Threshold-adaptive Unsupervised Focal Loss for Domain Adaptation of Semantic Segmentation

Weihao Yan, Yeqiang Qian, *Member, IEEE,* Chunxiang Wang, *Member, IEEE,* and Ming Yang, *Member, IEEE*

***Abstract*—Semantic segmentation is an important task for intelligent vehicles to understand the environment. Current deep learning methods require large amounts of labeled data for training. Manual annotation is expensive, while simulators can provide accurate annotations. However, the performance of the semantic segmentation model trained with the data of the simulator will significantly decrease when applied in the actual scene. Unsupervised domain adaptation (UDA) for semantic segmentation has recently gained increasing research attention, aiming to reduce the domain gap and improve the performance on the target domain. In this paper, we propose a novel two-stage entropy-based UDA method for semantic segmentation. In stage one, we design a threshold-adaptative unsupervised focal loss to regularize the prediction in the target domain, which has a mild gradient neutralization mechanism and mitigates the problem that hard samples are barely optimized in entropy-based methods. In stage two, we introduce a data augmentation method named cross-domain image mixing (CIM) to bridge the semantic knowledge from two domains. Our method achieves state-of-the-art 58.4% and 59.6% mIoUs on SYNTHIA-to-Cityscapes and GTA5-to-Cityscapes using DeepLabV2 and competitive performance using the lightweight BiSeNet.**

***Index Terms*—Semantic segmentation, unsupervised domain adaptation, entropy minimization, focal loss.**

I. Introduction

SEMANTIC segmentation is an important perceptual task for the intelligent transportation system, which can provide pixel-level semantic information, e.g., road, traffic sign, and pedestrian. Thanks to the development of deep learning and large-scale public datasets, semantic segmentation has made remarkable progress [1]–[4] in recent years. Most of these supervised deep learning methods have an indispensable requirement for high-quality labeled data, which are expensive to obtain. It takes about 90 minutes to label an image of Cityscapes [5]. So an alternative way is to utilize synthetic datasets like SYNTHIA [6] and GTA5 [7] for model training. However, the models trained on synthetic datasets (source domain) usually perform poorly in real scenarios (target domain) due to the large domain gap. To alleviate this severe problem, researchers have resorted to investigating unsupervised domain adaptation (UDA) methods for semantic segmentation.

This work is supported by the National Natural Science Foundation of China (62103261/62173228). *(Corresponding author: Ming Yang.)*

Weihao Yan, Chunxiang Wang and Ming Yang are with the Department of Automation, Shanghai Jiao Tong University, Key Laboratory of System Control and Information Processing, Ministry of Education of China, Shanghai, 200240, China (email: mingyang@sjtu.edu.cn)

Yeqiang Qian is with University of Michigan-Shanghai Jiao Tong University Joint Institute, Shanghai Jiao Tong University, Shanghai, 200240, China. (email: qianyeqiang@sjtu.edu.cn)

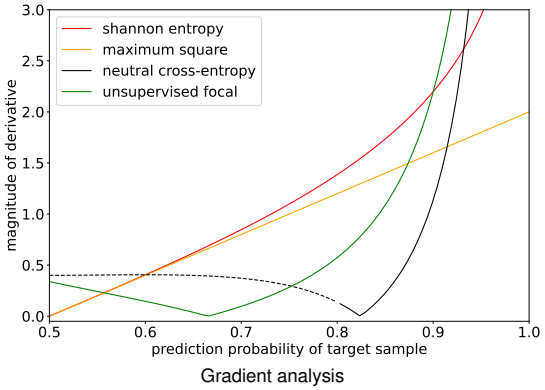


Fig. 1. Gradient analysis of entropy-based unsupervised loss. Shannon entropy sharpens the prediction distribution, and its gradient is strongly biased toward easy samples (prediction probabilities near 1). Maximum square loss reduces the gradient magnitude of easy samples, but the gradient tends to be 0 when probability approaches 0.5. Neutral cross-entropy smoothes the over-sharpness of entropy loss. However, it utilizes a high confidence threshold (0.8) to filter the prediction results. Unsupervised focal loss has a mild gradient neutralization mechanism and dynamic threshold adjustment strategy, making hard samples with low prediction probabilities optimized.

The literature on UDA for semantic segmentation is recently dominated by adversarial-based and self-training methods. Adversarial-based methods [8]–[11] have shown competitive adaptation performance by learning domain invariant representations at the input, feature, or output levels. But the training procedure is complex, and the adversarial loss [12] frequently converges to local optimal. Self-training methods [13]–[16] have state-of-the-art adaptation performance recently. Pseudo labels for the target domain are generated and used to re-train the segmentation model. However, they often require computationally expensive iterative training and act more like post-processing of pre-adapted models.

Recently, a new line of entropy-based UDA methods [8], [17]–[19] for semantic segmentation has shown progressive development, which regards the UDA for semantic segmentation as a regularization process on the target domain. The work in [8] proposes to minimize the entropy maps of prediction results on the target domain, making the network produce high-confidence predictions. Then [17] finds the gradient of entropy is biased towards samples that are easy to transfer and proposes the maximum square loss to balance the gradient of the well-classified target samples. Besides that, the over-sharpness of entropy minimization is mitigated in [18] by introducing the pixel-level consistency regularization and proposing the neutral cross-entropy loss. After that, [19] designs confidence-aware entropy to help the model focus more on high-confident data points. Entropy-based methods

have good theoretical support and generally train faster [8], [17]–[19]. Though efficient and effective, the performance of entropy-based methods still falls behind the other two types of state-of-the-art methods and needs further improvement.

There are two main problems that hinder the further improvement of these entropy-based methods. One is that the hard samples are barely optimized. In the training process, the model parameters are updated through backpropagation. Once that learning rate is selected, the degree of parameters updating is proportional to the magnitude of the gradient of the loss. The gradient analysis of these entropy-based losses [8], [17], [18] is shown in Fig. 1, where the binary classification case is used for a clearer presentation. The gradient of the shannon entropy loss is strongly biased towards samples with probabilities near 1. Still, it decreases to 0 when probability approaches 0.5, making the hard-to-transfer samples barely optimized. A similar situation happens in the maximum square loss, though it reduces the gradient of these well-classified target samples. For neutral cross-entropy loss, a confidence threshold like 0.8 is selected to filter the prediction results with low confidence for training stability. It may make the hard samples that are usually with low prediction probabilities not included in calculating the loss, let alone optimized. Another is that these entropy-based UDA methods only impose a regularization item on the target domain while ignoring the explicit connection of semantic knowledge between the source and the target domain. One way is to utilize adversarial training to learn domain invariant features, but these adversarial-based methods are well-known for difficult training.

In this paper, we propose a novel two-stage entropy-based UDA method to mitigate the above two problems. In stage one, the threshold-adaptive unsupervised focal loss is designed and applied to the target domain. It has a gentle gradient neutralization mechanism to smooth the over-sharpness of shannon entropy loss and a class-level dynamic threshold adjustment strategy, which helps optimize hard samples. In stage two, we introduce a data enhancement method named cross-domain image mixing (CIM) to bridge the semantic knowledge from two domains. Three image-label pairs from the source domain, target domain, and CIM are fed into the network for training. Experiments are conducted on two synthetic-to-real settings (SYNTHIA-to-Cityscapes and GTA5-to-Cityscapes) to verify the effectiveness of our method. It achieves state-of-the-art 58.4% and 59.6% mIoUs using DeepLabV2 and competitive 52.2% and 55.4% mIoUs using lightweight BiSeNet. Our contributions are summarized as follows:

- We design a threshold-adaptive unsupervised focal loss for the UDA problem of semantic segmentation. It adjusts the gradient contributions of easy and hard samples and has a class-level dynamic threshold adjustment strategy, helping the hard target samples get optimized.
- We introduce a data augmentation method named cross-domain image mixing (CIM) into entropy-based methods, which bridges the semantic knowledge of two domains.
- We propose a novel two-stage UDA framework for semantic segmentation, achieving state-of-the-art performance on major cross-domain benchmark datasets like SYNTHIA-to-Cityscapes and GTA5-to-Cityscapes.

