.0	28.7	35.0	45.3
	BDL [15]	90.2	**46.8**	84.3	31.7	29.0	32.1	38.9	31.6	84.4	41.3	80.1	58.2	30.0	83.3	28.3	43.7	1.9	26.8	38.6	47.4
	Ours	91.7	18.6	**90.0**	33.3	23.1	27.0	32.3	27.1	83.0	25.4	79.0	58.4	28.4	87.7	19.5	52.3	9.1	28.2	30.8	44.5
	Ours(+ST)	90.7	36.7	81.7	26.8	25.9	27.4	34.7	22.7	**85.6**	40.1	80.3	59.7	31.8	**91.8**	34.0	**56.7**	4.1	31.0	37.6	47.3
	Ours(+BDL)	90.7	45.9	84.5	**34.7**	**29.2**	31.9	37.6	33.1	84.4	**42.6**	**85.2**	58.1	**32.5**	83.0	34.7	50.1	4.4	29.5	30.7	**48.6**
VGG16	Without adaptation [19]	70.4	32.4	62.1	14.9	5.4	10.9	14.2	2.7	79.2	21.3	64.6	44.1	4.2	70.4	8.0	7.3	0.0	3.5	0.0	27.1
	CDA [75]	72.9	30.0	74.9	12.1	13.2	15.3	16.8	14.1	79.3	14.5	75.5	35.7	10.0	62.1	20.6	19.0	0.0	19.3	12.0	31.4
	CyCADA [8]	83.5	38.3	76.4	20.6	16.5	22.2	26.2	21.9	80.4	28.7	65.7	49.4	4.2	74.6	16.0	26.6	2.0	8.0	0.0	34.8
	Adapt-SegMap [21]	86.2	25.0	78.1	20.5	18.0	17.5	17.2	9.7	79.6	29.2	69.4	43.9	3.8	77.6	**27.3**	31.8	0.0	11.2	0.5	34.0
	MinEnt [9]	85.6	17.3	75.3	30.4	17.7	18.9	17.8	7.7	77.9	22.5	61.9	41.3	8.5	80.2	18.9	12.9	0.0	13.0	0.5	32.0
	AdvEnt [9]	88.5	30.6	78.2	27.6	17.5	17.2	14.0	7.0	81.2	30.3	68.8	43.8	9.3	79.4	19.8	13.8	2.4	7.0	0.7	33.5
	CBST [22]	84.2	41.4	71.9	15.5	18.1	**30.8**	25.4	9.2	77.6	15.2	29.6	49.3	6.0	78.0	4.0	4.5	0.3	10.4	11.6	30.7
	CRST(MRKLD) [76]	81.7	**46.1**	70.2	10.7	11.2	30.4	26.9	15.8	75.4	18.3	24.8	48.6	10.9	77.8	2.9	13.3	1.1	10.7	**31.4**	32.0
	FDA[16]	85.6	33.7	80.0	27.5	18.3	25.6	27.5	**22.4**	81.2	29.5	73.7	50.9	21.3	81.3	22.7	28.0	**22.3**	15.2	24.7	40.6
	Intra [23]	89.4	33.0	78.0	25.7	14.6	15.9	12.5	6.8	81.2	30.7	70.1	44.5	8.5	78.8	20.8	29.1	0.0	7.6	0.6	34.1
	BDL [15]	89.2	40.9	81.2	29.1	19.2	14.2	29.0	19.6	83.7	35.9	80.7	54.7	23.3	**82.7**	25.8	28.0	2.3	25.7	19.9	41.3
	Ours	88.8	33.6	77.8	**31.6**	20.6	19.6	18.6	8.6	79.8	31.4	74.1	48.2	3.2	80.4	26.9	28.9	14.4	2.4	0.5	36.2
	Ours(+ST)	**89.4**	33.0	78.0	25.7	14.6	15.9	12.5	6.8	81.2	30.7	70.1	44.5	8.5	78.8	20.8	29.1	0.0	7.6	0.6	34.1
	Ours(+BDL)	89.1	42.0	**81.2**	28.9	**23.1**	13.9	29.3	17.0	**83.6**	**36.6**	**81.7**	**56.2**	**25.5**	81.9	26.0	**32.0**	0.2	**26.8**	19.7	**41.8**

- **GTA5** [12] contains 24,966 synthesized images extracted from the 3D computer game Grand Theft Auto V based on the urban scenery of Los Angeles city. The pixel-accurate semantic annotations are generated automatically. Each image has a resolution of 1914 × 1052 pixels. During the training phase, we use the 19 common categories with the Cityscapes dateset to evaluate the performance.
- **SYNTHIA** [13] refers to the SYNTHIA-RAND-CITYSCAPES set. It is a synthetic dataset composed of 9,400 images based on rendering and corresponding ground-truth semantic labels. Each image has a resolution of 1280 × 960 pixels. During the training time, we consider the 16 common categories with the Cityscapes dataset to evaluate the performance.
- **Cityscapes** [82] is a real-world dataset collected from 50 cities in Germany which includes 5000 annotated images with fine annotations. In our experiment, following the prior works, we take 2,975 images as the training set and 500 images as the validation set. During the training phase, the semantic labels are excluded and not used. Each image has a resolution of 2048 × 1024 pixels. In the "GTA5-to-Cityscapes" benchmark, the 19 class mIoU (%) is used as the evaluation matric of semantic segmentation performance. In the "SYNTHIA-to-Cityscapes" benchmark, we introduce the 16 class mIoU (%) as the evaluation matric. We also choose 13 common classes between SYNTHIA and CITYSCAPES as our valid labels, following the same evaluation protocol as other works [16].

2) Network Architecture: In CRAM, DeepLab-v2 [25] with ResNet-101 [79] and VGG-16 [80] serve as the segmentation model $\mathcal{F}$. These models are pretrained on ImageNet [81].

For the CEA module, similar to [9], to improve the adaptation, we conduct the multi-level adversarial strategy on ResNet-101. In detail, features from both Conv4 and Conv5 are chosen as the outputs. While on VGG-16, we directly utilize single-level adaptation. For the discriminators, we adopt the architecture from the previous work [9], [21]. The confident-aware entropy of the outputs are passed into five 4 × 4 convolution layers with stride 2. Channel numbers of these convolution layers are 64,128, 256, 512, 1. Except for the last layer, a leaky ReLU parameterized by 0.2 is added after each convolutional layer to further process features.

3) Training: The architecture of our model is implemented with the PyTorch library [83]. The segmentation model $\mathcal{F}$ is trained using the SGD method with momentum 0.9. The initial learning rate is 2.5×10^{-4}. To optimize the discriminators $\mathcal{D}$, we apply Adam [84] optimizer ($\beta_1 = 0.9, \beta_2 = 0.99$) with learning rate 1×10^{-4}. For the learning rate schedule, we adopt the polynomial policy according to [9] for both $\mathcal{F}$ and $\mathcal{D}$. During training, images were resized to the resolution of 1024 × 512. The batch size is set to 1 on a single NVIDIA 1080TI GPU with 11 GB memory. We select the best model for validation within 120,000 training iterations. The entire training process takes approximately 42 hours.

In all experiments, we set $w_{adv} = 0.001$ in Equation (9), $w_l = 0.5$ in Equation (12), and $\lambda = 0.1$ in Equation (13).

4) Evaluation Matric: In our experiments, class mIoU (%) is used as the evaluation matric to measure the performance of semantic segmentation, which is defined as:

$$mIoU = (1/n)\sum_i n_{ii} / \left(\sum_j n_{ij} + \sum_j n_{ji} - n_{ii}\right), \quad (14)$$

where n is the number of classes. n_{ij} is the number of pixels of class i that are judged to be class j.

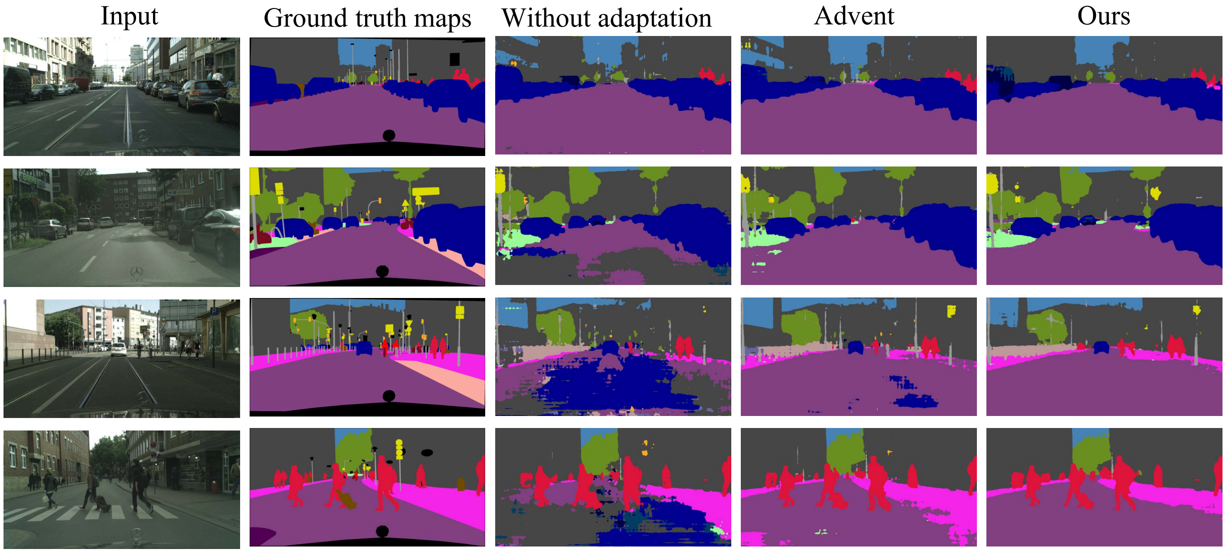


Fig. 4. Qualitative results in the GTA5-to-Cityscapes set-up.

B. Results

1) GTA5-to-Cityscapes: As shown in Table I, our method achieves competitive performance with state-of-the-art UDA methods [9], [76] on both VGG-16 and ResNet-101-based networks. With the ResNet-101 based model, our method significantly beats the accuracy of the basic segmentation model without adaptation by 7.9 absolute percentage points. And on VGG-16 based CNNs, the improvement is more, *i.e.*, +9.1 %. This phenomenon reveals that our two domain adaptation modules play an important role in minimizing the domain gap.

Compared with the method [9], [21], domain adaptation methods in the feature space, our method can obtain better results.

Compared with the UDA models [16] based on style transfer, our method can obtain similar performance. However, these methods always need complex image processing and fine-tuning with self-training. In contrast, our method is more effective yet considerably simpler. These UDA methods [15], [23] apply pseudo labels to obtain competitive performance. We modify and extend the architecture of these methods with our proposed UDA modules. As demonstrated in Table I, the modified networks, which correspond to **ours(+ST)** and **ours(+BDL)**, yield more precise segmentations than the original methods [15], [23]. This further verifies the adaptability and effectiveness of our UDA approach. The proposed two UDA modules can be directly applied to the framework of existing UDA methods to further enhance the generalization of the segmentation model. In addition, with the assistance of self-training based methods [15], [23], the performance of our approaches has been further improved. One exception is that when using VGG as the backbone, the mIoU obtained trough combining our method with [23] is 34.1%, which is lower than the result of our method without self-training, which is 36.2%. This may be due to the instability of [23]. In the self-training phase, [23] first divides the target dataset into two types of data: hard data and easy data. Then it realizes the intra-domain adaptation by adapting the segmentation model from easy data to hard data. However, the division of the target dataset depends on the training model in the previous stage. In addition, only high-confidence pseudo-labels are selected for easy data, so that some categories may be ignored and not participate in the self-training stage.

Qualitative results of our ResNet-101-based method are presented in Figure 4. Compared with the segmentation model without adaptation, our model yields better semantic predictions. It can be observed that without adaptation, the segmentation quality is always noisy on the target domain due to the domain gap. In contrast, our predictions are more precise. For example, for small objects such as traffic signs and edge of the road, our predictions are clearer. This shows that our method can address the severe domain mismatch effectively. Further, compared with AdvEnt which utilizes the normalized entropy, our method can also obtain better results. Compared with the common entropy, the modified entropy proposed in our paper better addresses the adaptation of semantic segmentation neural networks.

From the perspective of convergence speed, as presented in Figure 7, the loss of our ResNet-101-based model converges faster compared with other UDA methods for semantic segmentation. This proves that our network can achieve a faster convergence speed, especially when compared with the UDA method using common entropy.

2) SYNTHIA-to-Cityscapes: We present in Table II the segmentation performance on the benchmark "SYNTHIA-to-Cityscapes" for both ResNet101 and VGG16. Compared with "GTA5-to-Cityscapes", the cross-domain set-up "SYNTHIA-to-Cityscapes" has larger domain gap. The results for 13 and 16 categories are both listed. As shown in Table II, with the worse domain discrepancy, our method still gets 8.2 % improvement with ResNet101 and 15.8% improvement with VGG-16 compared with the segmentation model without adaptation. Compared with other competitive domain adaptation methods, our method achieves comparable performance.