II. Related Work

A. Semantic Segmentation

Semantic segmentation has made significant advances in recent years since the development of deep learning and the availability of public datasets. FCN [20] is the first segmentation model that achieves pixel-level predictions. After that, many works on semantic segmentation have emerged. One stream of methods aims to improve the accuracy of semantic segmentation models [1]–[3], [21]–[23]. DeepLabV2 [21] proposes atrous spatial pyramid pooling (ASPP) with filters at multiple sampling rates, improving the segmentation performance on multi-scale objects. The other trend intends to develop real-time models for applications [4], [24]–[26]. BiSeNet [24] consists of Spatial Path, Context Path, and Feature Fusion Module, which makes a balance between speed and segmentation performance. Both of them have indispensable demand for high-quality labeled datasets, which are expensive to acquire. One possible method to reduce data labeling costs is adopting synthetic datasets [6], [7]. However, models trained on synthetic datasets often perform poorly when applied in real scenarios due to the large domain gap. UDA methods are developed for this problem. In this paper, lightweight BiSeNet [24] with ResNet18 as the backbone is used for real-time performance and training efficiency. Meanwhile, we also adopt the widely used DeeplabV2 [21] for a fair comparison with state-of-the-art.

B. UDA for Semantic Segmentation

UDA methods can alleviate the domain gap between source and target domains, improving the performance of the segmentation model on the unlabeled target domain. Current advanced UDA approaches for semantic segmentation are dominated by adversarial-based and self-training methods. On one hand, adversarial-based methods [8]–[11] usually adopt generative adversarial networks (GAN) [12] to align the distributions of two domains, aiming to learn domain invariant representations at different levels. Meanwhile, image style translation (IST) methods [9], [27], [28] transfer the texture of source domain images to the target domain, reducing the domain gap at the input level. The work in [28] proposes a global photometric alignment module that aligns the image in the source domain with the reference image in the target domain. Although adversarial-based methods usually have competitive performance, they suffer from training instability. In this paper, a prevalently used IST method CycleGAN [9] is taken to preprocess the images from the source domain offline.

On the other hand, self-training methods [13]–[16] generate pseudo labels in the target domain and employ them for iterative training. Class balanced self-training [13] is the first to apply pseudo-labeling to UDA. After that, uncertainty estimation is applied to rectify the pseudo labels in [14]. The method in [29] generates co-evolving pseudo labels for their self-supervised framework. Current state-of-the-art ProDA [16] uses prototypical pseudo label denoising to update the pseudo labels in stage one online and adopts knowledge distillation to a self-supervised model in the following two stages. Nevertheless, producing high-quality pseudo labels

remains challenging, and iterative training is computationally expensive and time-consuming. Self-training methods are more like "post-processing" of the pre-adapted model, which can follow the adversarial-based and entropy-based methods to further improve the adaptation performance.

Moreover, some UDA methods [30], [31] tend to improve the adaptation performance of the semantic segmentation model from daytime to nighttime. DANNet [31] is the first one-stage adaptation framework for nighttime semantic segmentation via adversarial learning. In addition, the work in [32] investigates the adaptation in affinity space, which leverages co-occurring patterns between pairwise pixels. The guidance from self-supervised depth estimation is leveraged in [33] to strengthen the target semantic predictions. Recently, [34] explores the possibility of improving the network structure and training strategy for domain adaptative semantic segmentation, which utilizes the MiT-B5 [23] encoder and boosts the state-of-the-art. However, the training of transformer architecture is well-known for being complex and cumbersome. We still adopt the CNN architecture for experiments, and the choice of segmentation model is not the focus of our work.

Another stream of entropy-based UDA methods follows the cluster assumption [35], which regards the UDA for semantic segmentation as a regularization process on the target domain [8], [17]–[19]. Entropy minimization is proposed in [8], which minimizes the entropy maps of prediction results on the target domain, and encourages the network to produce predictions with high confidence. Maximum square loss is designed in [17] to balance the gradient of well-classified target samples, which alleviates the problem that the gradient of entropy is biased towards easy-to-transfer samples. Pixel-level consistency regularization is introduced in [18] to form neutral cross-entropy loss, which has a gradient neutralization mechanism to smooth the over-sharpness of entropy loss. The work in [19] updates the common entropy to confidence-aware entropy, forcing the network to focus on the high-confidence predictions. Though efficient and effective, entropy-based methods only bring little improvement and still lag behind current advanced UDA methods. As discussed before, they barely optimize hard-to-transfer samples and lack the explicit semantic knowledge connection between two domains. In this paper, we focus on the entropy-based methods, aiming to mitigate the above two problems and boost their adaptation performance to the state-of-the-art.

C. Image Mixing Strategy

Image mixing has proven effective in semi-supervised learning (SSL). The central idea is to mix two images and their labels, forming additional, highly perturbed training samples. ClassMix [36] randomly selects half of the classes in one image and pastes them onto another one to better respect semantic boundaries. Image Mixing is a preferable approach to bridge the semantic knowledge between two domains, and [37] has verified its effectiveness in the UDA problem of semantic segmentation. Different from [37], we emphasize the loss contribution of the pixels near the boundaries of mixing mask, which have the receptive field of both domains, and design long-tail class pasting to improve adaptation performance.

III. METHOD

The framework of our two-stage UDA method is shown in Fig 2. In stage one, CycleGAN [9] is used to transfer the image style of the source domain to the target domain offline when using BiSeNet, while DeepLabV2 does not. The image x_t from the target domain goes through perturbation g, and the augmented image is denoted as x_{t^*}. Then x_t and x_{t^*} are fed into the segmentation model $f(\theta)$, and the predictions $f(x_t,\theta)$ and $f(x_{t^*},\theta)$ are obtained. Apply the same perturbation on $f(x_t,\theta)$ to align with $f(x_{t^*},\theta)$, our threshold-adaptive unsupervised focal loss $L_u(\hat{f}(x_t,\theta), f(x_{t^*},\theta))$ is calculated. Meanwhile, the supervised cross-entropy loss $L_s(f(x_s,\theta), y_s)$ is computed in the source domain.

In stage two, the pre-adapted model from stage one $f(\tilde{\theta})$ generates pseudo labels for target samples. Then mixed image-label pairs (x_m,y_m) are acquired through CIM with boundary enhancement and long-tail class pasting. Supervised cross-entropy loss L_m is also used in stage two with the L_s and L_u. The details will be presented in the following.

A. Overview of entropy-based UDA methods

Use $D_s = \{(x_s,y_s)|x_s \in R^{3\times H\times W}, y_s \in R^{H\times W}\}$ and $D_t = \{x_t|x_t \in R^{3\times H\times W}\}$ to denote the labeled source domain and unlabeled target domain. The general loss function of the entropy-based UDA method can be formulated as:

$$Loss = L_s(x_s,y_s) + \lambda_u L_u(x_t) \tag{1}$$

where L_s is the supervised cross-entropy loss in the source domain, and L_u is the unsupervised loss applied on the target images with coefficient λ_u. The training objective is to adjust the parameters of model θ to minimize the loss function (1). Denote $f(x_t,\theta)$, $\hat{f}(x_t,\theta)$, and $f(x_{t^*},\theta)$ as p_t, $\hat{p}_t$, and p_{t^*} for convenience. The shannon entropy loss (L_{shan}) is directly utilized as the L_u in [8]:

$$L_{shan}(p_t) = -\frac{1}{|I_t|}\sum_{n\in I_t}\sum_{c=1}^{C} p_t^{n,c}\log(p_t^{n,c}) \tag{2}$$

where C is the number of classes, c represents channel number, n denotes pixel location, and I_t is the loss calculation mask. Suppose $p_t \in R^{C\times H\times W}$, and $p_m = \max_{dim=C} p_t$ is the prediction confidence map. $I_t = \{(h,w)|p_m^{h,w} > t, t \in [0,1]\}$ means that only pixels with prediction confidence greater than the threshold t are included in the loss calculation.

B. Threshold-adaptative unsupervised focal loss

Inspired by the focal loss [38], we propose the unsupervised focal loss for domain adaptation of semantic segmentation. Specifically, it consists of two parts, the first part is the shannon entropy for the prediction probability distribution $\hat{p}_t$ of weakly perturbed target image x_t, which makes the model tend to produce high-confidence prediction results:

$$L_{shan}(\hat{p}_t) = -\frac{1}{|I_t|}\sum_{n\in I_t}\sum_{c=1}^{C} \hat{p}_t^{n,c}\log(\hat{p}_t^{n,c}) \tag{3}$$

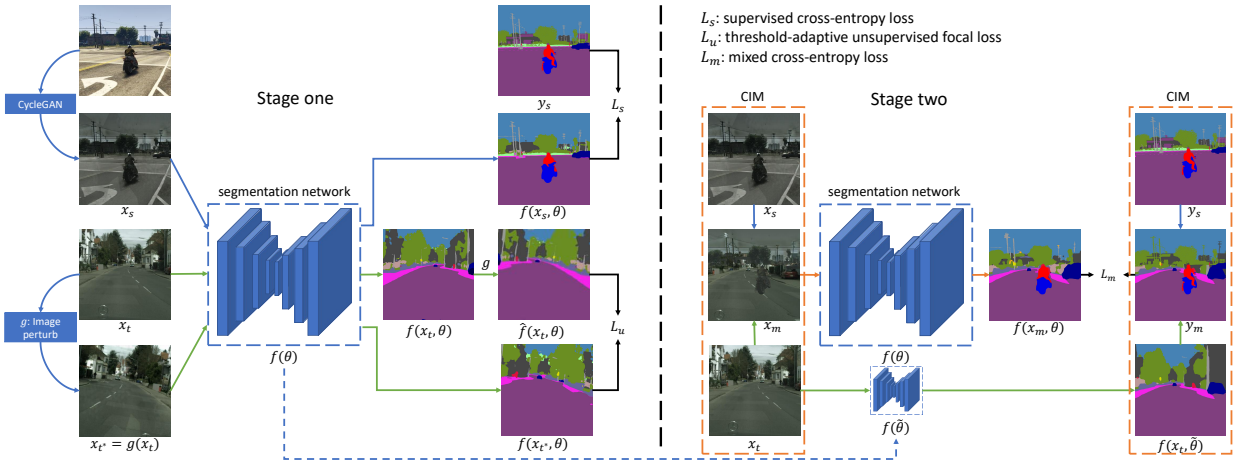


Fig. 2. Two-stage UDA framework for semantic segmentation. The data flow of the source domain, target domain, and CIM is denoted by blue, green, and orange lines, respectively. In stage one, CycleGAN is adopted to preprocess the source domain images. Then the threshold-adaptative unsupervised focal loss is applied in the target domain and the supervised cross-entropy loss in the source domain. In stage two, CIM is introduced and helps connect the semantic knowledge between two domains. L_s, L_u, and L_m present supervised cross-entropy loss, threshold-unsupervised focal loss, and mixed cross-entropy loss.

The second part is the KL-divergence $L_{KL'}\left(\hat{p}_t, p_{t^*}\right)$ with the loss adjustment item:

$$\frac{1}{|I_t|}\sum_{n\in I_t}\sum_{c=1}^{C}\hat{p}_t^{n,c}\cdot\left(\log\left(\hat{p}_t^{n,c}\right)-\left(1-p_{t^*}^{n,c}\right)^{\gamma}\log\left(p_{t^*}^{n,c}\right)\right) \quad (4)$$

where parameter γ controls the degree of regularization, the gradient of $\hat{p}_t$ is detached and serves as the soft "pseudo label." The function of KL-divergence is to make the model produce consistent prediction results for perturbed image pairs, that is, to align the p_{t^*} with $\hat{p}_t$, making it more robust. The $(1-p_{t^*})^{\gamma}$is introduced to balance the loss contribution of easy and hard samples. As $\hat{p}_t$ is fixed, the adjustment item tends to be 0 when p_{t^*} approaches 1 (easy-to-transfer) and vice versa. The unsupervised focal loss is obtained through the summation of shannon entropy and adjusted KL-divergence:

$$L_{focal}\left(\hat{p}_t, p_{t^*}\right)=L_{shan}\left(\hat{p}_t\right)+L_{KL'}\left(\hat{p}_t, p_{t^*}\right) \quad (5)$$

Then we discuss the relationship between our unsupervised focal loss and the supervised focal loss. The supervised focal loss (L_{s_focal}) is widely used in semantic segmentation for hard samples optimization, which is formulated as:

$$L_{s\_focal}\left(y_t, p_t\right)=-\frac{1}{|I_t|}\sum_{n\in I_t}\sum_{c=1}^{C}{y_t}^{n,c}\left(1-p_t^{n,c}\right)^{\gamma}\log\left(p_t^{n,c}\right) \quad (6)$$

We expand the supervised focal loss as:

$$\begin{aligned}&L_{s\_focal}\left(y_t, p_t\right)=L_{shan}\left(y_t\right)+L_{KL'}\left(y_t, p_t\right)\\&=-\frac{1}{|I_t|}\sum_{n\in I_t}\sum_{c=1}^{C}y_t^{n,c}\log\left(y_t^{n,c}\right)+\\&\frac{1}{|I_t|}\sum_{n\in I_t}\sum_{c=1}^{C}y_t^{n,c}(\log\left(y_t^{n,c}\right)-\left(1-{p_t}^{n,c}\right)^{\gamma}\log\left({p_t}^{n,c}\right))\end{aligned} \quad (7)$$

In the supervised focal loss, the $L_{shan}(y_t)$ is constant since the labels are available and fixed. It demonstrates that minimizing the supervised focal loss $L_{s\_focal}(y_t, p_t)$ is equal to optimizing the adjusted KL-divergence $L_{KL'}(y_t, p_t)$, which inspires us to use perturbed image pairs to establish an unsupervised form of adjusted KL-divergence $L_{KL'}\left(\hat{p}_t, p_{t^*}\right)$. Compared with the supervised one, our unsupervised focal loss replaces the ground truth y_t with the estimated prediction $\hat{p}_t$. Essentially, our unsupervised focal loss is a particular format of supervised focal loss, with the shannon entropy of soft "pseudo label" $L_{shan}(\hat{p}_t)$ as the learnable variable. It can adjust the contribution of easy and hard target samples to the unsupervised loss L_u so that the hard samples are optimized.

The computational mask I_t is another factor that hinders the optimization of hard samples, which adopts a confidence threshold like 0.8 to remove pixels with low maximum prediction probabilities to stabilize the unsupervised training process [18]. However, those hard samples are often with low maximum prediction probabilities since the model cannot discriminate their classes well. We visualize the average maximum prediction probability and proportion of pixels above the threshold (0.8) of each class in the GTA5-to-Cityscapes experiment, shown in Fig. 3. Classes with higher IoU generally have higher maximum prediction probabilities and vice versa. Those classes with low maximum prediction probabilities are rarely incorporated into the computation of the unsupervised loss, leading to low IoU.