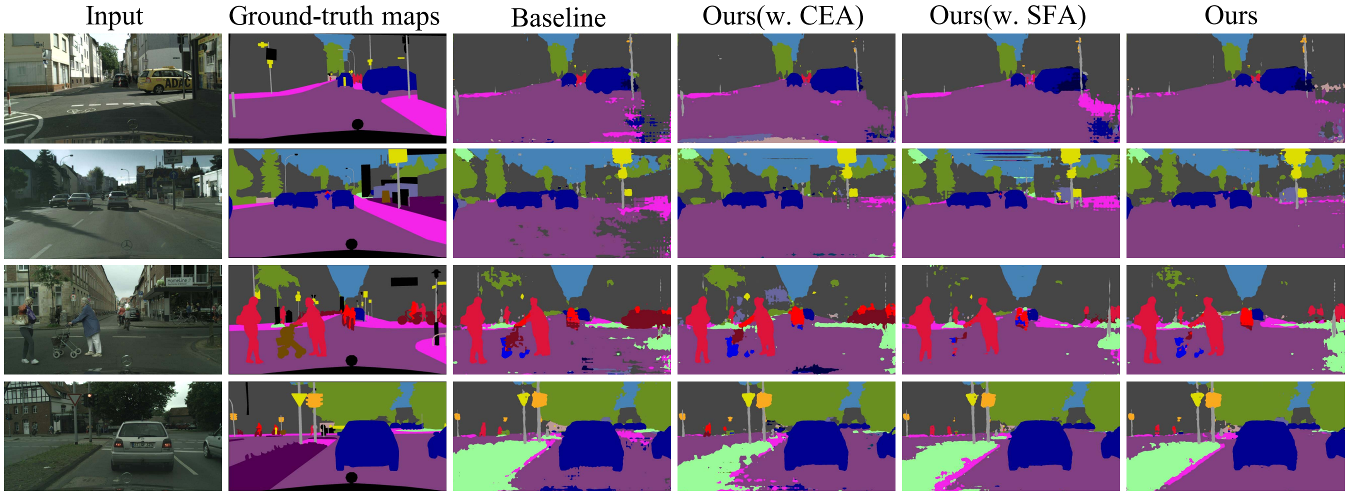


Fig. 5. Qualitative segmentation results of different network setting on the Cityscapes dataset.

TABLE II

RESULTS OF ADAPTING SYNTHIA TO CITYSCAPES. THE 16/13 CLASS MIOU (%) IS USED AS THE EVALUATION MATRIC OF SEMANTIC SEGMENTATION PERFORMANCE

Backbone	Method	road	sdwk	bldng	wall*	fence*	pole*	light	sign	vgttn	sky	person	rider	car	bus	mcycl	bcycl	mIoU	mIoU*
ResNet101	Without adaptation [19]	55.6	23.8	74.6	9.2	0.2	24.4	6.1	12.1	74.8	79.0	55.3	19.1	39.6	23.3	13.7	25.0	33.5	38.6
	KDN [72]	52.1	20.9	65.1	1.8	0.15	15.0	0.7	3.0	71.9	76.8	29.7	8.0	49.1	1.0	3.28	12.28	25.6	30.3
	AdaptSegNet [21]	77.5	33.5	78.6	5.4	0.5	22.3	2.8	8.3	79.1	82.7	41.1	16.3	69.4	29.5	14.7	27.9	36.8	43.2
	MinEnt [9]	73.5	29.2	77.1	7.7	0.2	27.0	7.1	11.4	76.7	82.1	57.2	21.3	69.4	29.2	12.9	27.9	38.1	44.2
	AdvEnt [9]	90.3	51.9	78.1	1.9	0.3	12.9	1.1	5.1	78.8	84.1	46.8	13.2	78.2	28.3	3.5	27.6	37.6	45.1
	SIBAN [77]	82.5	24.0	79.4	-	-	-	16.5	12.7	79.2	82.8	58.3	18.0	79.3	25.3	17.6	25.9	-	46.3
	AdaptPatch [78]	82.4	38.0	78.6	8.7	0.6	26.0	3.9	11.1	75.5	84.6	53.5	21.6	71.4	32.6	19.3	31.7	40.0	46.5
	FDA [16]	79.3	35.0	73.2	-	-	-	19.9	**24.0**	61.7	82.6	**61.4**	31.1	**83.9**	40.8	**38.4**	**51.1**	-	52.5
	Intra [23]	84.3	37.7	79.5	5.3	0.4	24.9	9.2	8.4	80.0	84.1	57.2	23.0	78.0	38.1	20.3	36.5	41.7	48.9
	BDL [15]	6.0	46.7	80.3	-	-	-	14.1	11.6	79.2	81.3	54.1	**27.9**	73.7	**42.2**	25.7	45.3	-	51.4
	Ours	88.4	48.5	76.7	9.3	**0.6**	18.9	17.2	14.6	77.8	82.3	41.7	16.1	74.3	30.6	10.9	30.0	40.0	46.8
	Ours(+ST)	**92.0**	**53.8**	80.7	2.0	0.1	21.0	0.9	6.5	81.2	84.6	51.6	20.2	82.6	38.9	22.9	42.4	42.6	50.6
	Ours(+BDL)	87.6	46.1	**82.0**	**10.0**	0.4	**33.6**	**21.4**	14.9	**81.2**	**85.2**	57.2	26.4	83.0	33.3	24.0	46.8	**45.8**	**53.0**
VGG16	Without adaptation [19]	11.5	19.6	30.8	-	-	-	0.1	11.7	42.3	68.7	51.2	3.8	54.0	3.2	0.2	0.6	-	22.9
	AdaptSegNet [21]	72.8	**37.9**	68.1	3.1	0.3	21.6	0.6	8.5	76.3	78.0	40.4	11.5	64.0	20.2	4.6	15.9	32.7	38.4
	MinEnt [9]	37.8	18.2	65.8	2.0	0.0	15.5	0.0	0.0	76	73.9	45.7	11.3	66.6	13.3	1.5	13.1	27.5	32.5
	AdvEnt [9]	67.9	29.4	71.9	6.3	0.3	19.9	0.6	2.6	74.9	74.9	35.4	9.6	67.8	21.4	4.1	15.5	31.4	36.6
	CDA [75]	57.4	23.1	74.7	0.5	0.6	14	5.3	4.3	77.8	73.7	45	11	44.8	21.2	1.9	20.3	29.7	35.0
	CBST [22]	75.7	32.3	70.2	3.5	0.0	28.6	1.4	9.0	79.8	65.6	52.9	13.7	65.8	9.1	1.5	36.4	34.1	39.5
	CRST(MRKLD) [76]	75.1	33.5	70.8	5.6	0.0	**28.7**	2.0	9.7	78.9	72.5	51.7	11.6	63.4	7.3	1.4	38.6	34.4	39.7
	FDA [16]	**84.2**	35.1	78.0	6.1	0.44	27.0	8.5	**22.1**	77.2	79.6	55.5	19.9	74.8	**24.9**	**14.3**	40.7	35.0	40.5
	Intra [23]	81.5	32.9	72.4	0.9	0.2	20.0	0.0	1.5	76.9	78.8	44.9	18.5	73.9	18.4	4.6	17.3	33.9	40.1
	BDL [15]	72.0	30.3	74.5	0.1	0.3	24.6	**10.2**	25.2	80.5	80.0	**54.7**	**23.2**	72.7	24.0	7.5	**44.9**	33.2	39.0
	Ours	73.1	28.0	74.4	4.0	0.1	22.3	0.0	2.4	76.0	74.2	44.5	16.6	68.5	18.5	4.8	22.7	33.1	38.7
	Ours(+ST)	78.0	31.0	74.9	2.9	0	22.4	0.0	1.6	76.0	75.0	46.6	16.3	71.7	16.7	3.3	25.0	33.8	39.7
	Ours(+BDL)	79.5	33.4	**79.2**	**7.2**	**0.5**	28.3	0.0	12.9	**81.7**	**80.6**	46.0	18.7	**76.2**	22.6	9.5	43.2	**38.7**	**45.5**