We introduce a class-level dynamic threshold adjustment strategy for I_t, where the threshold is updated online during the training process. Denote the threshold for each class as $\alpha \in R^C$. We use the Exponential Moving Average strategy to update it for the current sample from the target domain:

$$\begin{aligned}&\alpha_0=[t,...,t]\in R^C\\&\alpha_k=a\alpha_{k-1}+(1-a)\alpha_{k'}, k\geq 1\end{aligned} \quad (8)$$

where α_0 represents the initial threshold t (set as 0.8), α_k and α_{k-1} denote the class threshold in the kth and $(k-1)$th iteration, a is the historical memory parameter, and $\alpha_{k'}$ is the

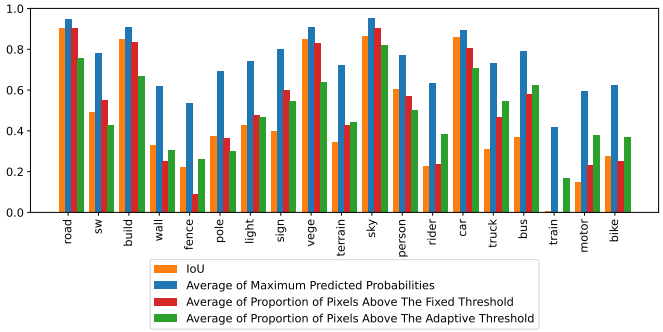


Fig. 3. Numerical characteristic analysis of each class on GTA5-to-Cityscapes. When using a fixed high confidence threshold (0.8), some classes like fence and train are barely included in the loss calculation. Our adaptative threshold strategy helps these classes contribute more to the loss.

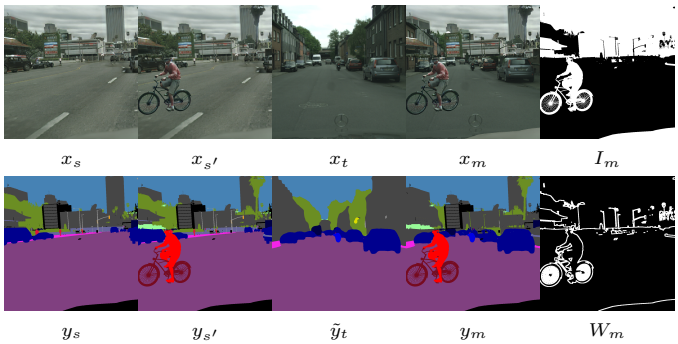


Fig. 4. Qualitative demonstration of Cross-domain Image Mixing. (x_s,y_s), $(x_{s'},y_{s'})$, $(x_t,\tilde{y}_t)$, and (x_m,y_m) represent the image-label pairs from the source domain, source domain after long-tail class pasting, target domain, and CIM. I_m is the mixing mask and W_m is the coefficient mask.

class threshold calculated from the current sample:

$$\begin{aligned} p_m^c &= p_m[argmax(p_t) = c] \\ \alpha_{k'}^c &= descend\,(p_m^c)\,[b(e^{\alpha_{k-1}^c-1})^d|p_m^c|] \end{aligned} \tag{9}$$

where p_m^c is the prediction confidence list for class c, obtained by choosing the pixels with pseudo label c. $\alpha_{k'}^c$ represents the threshold for class c computing from the current sample. b is the global proportion of pixels used for loss calculation, like pixels with top 80% prediction confidence. α_{k-1}^c denotes the threshold of class c in $(k-1)$th iteration, and d regularizes the proportion of classes with low α_{k-1}^c. In other words, after sorting the p_m^c in descending order, the $[b(e^{\alpha_{k-1}^c-1})^d|p_m^c|]$th item is utilized as the threshold $\alpha_{k'}^c$.

This class-level threshold adjustment strategy has three advantages compared with a fixed threshold: First, it considers the difficulty of different categories so that the classes with lower prediction confidence probabilities have lower thresholds. Second, the threshold is updated on each image, which is more suitable for the current segmentation result. Third, parameters b and d make each class balance between loss contribution and noise suppression, presented in Fig. 3.

Our threshold-adaptative unsupervised focal loss is finally obtained with the class-level threshold adjustment strategy. The calculation method for I_t is updated to $I_t = \{(h,w)|argmax(p_t^{h,w} = c), p_m^{h,w} > \alpha[c], \alpha[c] \in [0,1]\}$. Then we analyze the gradient of these entropy-based losses, shown in Fig. 1. For simplicity, we consider the binary classification case. Use p and $\hat{p}$ to denote learnable classification probability and estimated prediction. Shannon entropy tends to predict at 0 and 1, sharpening the prediction distribution, and its gradient is strongly biased toward easy samples. Maximum square loss reduces the gradient when p approaches 1, but the gradient of hard samples with p near 0.5 tends to be 0. When set $\hat{p} = 0.6$, neutral cross-entropy loss shifts the global minimum towards the middle location (0.82). However, it utilizes a high confidence threshold (0.8), and the samples with low confidence ($p < 0.8$) are not included in the loss calculation. Our threshold-adaptative unsupervised focal loss also shifts the global minimum to the middle location (0.67), with a milder gradient neutralization mechanism and dynamic threshold adjustment strategy. Moreover, the gradient increases when p gets to 0.5, making hard samples contribute more to the optimization process and get optimized.

C. CIM

To improve the adaptation performance on long-tail classes, we use the image-label pairs in the source domain to make a category database and conduct long-tail class pasting on the source image. Specifically, we build the mapping between classes and source domain image-label pairs offline: $M_{c_i} = \{(x_s,y_s)|c_i \in y_s\}$, where c_i is the ith class. When class c_i is demanded, an image-label pair (x_s,y_s) from M_{c_i} is randomly selected, and the corresponding contents are pasted into the current sample. In the first and second columns in Fig. 4, the rider and bike are pasted to the source image and label. The inverse of the dynamic threshold α after softmax is used as the probability of the class selection. The enhanced image-label pair from the source domain are denoted as $(x_{s'}, y_{s'})$.

Denote the adapted model from stage one as $f(\tilde{\theta})$. The target image x_t goes through $f(\tilde{\theta})$ and gets its pseudo label $\tilde{y}_t$. The image-label pairs from the source and target domain are mixed at the class level [36]:

$$\begin{aligned} x_m &= I_m \cdot x_{s'} + (1 - I_m) \cdot x_t \\ y_m &= I_m \cdot y_{s'} + (1 - I_m) \cdot \tilde{y}_t \end{aligned} \tag{10}$$

where I_m is the mixing mask obtained by randomly choosing half of the classes in the $y_{s'}$. Qualitative demonstration of Cross-domain Image Mixing is shown in Fig. 4.

The cross-entropy loss is also used for mixed image-label pairs. Since the receptive field of pixels near the boundaries of the mixing mask I_m (denoted as $p \in b(I_m)$) after passing through the convolutional network will contain both real and virtual contents, we enhance the loss contribution of these pixels. Expressly, for the pixels within the 7×7 field of the boundary point, their loss coefficient $W_m(p)$ is set to 2 while others remain 1:

$$W_m(p) = \begin{cases} 2, & if \quad p \in b(I_m) \\ 1, & otherwise \end{cases} \tag{11}$$

The training loss of stage two is formulated as follows, and λ_m is the coefficient for the mixed part.

$$Loss = L_s(x_s,y_s) + \lambda_u L_u(x_t) + \lambda_m L_m(x_m,y_m) \tag{12}$$

IV. EXPERIMENTS

A. Datasets

We conduct experiments on two popular synthetic-to-real settings: SYNTHIA-to-Cityscapes and GTA5-to-Cityscapes. The synthetic datasets SYNTHIA [6] and GTA5 [7] are used as the source domain datasets, and the actual driving dataset Cityscapes is the target domain dataset.