When integrated with the self-training based UDA methods [15], [23], the performance of our methods can be further improved. Additionally, our method also encourages these UDA approaches to promote the adaptation of semantic segmentation model from source domain to target domain, which validates the adaptability of the proposed modules on the SYNTHIA-to-Cityscapes benchmark.

C. Ablation Study

1) Baselines: We treat AdaptSegNet [21] as the baseline model, which directly aligns the outputs of the segmentation across domains via adversarial mechanism. Based on AdaptSegNet, we also conduct an experiment on two UDA benchmarks which aims to align the result of the entropy of outputs, which corresponds to Ours(w.EA) in Table III and Table IV. Besides, in the above two tables, Ours(w.CEA) refers to the model which performs adaptation in the CE space with adversarial training, and Ours(w.SFA) represents the baseline model with the SFA module added.

As presented in Table III and IV, compared with the baseline model, both CEA and SFA lead to the improved performance. On one hand, only with the CEA module, our method beats the accuracy of Ours(w.EA) in most settings. Compared with the common entropy, the proposed CE can better tackle the severe domain mismatch. Although in Table IV, the accuracy of Ours(w.EA) is higher than that of Ours(w.CEA) when ResNet is taken as the backbone, it takes much more time for the former to achieve the best performance. In the practical perspective, after the same 8000th iterations, the accuracy of Ours(w.EA) is 28.3%, which is much lower than our accuracy, which reaches 42.9%. This further validates that CEA can make the segmentation model converge faster. On the other hand, for the SFA module, our method significantly beats the

TABLE III

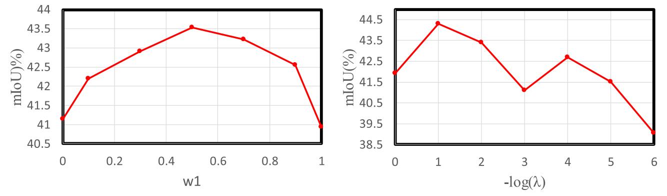
RESULTS OF ADAPTING GTA5 TO CITYSCAPES. THE 19 CLASS MIOU (%) IS USED AS THE EVALUATION MATRIC OF SEMANTIC SEGMENTATION PERFORMANCE

Backbone	Method	road	sdwk	bldng	wall	fence	pole	light	sign	vgttn	trrn	sky	person	rider	car	truck	bus	train	mcycl	bcycl	mIoU	best iter.
ResNet101	Baseline	88.9	**26.1**	80.1	22.2	21.9	27.6	**34.2**	21.1	**83.1**	31.5	76.2	56.1	28.2	82.1	27.1	35.0	2.0	26.3	28.6	42.0	96,000
	Ours(w.EA)	87.9	20.1	80.8	27.8	23.3	**30.1**	31.5	22.9	82.9	**31.9**	75.8	57.4	26.2	80.7	25.8	43.5	1.2	25.5	32.8	42.5	114,000
	Ours(w.CEA)	91.6	18.8	88.9	26.9	**25.4**	29.3	33.3	21.5	83.0	24.5	76.4	**59.2**	27.0	86.3	16.8	44.6	7.9	26.9	26.4	42.9	**58,000**
	Ours(w.SFA)	87.8	20.5	81.9	26.9	22.2	**28.8**	32.5	24.4	82.6	28.2	77.1	58.0	**28.8**	84.5	31.2	44.3	9.1	**31.9**	**34.1**	43.9	80,000
	Ours	**91.7**	18.6	**90.0**	**33.3**	23.1	27.0	32.3	**27.1**	83.0	25.4	**79.0**	58.4	28.4	**87.7**	**19.5**	**52.3**	**9.1**	28.2	30.8	**44.5**	70,000
VGG16	Baseline	87.0	28.3	77.4	30.2	19.1	**20.3**	16.5	7.5	79.1	22.5	66.2	39.3	8.3	79.3	19.7	17.1	0.0	**12.9**	**2.5**	33.3	62,000
	Ours(w.EA)	88.5	30.6	**78.2**	27.6	17.5	17.2	14.0	7.0	**81.2**	30.3	68.8	43.8	**9.3**	79.4	19.8	13.8	2.4	7.0	0.7	33.5	56,000
	Ours(w.CEA)	85.9	27.2	77.8	24.1	19.7	15.2	16.3	8.6	79.3	31.4	73.3	48.2	3.2	79.1	26.7	28.9	0.0	12.5	0.5	34.6	**20000**
	Ours(w.SFA)	87.6	27.0	76.5	24.8	19.0	16.5	**19.6**	7.4	80.0	27.9	70.8	45.0	5.2	80.2	23.1	14.8	2.4	8.9	0.5	34.5	72,000
	Ours	**88.8**	**33.6**	77.8	**31.6**	**20.6**	19.6	18.6	**8.6**	79.8	**31.4**	**74.1**	**48.2**	3.2	**80.4**	**26.9**	**28.9**	**14.4**	2.4	0.5	**36.2**	88,000

TABLE IV

THE RESULTS OF ABLATION STUDY ON THE NETWORK ARCHITECTURE. IN ORDER TO DETERMINE THE NUMBER OF ITERATIONS IN WHICH THE MODEL PERFORMS BEST DURING TRAINING, WE ADOPT THE METHOD SIMILAR TO [23]. EXPLICITLY, THE NUMBER OF ITERATIONS IS SET TO 120000, AND THE MODEL ARE SAVED EVERY 2000 ITERATIONS. THE BEST MODEL IS SELECTED AMONG THESE SAVED MODELS