SYNTHIA consists of 9400 photo-realistic frames, which are 760 $\times$ 1280. GTA5 has 24966 annotated images with a resolution of 1052 $\times$ 1914. The label classes of the two synthetic datasets are consistent with Cityscapes. GTA5 contains 19 classes, while SYNTHIA has 16 categories. For the target domain, Cityscapes has 2975 and 500 precisely annotated images for training and validation and 19997 roughly annotated images. To fully exploit the advantage of our UDA method, we use 19997 images for BiSeNet like [18], while 2975 images for DeepLabV2 for a fair comparison.

B. Segmentation Network

For training efficiency and real-time inference, BiSeNet [24] with ResNet18 [39] as the backbone is adopted as the segmentation network in most experiments. Meanwhile, DeepLabV2 with ResNet101 as the backbone is also utilized for a fair comparison with other state-of-the-art methods.

C. Evaluation Methods and Experiment Settings

Intersection-over-union (IoU) for each class and mean-intersection-over-union (mIoU) for all classes are the evaluation metrics in our experiments. For GTA5-to-Cityscapes, 19 classes are evaluated, while 16 and 13 are for the SYNTHIA-to-Cityscapes setting.

For BiSeNet, the images in SYNTHIA keep their original size of 760 $\times$ 1280, while the ones in GTA5 are randomly cropped into 1000 $\times$ 1000 during the training. The model is pretrained on the source domain dataset. The batch size is 12, and total training iterations are 20000, using two 1080Ti GPUs. We use the SGD optimizer with the learning rate of 2.5×10^{-4}, momentum of 0.9, and weight decay of 5×10^{-4}. For learning rate adjustment, we use warm-up for the first 100 iterations, followed by a poly-type tuning strategy, decaying the initial learning rate with $(1-\frac{iter-warm\_iter}{total\_iter-warm\_iter})^{0.9}$. The cross-entropy loss is adopted as the optimization objective.

In stage one of our UDA framework, unsupervised loss on the target domain is added to the optimization process. Specifically, shannon entropy loss, maximum square loss, neutral cross-entropy loss, and our threshold-adaptative unsupervised focal loss are adopted as the L_u. We first compare their performance with fixed confidence thresholds from 0.2 to 0.8 and then apply the dynamic threshold adjustment strategy to our unsupervised focal loss. The image perturbation methods are like [18], including random flipping, scaling, rotation, Gaussian noise, etc. The batch size is set as 2 for both source and target domain images and the initial learning rate is decayed as 0.5×10^{-5}, training for 20000 iterations.

In stage two of our UDA framework, mixed loss item L_m is added to the optimization loss. The main segmentation network for optimization $f(\theta)$ loads weights from the pretrained model on the source domain, while the segmentation network $f(\tilde{\theta})$ for producing the mixed image-label pairs is initialized from the weights of the model obtained in stage one. The batch size is also 2, with a learning rate of 0.5×10^{-5} and 20000 iterations. The λ_u and λ_m are set as 0.05 and 1.

In the testing process, we conduct experiments on the Cityscapes validation set to calculate the evaluation metrics of the UDA methods like most previous work. For DeepLabV2, we follow most of the settings in ProDA [16], and two 3090 GPUs are used for training and inference.

D. Experimental results

1) SYNTHIA-to-Cityscapes: The overall performance of our UDA method on the SYNTHIA-to-Cityscapes is shown in Table I. The baseline and oracle models are trained on the transferred source domain (IST) and target domain. The Shannon, Maximum, Neural represent shannon entropy loss, maximum square loss, and neural cross-entropy loss.

Our two-stage UDA method using BiSeNet reaches 52.2% mIoU of 16 classes and 59.1% mIoU* of 13 classes in SYNTHIA-to-Cityscapes, which are 11.4% and 12.6% higher than the baseline and comparable with the performance of some self-training methods [14], [15] using DeeplabV2. Threshold-adaptative unsupervised focal loss helps the model achieve 49.0% mIoU and 55.7% mIoU*, outperforming previously advanced neutral cross-entropy loss with 4.1% and 4.2%, and all other entropy-based UDA methods. It is supervised that maximum square loss degrades the performance in our experiments, implying that it relies on the image-wise weighting and multi-level self-guided approach [17]. The model has steady improvement on some difficult classes, such as sidewalk, wall, pole, traffic light, traffic sign, bus, and bike, indicating our method help to optimize the "hard-to-transfer" samples. When using the DeepLabV2, our method reaches 58.4% mIoU and 65.4% mIoU*, which are state-of-the-art and exceed previous adversarial-based [10], [11] and self-training [14]–[16] methods with healthy margins.

The qualitative adaptation results of different entropy-based UDA methods using BiSeNet are shown in Fig. 5, which is consistent with the quantitative results in Table I. Our two-stage UDA method helps the model make more explicit predictions in the target domain and performs better on categories like sidewalk and traffic sign. Meanwhile, the entropy maps produced by our method are much clearer, implying that many hard-to-transfer samples got optimized.

2) GTA5-to-Cityscapes: The overall performance of our UDA method on the GTA5-to-Cityscapes is shown in Table II.

Our two-stage UDA method achieves 55.4% mIoU of 19 classes and is 8.9% higher than the baseline. It brings steady improvement for low-IoU classes like wall, traffic light, traffic sign, truck, and bike. Meanwhile, the proposed threshold-adaptative unsupervised focal loss brings 6.8% mIoU improvement to the baseline, surpassing all previous entropy-based UDA methods. What's more, our method achieves state-of-the-art 59.6% mIoU when adopting DeepLabV2.

Fig. 6 shows the qualitative adaptation results, where our method performs better than other entropy-based UDA methods and demonstrates its effectiveness.

TABLE I
THE ADAPTATION PERFORMANCE AND COMPARISON ON SYNTHIA-TO-CITYSCAPES (MIOU: 16 CLASSES; MIOU*: 13 CLASSES)

Methods	Network	road	sw	build	wall*	fence*	pole*	light	sign	vege	sky	person	rider	car	bus	motor	bike	mIoU	mIoU*
AdaptSeg [10]	DeeplabV2 (ResNet101)	84.3	42.7	77.5	-	-	-	4.7	7.0	77.9	82.5	54.3	21.0	72.3	32.2	18.9	32.3	-	46.7
Shannon [8]		85.6	42.2	79.7	8.7	0.4	25.9	5.4	8.1	80.4	84.1	57.9	23.8	73.3	36.4	14.2	33.0	41.2	48.0
Maximum [17]		82.9	40.7	80.3	10.2	0.8	25.8	12.8	18.2	82.5	82.2	53.1	18.0	79.0	31.4	10.4	35.6	41.4	48.2
BDL [11]		86.0	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
FDA [27]		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
Confidence [19]		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
DACS [37]		80.6	25.1	81.9	24.5	2.9	37.2	22.7	24.0	83.7	90.8	67.6	38.3	82.9	38.9	28.5	47.6	48.3	54.8
Rectifying [14]		87.6	41.9	83.1	14.7	1.7	36.2	31.3	19.9	81.6	80.6	63.0	21.8	86.2	40.7	23.6	53.1	47.9	54.9
IAST [15]		81.9	41.5	83.3	17.7	4.6	32.3	30.9	28.8	83.4	85.0	65.5	30.8	86.5	38.2	33.1	52.7	49.8	57.0
SAC [29]		89.3	47.2	**85.5**	26.5	1.3	43.0	45.5	32.0	87.1	89.3	63.6	25.4	86.9	35.6	30.4	53.0	52.6	59.3
CorDA [33]		**93.3**	**61.6**	85.3	19.6	**5.1**	37.8	36.6	42.8	84.9	90.4	69.7	**41.8**	85.6	38.4	32.6	**53.9**	55.0	62.8
ProDA [16]		87.8	45.7	84.6	37.1	0.6	44.0	**54.6**	37.0	88.1	84.4	74.2	24.3	88.2	51.1	40.5	45.6	55.5	62.0
Ours		88.6	52.4	**85.5**	**39.4**	0.3	44.9	51.4	**60.3**	88.1	88.1	**75.5**	28.6	**88.7**	**52.5**	**42.2**	48.2	**58.4**	**65.4**
Oracle	BiSeNet (ResNet18)	85.1	74.0	80.2	46.4	49.7	50.7	55.1	64.8	89.4	66.0	70.0	47.2	83.5	72.6	45.2	66.5	65.4	69.2
Baseline (IST)		64.0	34.8	65.3	10.2	0.7	37.5	24.7	37.5	85.1	87.9	53.2	19.1	70.5	14.3	9.2	38.7	40.8	46.5
Shannon [8]		84.9	41.9	80.0	5.1	0.7	37.1	25.4	37.5	85.2	87.4	52.8	19.7	71.9	17.9	9.4	34.9	43.2	49.9
Maximum [17]		51.2	31.2	57.8	8.6	0.3	42.0	28.2	35.4	86.5	88.9	56.9	16.8	72.4	15.8	9.5	34.3	39.7	45.0
Confidence [19]		85.9	44.0	78.2	6.0	0.6	36.8	22.1	32.5	85.7	87.1	54.9	19.7	81.5	23.0	9.7	33.4	43.8	50.6
Neutral [18]		85.7	46.6	81.6	10.7	0.7	38.1	23.9	37.2	86.1	88.1	54.7	19.5	74.1	25.9	9.5	36.5	44.9	51.5
Ours(stage 1)		89.8	52.0	83.9	14.9	0.5	44.2	32.1	42.3	87.6	89.5	58.6	19.0	84.4	34.3	11.8	39.5	49.0	55.7
Ours(stage 2)		87.3	53.4	82.9	20.5	0.4	**45.6**	45.5	58.7	**88.6**	**91.7**	66.4	25.3	79.2	29.3	13.5	46.3	52.2	59.1