Base Model	Method	road	sdwk	bldng	wall*	fence*	pole*	light	sign	vgttn	sky	person	rider	car	bus	mcycl	bcycl	mIoU	mIoU*	best iter.
ResNet101	Baseline	77.5	33.5	78.6	**5.4**	**0.5**	22.3	2.8	8.3	79.1	82.7	41.1	16.3	69.4	29.5	14.7	27.9	36.8	43.2	48,000
	Ours(w.EA)	77.2	33.2	77.8	5.1	0.4	24.7	4.1	10.1	78.5	81.9	47.2	19.3	73.3	28.6	10.2	35.0	37.9	44.3	10,2000
	Ours(w.CEA)	69.5	29.6	73.3	2.3	0.1	**26.7**	**13.1**	**11.8**	78.0	78.9	46.4	**20.2**	58.0	31.1	7.0	**40.1**	36.6	42.9	**8,000**
	Ours(w.SFA)	88.4	48.5	76.7	2.1	0.2	18.9	1.0	5.0	77.8	82.3	41.7	16.1	74.3	30.6	10.9	30.0	37.8	44.9	94,000
	Ours	**90.5**	**51.9**	**80.2**	2.7	0.1	23.1	2.2	7.3	**80.0**	**84.6**	**48.7**	15.8	**79.8**	**32.6**	**16.6**	32.4	**40.5**	**47.9**	100,000
VGG16	Baseline	72.8	**37.9**	68.1	3.1	0.3	21.6	**0.6**	8.5	76.3	78.0	40.4	11.5	64.0	**20.2**	4.6	15.9	32.7	38.4	86,000
	Ours(w.EA)	53.8	20.9	71.5	**4.1**	0.1	21.0	0.0	6.4	75.7	76.9	40.5	15.4	68.1	20.1	5.1	21.0	31.3	36.6	84,000
	Ours(w.CEA)	52.1	21.2	69.4	2.0	**0.5**	22.6	0.4	**9.4**	**77.6**	**79.0**	43.2	13.7	67.8	17.6	**5.8**	**24.0**	31.6	37.0	26,000
	Ours(w.SFA)	**77.3**	29.4	73.9	0.3	0.2	**23.2**	0.0	1.4	75.8	73.6	**45.4**	**17.2**	65.0	16.8	3.0	18.1	32.5	38.2	**20,000**
	Ours	73.1	28.0	**74.4**	4.0	0.1	22.3	0.0	2.4	76.0	74.2	44.5	16.6	**68.5**	18.5	4.8	22.7	**33.1**	**38.7**	38,000



(a) Style Loss **(b)Total Loss**

Fig. 6. The results of ablation study for hyper-parameters. (a) presents the result of the experiment on w_l. Since we only select two layers (Conv2 and Conv3), we choose the proportion of the style loss in the Conv2 layer to the total style loss as w_1, treated as the abscissa of (a). So the corresponding weighting factor of Conv3 is: $w_2 = 1 - w_1$. (b) refers to the study on λ.

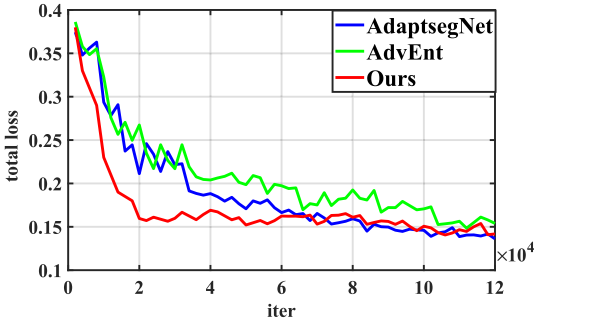


Fig. 7. Compared with other UDA methods in convergence speed. Obviously, the loss of our ResNet-101-based model converges faster compared with other UDA methods for semantic segmentation.

accuracy of AdaptSegNet. This proves the effectiveness of using style features for model transfer. In contrast with CEA, SFA contributes more to the improvement of the model.

The qualitative results is shown in Figure 5. It is clear that both domain adaptation modules contribute to the improvement of segmentation quality. Compared with CEA, SFA yields better performance, which is consistent with the conclusion from Table IV and III above. Notably, with CEA, the model also captures the information of some small objects such as the car steering wheel, which is neglected by other models.

In addition, our models both obtain better results than the baseline models whether the backbone is Resnet-101 or VGG16. Our two domain adaptation modules can operate on different segmentation models.

2) Hyper-Parameters: In the following experiments, performance of the models is presented in Figure 6. We select the ResNet101 based CNN as the basic segmentation network.

For CEA, following the prior works, we set $w_{adv} = 0.001$ in Equation (9). Next, we describe how to choose w_l of layer l for the style loss. The value of λ is arbitrarily set to 10^{-4}. First, features of layer4 and layer5 are chosen to compute the style feature. However, when optimizing the segmentation model with the total loss, loss divergence appears. This could be caused by conflicts of CEA and SFA when both modules perform adaptation in the output space. Then, we select the intermediate layers layer2 and layer3, which contain rich texture information.

As presented in Figure 6, when $w_l = 0.5$ and $w_2 = 0.5$, the segmentation performance mIoU can reach 43.53%, which is the best performance. This phenomenon reveals that setting the style loss of layer2 equal to that of layer3 is the optimal choice.

When fixing $w_l = 0.5$, the performance of CRAM excels at $\lambda = 0.1$, which demonstrates that loss of CEA set ten times to that of SFA can reach the best balance. In the Cityscapes validation dateset, the segmentation accuracy mIOU is 44.30%.

V. Conclusion and Future Work

In this paper, we propose a Confidence-and-Refinement adaptation model (CRAM) to deal with the UDA task for semantic segmentation. In CRAM, we conduct a multi-level adaptation strategy for model transfer. Specifically, two simple yet effective UDA modules are designed. First, the CFA model conducts adaptation in the structured output space, making the segmentation model put emphasis on the high-confident model. Second, the SFA module enhances the adaptation through aligning the style features across domains to minimize the appearance gap. Both domain adaptation modules contribute to the improvement of segmentation quality, which be treated as transferable modules applied to other UDA approaches. Furthermore, our model achieves promising segmentation performance on two challenging "synthetic-2-real" benchmarks, which demonstrates the effectiveness of the proposed UDA modules.

Despite the effectiveness of our method, there are still some technical limitations to the algorithm. We observe that the proposed confidence-aware entropy module does not always succeed in the UDA task for semantic segmentation. As presented in Table IV, the method using the common entropy yields better segmentation performance on "synthetic-2-real" benchmark when resnet101 is taken as the backbone. Probably the limiting factor of the proposed CEA module is the initialization mechanism. Through introducing the second power distribution in the target distribution Q, the segmentation model can focus on the high-confident predictions. However, dependence on the initialized model is also strengthened in that the target distribution is calculated with the segmentation output. Without a proper initialization model, in the case of unsupervised training, the performance of the model tend to be unstable. In the paper, the segmentation model is pretrained on ImageNet [81]. However, the characteristics of ImageNet are quite different from that of the datasets used in the UDA task for segmentation. It is likely that future improvement of the initialization strategy will increase the performance of the CEA module.