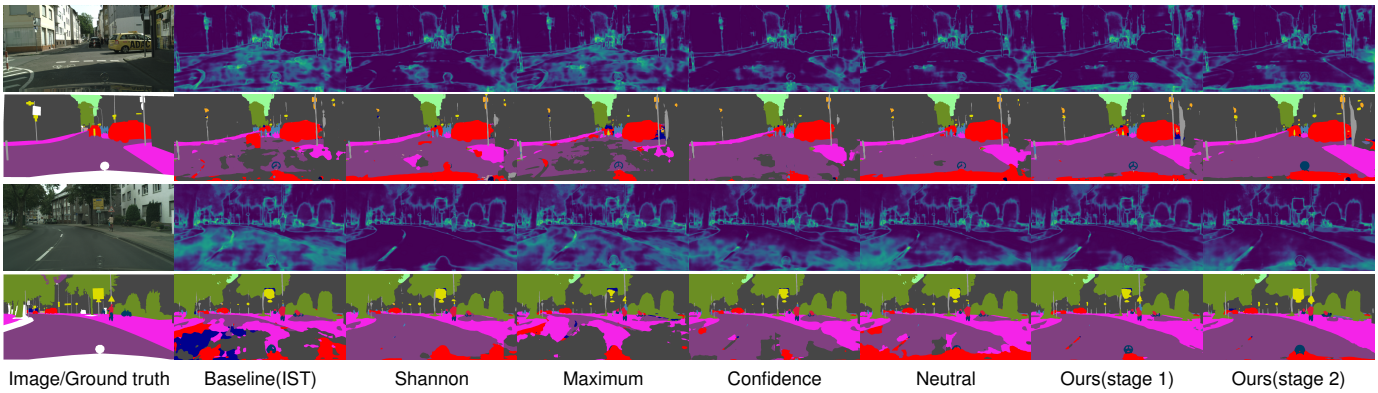


Fig. 5. Qualitative adaptation results of different entropy-based UDA methods using BiSeNet on SYNTHIA-to-Cityscapes. From left to right are images/ground truth, the predictions of baseline, shannon entropy loss, maximum square loss, neutral cross-entropy loss, and our two-stage UDA method. The second to seventh columns of the first and third rows are the entropy maps of the prediction results of corresponding methods.

E. Ablation Study

1) The improvement of each method: We gradually add unsupervised focal loss (focal), class-level dynamic threshold adjustment strategy (threshold), and cross-domain image mixing (CIM) to the baseline (IST) to explore their performance. The results on SYNTHIA-to-Cityscapes are shown in Table III. The unsupervised focal loss brings 5.1% mIoU improvement, the dynamic threshold adjustment strategy further boosts 3.1% mIoU, while the CIM improves the mIoU to 52.2%, demonstrating their effectiveness.

2) The coefficient γ of unsupervised focal loss: The γ in unsupervised focal loss regularizes the loss contribution of easy-to-transfer samples. We fix the confidence threshold to 0.8 and set γ to 1.0, 2.0, and 3.0. Table IV presents the experimental results, where $\gamma = 2$ performs best and is selected in our experiments.

3) The parameters of class-level dynamic threshold adjustment strategy: We set $a = 0.9$ and gradually tune the b and d, which is shown in Table V. The first row where $a = 1.0$ denotes the unsupervised focal loss with a fixed confidence threshold of 0.8. When increasing the d from 4 to 10, the mIoU first increases then decreases, which indicates that we need to regularize the thresholds of the "hard" classes to some extent, but not excessively. Meanwhile, bigger b (0.9) may bring more noise, while lower b (0.5) makes fewer pixels be included in loss calculation, and the middle one (0.8) is finally adopted. Nevertheless, the class-level dynamic threshold adjustment strategy can improve the adaptation performance within the appropriate parameter range, only needing fine-tuning to the best. Finally, a, b, and d are set as 0.9, 0.8, and 8.

4) Fixed threshold vs. dynamic threshold: We set the confidence threshold to 0.2, 0.4, 0.6, 0.8, and the results are presented in Table VI. It can be observed that a fixed threshold is hard to balance noise suppression (high threshold) and incorporating more pixels into the loss calculation (low threshold), making it hard for entropy-based UDA methods to achieve advanced performance. Our unsupervised focal loss performs better at all fixed thresholds and further increases the mIoU* to 55.7 with the dynamic threshold adjustment strategy.

5) The coefficients of loss function: In stage one, we set the coefficient λ_u of unsupervised loss L_u to 0.1, 0.05, 0.01, 0.005, and the results are shown in Fig. 7. Shannon entropy loss, maximum square loss, neutral cross-entropy loss, and our threshold-adaptative unsupervised focal loss peak at 0.01, 0.005, 0.01, and 0.05, respectively. It indicates that entropy-based losses essentially act as the regularization term. In stage