Although the feature adaptation-based methods can promote the adaptation of semantic segmentation model, they can not guarantee that the fine-grained feature alignment can be conducted in a class-wise manner. Recently, the centroid-aware methods [47], [54], [58], [59] have gained popularity in the area of domain adaptive Semantic Segmentation. For the convenience of separating different categories in the target domain, these UDA approaches turn to reduce the distance between the corresponding classes of two domains. Thus it could be possible to boost the performance of our model with the assistance of the centroid-aware techniques. We hope that the model can be encouraged to align features at a fine-grained level through introducing the centroid-aware techniques.

In the field of UDA for semantic segmentation, pseudo labels are always exploited to guide the retraining of the network. However, for most self-training methods, the generated pseudo labels are inevitably noisy. Recent work [54] is committed to online correction of the false pseudo labels by means of the prototypical strategy, which is proven effective. Apart from the self-training methods applied in this paper, it may be a good choice to facilitate our framework with the novel self-supervised learning strategy.

In addition to the centroid-aware methods, there are also many trivial solutions for domain adaptation, e.g., Knowledge Distillation [54], [60] and Mixing [59], [61]. These popular methods indeed boost the performance to a record high, yet the training process is prone to be time-consuming and requires much computational resources, making unsupervised domain adaptation impractical in industrialized scenarios. In light of the key role of semantic segmentation in autonomous driving, a new UDA solution is urgently needed, which can yield unprecedented state-of-the-art performance both in accuracy and efficiency. This also formulates an innovative and promising research direction for our future work.

References

[1] L. Zhou, H. Zhang, Y. Long, L. Shao, and J. Yang, "Depth embedded recurrent predictive parsing network for video scenes," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 12, pp. 4643–4654, Dec. 2019.

[2] V. Badrinarayanan, A. Kendall, and R. Cipolla, "SegNet: A deep convolutional encoder-decoder architecture for scene segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 12, pp. 2481–2495, Dec. 2017.

[3] X. Zhang, Y. Chen, H. Zhang, S. Wang, J. Lu, and J. Yang, "When visual disparity generation meets semantic segmentation: A mutual encouragement approach," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 3, pp. 1853–1867, Mar. 2021.

[4] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. CVPR*, 2015, pp. 3431–3440.

[5] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. Med. Image Comput. Comput.-Assist. Intervent.*, 2015, pp. 234–241.

[6] S. Liu, H. Zhang, L. Shao, and J. Yang, "Built-in depth-semantic coupled encoding for scene parsing, vehicle detection, and road segmentation," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 9, pp. 5520–5534, Sep. 2021.

[7] Y. Chen, W. Li, and L. Van Gool, "Road: Reality oriented adaptation for semantic segmentation of urban scenes," in *Proc. CVPR*, 2018, pp. 7892–7901.

[8] J. Hoffman *et al.*, "CyCADA: Cycle-consistent adversarial domain adaptation," in *Proc. ICML*, 2018, pp. 1989–1998.

[9] T.-H. Vu, H. Jain, M. Bucher, M. Cord, and P. Pérez, "ADVENT: Adversarial entropy minimization for domain adaptation in semantic segmentation," in *Proc. CVPR*, 2019, pp. 2517–2526.

[10] S. Zhao *et al.*, "Multi-source domain adaptation for semantic segmentation," in *Proc. NeurIPS*, 2019, pp. 7287–7300.

[11] Z. Wang *et al.*, "Differential treatment for stuff and things: A simple unsupervised domain adaptation method for semantic segmentation," in *Proc. CVPR*, 2020, pp. 12635–12644.

[12] S. R. Richter, V. Vineet, S. Roth, and V. Koltun, "Playing for data: Ground truth from computer games," in *Proc. ECCV*, 2016, pp. 102–118.

[13] G. Ros, L. Sellart, J. Materzynska, D. Vazquez, and A. M. Lopez, "The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes," in *Proc. CVPR*, 2016, pp. 3234–3243.

[14] T.-H. Vu, H. Jain, M. Bucher, M. Cord, and P. Pérez, "DADA: Depth-aware domain adaptation in semantic segmentation," in *Proc. ICCV*, 2019, pp. 7364–7373.

[15] Y. Li, L. Yuan, and N. Vasconcelos, "Bidirectional learning for domain adaptation of semantic segmentation," in *Proc. CVPR*, 2019, pp. 6936–6945.

[16] Y. Yang and S. Soatto, "FDA: Fourier domain adaptation for semantic segmentation," in *Proc. CVPR*, 2020, pp. 4085–4095.

[17] M. Kim and H. Byun, "Learning texture invariant representation for domain adaptation of semantic segmentation," in *Proc. CVPR*, 2020, pp. 12975–12984.

[18] X. Zhang, H. Zhang, J. Lu, L. Shao, and J. Yang, "Target-targeted domain adaptation for unsupervised semantic segmentation," in *Proc. ICRA*, 2021, pp. 13560–13566.

[19] J. Hoffman, D. Wang, F. Yu, and T. Darrell, "FCNS in the wild: Pixel-level adversarial and constraint-based adaptation," 2016, *arXiv:1612.02649*.

[20] Y.-H. Chen, "No more discrimination: Cross city adaptation of road scene segmenters," in *Proc. ICCV*, 2017, pp. 1992–2001.

[21] Y.-H. Tsai, W.-C. Hung, S. Schulter, K. Sohn, M.-H. Yang, and M. Chandraker, "Learning to adapt structured output space for semantic segmentation," in *Proc. CVPR*, 2018, pp. 7472–7481.

[22] Y. Zou, Z. Yu, B. Vijaya Kumar, and J. Wang, "Unsupervised domain adaptation for semantic segmentation via class-balanced self-training," in *Proc. ECCV*, 2018, pp. 289–305.

[23] F. Pan, I. Shin, F. Rameau, S. Lee, and I. S. Kweon, "Unsupervised intra-domain adaptation for semantic segmentation through self-supervision," in *Proc. CVPR*, 2020, pp. 3764–3773.

[24] L. A. Gatys, A. S. Ecker, and M. Bethge, "Image style transfer using convolutional neural networks," in *Proc. CVPR*, 2016, pp. 2414–2423.

[25] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 4, pp. 834–848, Apr. 2017.

[26] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam, "Rethinking atrous convolution for semantic image segmentation," 2017, *arXiv:1706.05587*.

[27] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 801–818.

[28] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *Proc. CVPR*, 2017, pp. 2881–2890.

[29] Y. Yuan, L. Huang, J. Guo, C. Zhang, X. Chen, and J. Wang, "OCNet: Object context for semantic segmentation," in *Proc. IJCV*, 2021, pp. 1–24.

[30] Z. Huang, X. Wang, L. Huang, C. Huang, Y. Wei, and W. Liu, "CCNet: Criss-cross attention for semantic segmentation," in *Proc. ICCV*, 2019, pp. 603–612.

[31] S. Zheng *et al.*, "Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers," in *Proc. CVPR*, 2021, pp. 6881–6890.

[32] E. Xie, W. Wang, Z. Yu, A. Anandkumar, J. M. Alvarez, and P. Luo, "SegFormer: Simple and efficient design for semantic segmentation with transformers," in *Proc. NeurIPS*, 2021, pp. 1–18.

[33] X. Ding, Y. Guo, G. Ding, and J. Han, "ACNet: Strengthening the kernel skeletons for powerful CNN via asymmetric convolution blocks," in *Proc. ICCV*, 2019, pp. 1911–1920.

[34] G. Sun, W. Wang, J. Dai, and L. Van Gool, "Mining cross-image semantics for weakly supervised semantic segmentation," in *Proc. ECCV*. Cham, Switzerland: Springer, 2020, pp. 347–365.

[35] T. Zhang, G. Lin, W. Liu, J. Cai, and A. Kot, "Splitting vs. merging: Mining object regions with discrepancy and intersection loss for weakly supervised semantic segmentation," in *Proc. ECCV*, 2020, pp. 663–679.

[36] J. Fan, Z. Zhang, C. Song, and T. Tan, "Learning integral objects with intra-class discriminator for weakly-supervised semantic segmentation," in *Proc. CVPR*, 2020, pp. 4283–4292.

[37] A. Kolesnikov and C. H. Lampert, "Seed, expand and constrain: Three principles for weakly-supervised image segmentation," in *Proc. ECCV*. Cham, Switzerland: Springer, 2016, pp. 695–711.

[38] Y.-T. Chang, Q. Wang, W.-C. Hung, R. Piramuthu, Y.-H. Tsai, and M.-H. Yang, "Weakly-supervised semantic segmentation via sub-category exploration," in *Proc. CVPR*, 2020, pp. 8991–9000.

[39] X. Xu and G. H. Lee, "Weakly supervised semantic point cloud segmentation: Towards 10x fewer labels," in *Proc. CVPR*, 2020, pp. 13706–13715.

[40] L. Chen, W. Wu, C. Fu, X. Han, and Y. Zhang, "Weakly supervised semantic segmentation with boundary exploration," in *Proc. ECCV*. Cham, Switzerland: Springer, 2020, pp. 347–362.

[41] O. Veksler, "Regularized loss for weakly supervised single class semantic segmentation," in *Proc. ECCV*. Cham, Switzerland: Springer, 2020, pp. 348–365.

[42] P. Pandey, A. K. Tyagi, S. Ambekar, and A. Prathosh, "Unsupervised domain adaptation for semantic segmentation of NIR images through generative latent search," in *Proc. ECCV*. Cham, Switzerland: Springer, 2020, pp. 413–429.

[43] S. Paul, Y.-H. Tsai, S. Schulter, A. K. Roy-Chowdhury, and M. Chandraker, "Domain adaptive semantic segmentation using weak labels," in *Proc. ECCV*, 2020, pp. 571–587.

[44] J. Huang, S. Lu, D. Guan, and X. Zhang, "Contextual-relation consistent domain adaptation for semantic segmentation," in *Proc. ECCV*. Cham, Switzerland: Springer, 2020, pp. 705–722.

[45] G. Li, G. Kang, W. Liu, Y. Wei, and Y. Yang, "Content-consistent matching for domain adaptive semantic segmentation," in *Proc. ECCV*. Cham, Switzerland: Springer, 2020, pp. 440–456.

[46] J. Fan, Z. Zhang, and T. Tan, "Employing multi-estimations for weakly-supervised semantic segmentation," in *Proc. ECCV*. Cham, Switzerland: Springer, 2020, pp. 332–348.

[47] S. Li *et al.*, "Semantic distribution-aware contrastive adaptation for semantic segmentation," 2021, *arXiv:2105.05013*.

[48] H. Wang, T. Shen, W. Zhang, L.-Y. Duan, and T. Mei, "Classes matter: A fine-grained adversarial approach to cross-domain semantic segmentation," in *Proc. ECCV*. Cham, Switzerland: Springer, 2020, pp. 642–659.

[49] Y. Ganin and V. Lempitsky, "Unsupervised domain adaptation by backpropagation," in *Proc. ICML*, 2015, pp. 1180–1189.

[50] E. Tzeng, J. Hoffman, T. Darrell, and K. Saenko, "Simultaneous deep transfer across domains and tasks," in *Proc. ICCV*, 2015, pp. 4068–4076.

[51] I. Shin, S. Woo, F. Pan, and I. S. Kweon, "Two-phase pseudo label densification for self-training based domain adaptation," in *Proc. ECCV*. Cham, Switzerland: Springer, 2020, pp. 532–548.

[52] M. Naseer Subhani and M. Ali, "Learning from scale-invariant examples for domain adaptation in semantic segmentation," 2020, *arXiv:2007.14449*.

[53] Y. Ganin *et al.*, "Domain-adversarial training of neural networks," *J. Mach. Learn. Res.*, vol. 17, no. 1, pp. 2030–2096, May 2015.

[54] P. Zhang, B. Zhang, T. Zhang, D. Chen, Y. Wang, and F. Wen, "Prototypical pseudo label denoising and target structure learning for domain adaptive semantic segmentation," in *Proc. CVPR*, 2021, pp. 12414–12424.

[55] W. Zhou, Y. Wang, J. Chu, J. Yang, X. Bai, and Y. Xu, "Affinity space adaptation for semantic segmentation across domains," *IEEE Trans. Image Process.*, vol. 30, pp. 2549–2561, 2021.

[56] E. Romera, L. M. Bergasa, K. Yang, J. M. Alvarez, and R. Barea, "Bridging the day and night domain gap for semantic segmentation," in *Proc. IV*, 2019, pp. 1312–1318.

[57] X. Wu, Z. Wu, H. Guo, L. Ju, and S. Wang, "DanNet: A one-stage domain adaptation network for unsupervised nighttime semantic segmentation," in *Proc. CVPR*, 2021, pp. 15769–15778.

[58] Q. Zhang, J. Zhang, W. Liu, and D. Tao, "Category anchor-guided unsupervised domain adaptation for semantic segmentation," 2019, *arXiv:1910.13049*.

[59] Y. Liu, J. Deng, X. Gao, W. Li, and L. Duan, "BAPA-Net: Boundary adaptation and prototype alignment for cross-domain semantic segmentation," in *Proc. ICCV*, 2021, pp. 8801–8811.

[60] D. Kothandaraman, A. M. Nambiar, and A. Mittal, "Domain adaptive knowledge distillation for driving scene semantic segmentation," in *Proc. WACV*, 2021, pp. 134–143.