TABLE II
THE ADAPTATION PERFORMANCE AND COMPARISON ON GTA5-TO-CITYSCAPES

Methods	Network	road	sw	build	wall	fence	pole	light	sign	vege	terrain	sky	person	rider	car	truck	bus	train	motor	bike	mIoU
AdaptSeg [10]	DeeplabV2 (ResNet101)	86.5	36.0	79.9	23.4	23.3	23.9	35.2	14.8	83.4	33.3	75.6	58.5	27.6	73.7	32.5	35.4	3.9	30.1	28.1	42.4
Shannon [8]		89.4	33.1	81.0	26.6	26.8	27.2	33.5	24.7	83.9	36.7	78.8	58.7	30.5	84.8	38.5	44.5	1.7	31.6	32.4	45.5
Maximum [17]		89.4	43.0	82.1	30.5	21.3	30.3	34.7	24.0	85.3	39.4	78.2	63.0	22.9	84.6	36.4	43.0	5.5	34.7	33.5	46.4
BDL [11]		91.0	44.7	84.2	34.6	27.6	30.2	36.0	36.0	85.0	43.6	83.0	58.6	31.6	83.3	35.3	49.7	3.3	28.8	35.6	48.5
Confidence [19]		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
Rectifying [14]		90.4	31.2	85.1	36.9	25.6	37.5	48.8	48.5	85.3	34.8	81.1	64.4	36.8	86.3	34.9	52.2	1.7	29.0	44.6	50.3
FDA [27]		92.5	53.3	82.4	26.5	27.6	36.4	40.6	38.9	82.3	39.8	78.0	62.6	34.4	84.9	34.1	53.1	16.9	27.7	46.4	50.5
DACS [37]		89.9	39.7	**87.9**	30.7	39.5	38.5	46.4	52.8	88.0	44.0	88.8	67.2	35.8	84.5	45.7	50.2	0.0	27.3	34.0	52.1
IAST [15]		94.1	58.8	85.4	39.7	29.2	25.1	43.1	34.2	84.8	34.6	88.7	62.7	30.3	87.6	42.3	50.3	**24.7**	35.2	40.2	52.2
SAC [29]		90.4	53.9	86.6	42.4	27.3	45.1	48.5	42.7	87.4	40.1	86.1	67.5	29.7	88.5	49.1	54.6	9.8	26.6	45.3	53.8
CorDA [33]		**94.7**	**63.1**	87.6	30.7	40.6	40.2	47.8	51.6	87.6	47.0	89.7	66.7	35.9	**90.2**	48.9	57.5	0	39.8	56.0	56.6
ProDA [16]		87.8	56.0	79.7	46.3	**44.8**	45.6	53.5	53.5	88.6	45.2	82.1	70.7	39.2	88.8	45.5	**59.4**	1.0	48.9	56.4	57.5
Ours		92.0	59.1	84.6	**48.0**	40.3	**48.0**	**55.1**	**61.7**	**89.1**	**51.2**	84.0	**72.6**	**43.4**	87.5	50.9	51.2	6.4	**50.9**	55.6	**59.6**
Oracle	BiSeNet (ResNet18)	95.6	79.4	87.6	53.5	50.7	52.3	57.9	67.0	90.4	58.0	91.7	71.5	47.8	91.1	59.5	72.7	57.5	45.6	66.7	68.2
Baseline (IST)		88.8	49.0	84.8	32.7	23.7	36.3	42.7	40.6	83.4	31.7	84.9	59.6	24.4	84.5	33.8	39.1	4.1	13.2	26.4	46.5
Shannon [8]		89.8	49.2	84.8	34.6	22.3	36.0	42.1	37.5	84.7	33.8	87.1	60.0	24.5	86.1	32.2	41.8	4.0	18.6	33.8	47.5
Maximum [17]		87.1	44.7	85.0	33.3	25.3	37.6	42.9	41.0	84.1	30.9	86.0	60.9	21.6	85.7	31.8	38.8	4.7	18.1	31.0	46.9
Confidence [19]		90.9	48.0	84.9	33.7	25.1	38.3	43.6	40.6	85.4	36.9	83.9	60.1	24.2	85.1	27.6	32.8	5.1	17.2	41.4	47.6
Neutral [18]		90.9	52.9	86.1	38.5	24.5	41.5	46.7	42.1	86.7	37.0	86.4	63.3	24.2	88.1	36.8	42.4	2.3	19.0	30.4	49.5
Ours(stage 1)		91.7	57.2	87.4	39.3	29.9	42.7	51.7	57.3	87.3	39.7	89.9	63.2	26.3	88.5	46.8	53.1	2.1	18.5	41.0	53.3
Ours(stage 2)		89.8	54.9	87.1	47.0	34.0	41.8	53.0	58.6	86.6	38.4	**92.3**	57.7	28.3	87.3	**51.0**	50.2	9.3	26.7	**58.3**	55.4

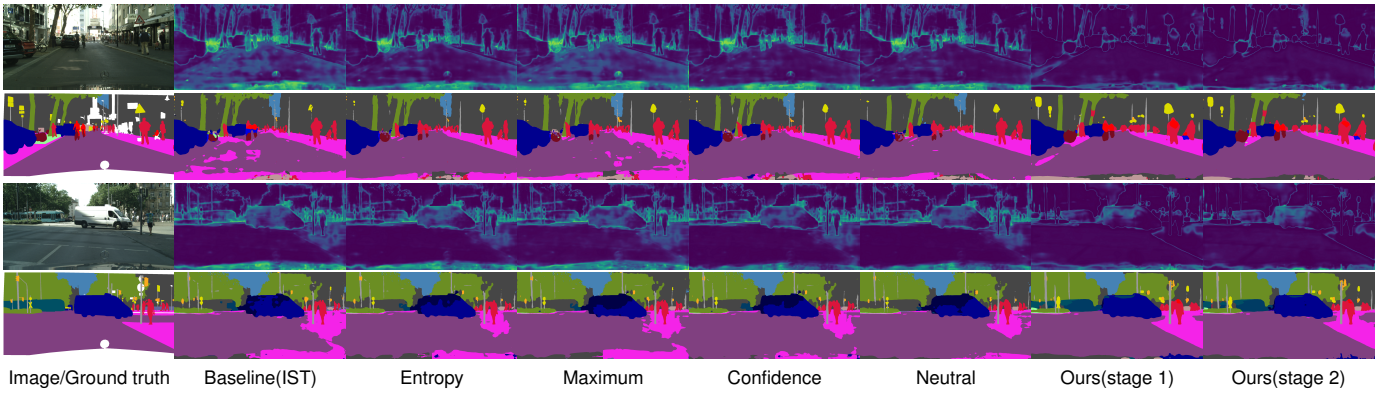


Fig. 6. Qualitative adaptation results of different entropy-based UDA methods using BiSeNet on GTA5-to-Cityscapes. From left to right are images/ground truth, the predictions of baseline, shannon entropy loss, maximum square loss, neutral cross-entropy loss, and our two-stage UDA method. The second to seventh columns of the first and third rows are the entropy maps of the prediction results of corresponding methods.

TABLE III
IMPROVEMENT OF EACH METHOD ON SYNTHIA-TO-CITYSCAPES

focal	threshold	mixing	mIoU	mIoU*
			40.8	46.5
√			45.9	52.6
√	√		49.0	55.7
√	√	√	**52.2**	**59.1**

TABLE IV
THE PERFORMANCE OF DIFFERENT γ ON SYNTHIA-TO-CITYSCAPES

γ	1.0	2.0	3.0
mIoU	45.5	**45.9**	44.5
mIoU*	52.2	**52.6**	51.1

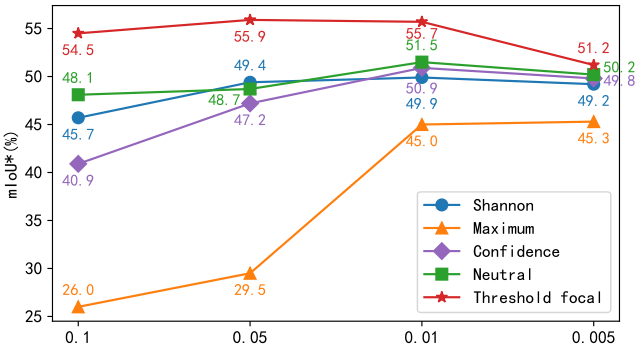


Fig. 7. The performance of different λ_u on SYNTHIA-to-Cityscapes.

two, we set the coefficient λ_m of the mixed loss to 0.4, 0.6, 1.0, 1.4, and 1.6, and Fig. 8 presents the results. With the increase of λ_m, the performance of the model first improves and then degrades, and finally, 1.0 is selected in our experiments.