[61] W. Tranheden, V. Olsson, J. Pinto, and L. Svensson, "DACS: Domain adaptation via cross-domain mixed sampling," in *Proc. WACV*, 2021, pp. 1379–1389.

[62] Y. Grandvalet *et al.*, "Semi-supervised learning by entropy minimization," in *Proc. CAP*, 2005, pp. 281–296.

[63] H. Jain, J. Zepeda, P. Pérez, and R. Gribonval, "SUBIC: A supervised, structured binary code for image search," in *Proc. ICCV*, 2017, pp. 833–842.

[64] H. Jain, J. Zepeda, P. Perez, and R. Gribonval, "Learning a complete image indexing pipeline," in *Proc. CVPR*, 2018, pp. 4933–4941.

[65] M. Long, H. Zhu, J. Wang, and M. I. Jordan, "Unsupervised domain adaptation with residual transfer networks," 2016, *arXiv:1602.04433*.

[66] P. Morerio, J. Cavazza, and V. Murino, "Minimal-entropy correlation alignment for unsupervised deep domain adaptation," 2017, *arXiv:1711.10288*.

[67] F. M. Carlucci, L. Porzi, B. Caputo, E. Ricci, and S. R. Bulo, "Auto-DIAL: Automatic domain alignment layers," in *Proc. ICCV*, 2017, pp. 5077–5085.

[68] A. Ma, J. Li, K. Lu, L. Zhu, and H. T. Shen, "Adversarial entropy optimization for unsupervised domain adaptation," *IEEE Trans. Neural Netw. Learn. Syst.*, early access, May 3, 2021, doi: 10.1109/TNNLS.2021.3073119.

[69] H. Xu, M. Yang, L. Deng, Y. Qian, and C. Wang, "Neutral cross-entropy loss based unsupervised domain adaptation for semantic segmentation," *IEEE Trans. Image Process.*, vol. 30, pp. 4516–4525, 2021.

[70] J. Xie, R. Girshick, and A. Farhadi, "Unsupervised deep embedding for clustering analysis," in *Proc. ICML*, 2016, pp. 478–487.

[71] L. A. Gatys, A. S. Ecker, and M. Bethge, "Texture synthesis using convolutional neural networks," 2015, *arXiv:1505.07376*.

[72] Y. Zhang, M. Ye, Y. Gan, and W. Zhang, "Knowledge based domain adaptation for semantic segmentation," *Knowl.-Based Syst.*, vol. 193, Jun. 2020, Art. no. 105444.

[73] Y. Luo, L. Zheng, T. Guan, J. Yu, and Y. Yang, "Taking a closer look at domain shift: Category-level adversaries for semantics consistent domain adaptation," in *Proc. CVPR*, 2019, pp. 2507–2516.

[74] C.-Y. Lee, T. Batra, M. H. Baig, and D. Ulbricht, "Sliced wasserstein discrepancy for unsupervised domain adaptation," in *Proc. CVPR*, 2019, pp. 10285–10295.

[75] Y. Zhang, P. David, H. Foroosh, and B. Gong, "A curriculum domain adaptation approach to the semantic segmentation of urban scenes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 8, pp. 1823–1841, Aug. 2019.

[76] Y. Zou, Z. Yu, X. Liu, B. Kumar, and J. Wang, "Confidence regularized self-training," in *Proc. ICCV*, 2019, pp. 5982–5991.

[77] Y. Luo, P. Liu, T. Guan, J. Yu, and Y. Yang, "Significance-aware information bottleneck for domain adaptive semantic segmentation," in *Proc. ICCV*, 2019, pp. 6778–6787.

[78] Y.-H. Tsai, K. Sohn, S. Schulter, and M. Chandraker, "Domain adaptation for structured output via discriminative patch representations," in *Proc. ICCV*, 2019, pp. 1456–1465.

[79] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. CVPR*, 2016, pp. 770–778.

[80] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*.

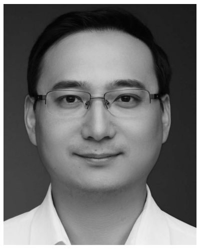
[81] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *Proc. CVPR*, 2009, pp. 248–255.

[82] M. Cordts *et al.*, "The cityscapes dataset for semantic urban scene understanding," in *Proc. CVPR*, 2016, pp. 3213–3223.

[83] A. Paszke *et al.*, "Pytorch: An imperative style, high-performance deep learning library," in *Proc. NeurIPS*, 2019, pp. 8026–8037.

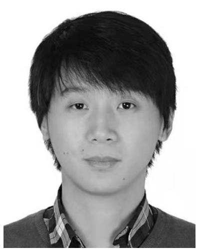
[84] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. ICLR*, 2015, pp. 1–15.

Xiaohong Zhang (Graduate Student Member, IEEE) received the B.S. degree in automation from the Nanjing University of Science and Technology, Nanjing, China, in 2019, where she is currently pursuing the Ph.D. degree with the School of Computer Science and Engineering. Her current research interests include semantic segmentation, stereo vision, and deep learning.

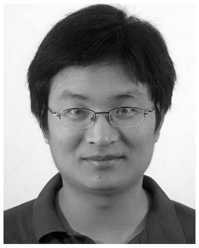


Yi Chen (Member, IEEE) received the Ph.D. degree from the School of Computer Science and Technology, Nanjing University of Science and Technology, in 2012. He is currently an Associate Professor with the School of Computer Science and Technology, Nanjing Normal University. His current research interests include pattern recognition, compute vision, and machine learning.

Ziyi Shen received the Ph.D. degree in optical engineering from the Beijing Institute of Technology in 2018. She is currently a Research Fellow at University College London. Before that, she worked at the Inception Institute of Artificial Intelligence. Her current research interests include low-level vision, video understanding, image processing, and medical image processing.



Yuming Shen received the B.Eng. degree from Tongji University in 2011, the M.Sc. degree from The University of Sheffield in 2012, and the Ph.D. degree from the University of East Anglia in 2018. He is currently a Post-Doctoral Researcher at the University of Oxford. His research interests include unsupervised learning, deep learning, and vision and language.



Haofeng Zhang (Member, IEEE) received the B.Eng. and Ph.D. degrees from the School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing, China, in 2003 and 2007, respectively. From December 2016 to December 2017, he was an Academic Visitor at the University of East Anglia, Norwich, U.K. He is currently a Professor with the School of Computer Science and Engineering, Nanjing University of Science and Technology. His research interests include computer vision and robotics.

Yudong Zhang (Senior Member, IEEE) received the Ph.D. degree from Southeast University in 2010. He worked as a Post-Doctoral Researcher with Columbia University, USA, from 2010 to 2012. He is currently working as a Professor with the School of Informatics, University of Leicester, U.K. His research interests include deep learning and medical image analysis.