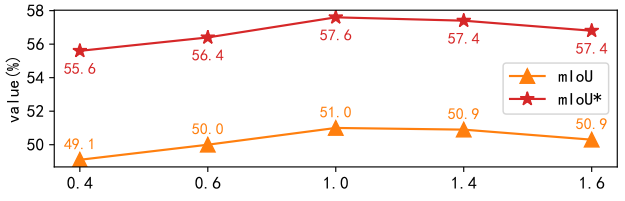


Fig. 8. The performance of different λ_m on SYNTHIA-to-Cityscapes.

F. Comparison of UDA method details

Table VII exhibits some details of our UDA method and the state-of-the-art ProDA [16]. It can be seen that DeeplabV2 adapts better than the BiSeNet, but will bring more computation, parameters and memory usage, and take more time

TABLE V
THE PERFORMANCE OF DIFFERENT PARAMETERS IN DYNAMIC THRESHOLD ADJUSTMENT STRATEGY ON SYNTHIA-TO-CITYSCAPES

a	b	d	mIoU	mIoU*
1.0	-	-	45.9	52.6
0.9	0.8	4	47.5	54.0
0.9	0.8	8	**48.4**	**55.7**
0.9	0.8	10	46.9	53.4
0.9	0.9	8	46.8	53.2
0.9	0.5	8	45.2	51.7

TABLE VI
THE PERFORMANCE OF FIXED THRESHOLD AND DYNAMIC THRESHOLD ON SYNTHIA-TO-CITYSCAPES

Method	mIoU*				
	0.2	0.4	0.6	0.8	dynamic
Shannon	50.2	50.6	49.4	49.9	-
Maximum	41.8	43.5	42.7	45.0	-
Confidence	47.8	49.4	50.5	50.9	-
Neutral	51.5	50.5	51.6	51.5	-
Focal	**51.9**	**51.7**	**52.8**	**52.6**	**55.7**

TABLE VII
THE DETAILS OF OUR UDA METHOD AND PRODA

Method	Our		ProDA [16]
Model	BiSeNet	DeepLabV2	DeepLabV2
MACs (G)	**121**	2172	2172
Params (M)	**13**	65	65
Disk usage (MB)	**54**	261	261
Training time (h)	**7**	87	104
Inference time (ms)	**28**	94	94
mIoU from GTA5 (%)	55.4	**59.6**	57.5
mIoU from SYNTHIA (%)	52.2	**58.4**	55.5

for training and inference. Due to the optimization of hard samples, our method converges faster than ProDA and achieves higher mIoUs. Moreover, our two-stage method only needs about 7 hours for training on two 3090 GPUs and 28 ms for inference on a single one when using BiSeNet, dramatically reducing the training time and enabling real-time inference.

G. Discussion

Experiments on two synthetic-to-real settings verify the effectiveness of our methods. Here we discuss the relationship of our methods with the self-training methods.

The shannon entropy loss can be seen as a soft-assignment version of the pseudo label cross-entropy [8], which tends to make prediction probabilities at 0 and 1, making p_t $(\hat{p}_t)$ be like pseudo labels in the target domain. Then KL divergence measures the prediction differences $(\hat{p}_t, p_{t^*})$ between perturbed image pairs, which is similar to making predictions of target samples to be like the pseudo labels, but in a soft manner. Threshold-adaptative unsupervised focal loss adopts loss regularization and dynamic threshold adjustment strategy to optimize hard samples. In stage two, pseudo labels are generated for CIM. What we focus on is bridging the semantic knowledge between two domains while not using pseudo labels to re-train the model in the target domain. Meanwhile, our method only contains two-stage of training, unlike self-training methods that usually require iterative training. Moreover, self-training methods can be used to further adapt the model from stage two and will be our future work.

V. CONCLUSION

In this paper, we propose a two-stage UDA framework for domain adaptation of semantic segmentation. In stage one, we design the threshold-adaptative unsupervised focal loss with a class-level dynamic threshold adjustment strategy, which helps optimize hard samples and outperforms all previous entropy-based methods. In stage two, we introduce CIM with long-tail class pasting to bridge the semantic knowledge between two domains, further boosting the adaptation performance. Extensive experiments on two synthetic-to-real benchmarks demonstrate that our method achieves state-of-the-art. In the future, we will explore the adaptation performance of our method on more semantic segmentation networks.

REFERENCES

[1] L. Deng, M. Yang, H. Li, T. Li, B. Hu, and C. Wang, "Restricted deformable convolution-based road scene semantic segmentation using surround view cameras," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 10, pp. 4350–4362, 2019.

[2] Y. Qian, L. Deng, T. Li, C. Wang, and M. Yang, "Gated-residual block for semantic segmentation using rgb-d data," *IEEE Transactions on Intelligent Transportation Systems*, 2021.

[3] K. Yang, X. Hu, L. M. Bergasa, E. Romera, and K. Wang, "Pass: Panoramic annular semantic segmentation," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 10, pp. 4171–4185, 2019.

[4] E. Romera, J. M. Alvarez, L. M. Bergasa, and R. Arroyo, "Erfnet: Efficient residual factorized convnet for real-time semantic segmentation," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 1, pp. 263–272, 2017.

[5] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The cityscapes dataset for semantic urban scene understanding," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2016, pp. 3213–3223.

[6] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2016, pp. 3234–3243.

[7] S. R. Richter, V. Vineet, S. Roth, and V. Koltun, "Playing for data: Ground truth from computer games," in *European Conference on Computer Vision*. Springer, 2016, pp. 102–118.

[8] T.-H. Vu, H. Jain, M. Bucher, M. Cord, and P. Pérez, "Advent: Adversarial entropy minimization for domain adaptation in semantic segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 2517–2526.

[9] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 2223–2232.

[10] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 7472–7481.

[11] Y. Li, L. Yuan, and N. Vasconcelos, "Bidirectional learning for domain adaptation of semantic segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 6936–6945.

[12] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," *Advances in Neural Information Processing Systems*, vol. 27, 2014.

[13] Y. Zou, Z. Yu, B. Kumar, and J. Wang, "Unsupervised domain adaptation for semantic segmentation via class-balanced self-training," in *Proceedings of the European Conference on Computer Vision*. Springer, 2018, pp. 289–305.

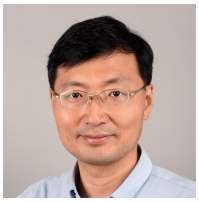
[14] Z. Zheng and Y. Yang, "Rectifying pseudo label learning via uncertainty estimation for domain adaptive semantic segmentation," *International Journal of Computer Vision*, vol. 129, no. 4, pp. 1106–1120, 2021.

[15] K. Mei, C. Zhu, J. Zou, and S. Zhang, "Instance adaptive self-training for unsupervised domain adaptation," in *European Conference on Computer Vision*. Springer, 2020, pp. 415–430.
[16] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 12 414–12 424.
[17] M. Chen, H. Xue, and D. Cai, "Domain adaptation for semantic segmentation with maximum squares loss," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 2090–2099.
[18] H. Xu, M. Yang, L. Deng, Y. Qian, and C. Wang, "Neutral cross-entropy loss based unsupervised domain adaptation for semantic segmentation," *IEEE Transactions on Image Processing*, vol. 30, pp. 4516–4525, 2021.
[19] X. Zhang, Y. Chen, Z. Shen, Y. Shen, H. Zhang, and Y. Zhang, "Confidence-and-refinement adaptation model for cross-domain semantic segmentation," *IEEE Transactions on Intelligent Transportation Systems*, 2022.
[20] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3431–3440.
[21] 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 Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 4, pp. 834–848, 2017.
[22] J. Wang, K. Sun, T. Cheng, B. Jiang, C. Deng, Y. Zhao, D. Liu, Y. Mu, M. Tan, X. Wang *et al.*, "Deep high-resolution representation learning for visual recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 10, pp. 3349–3364, 2020.
[23] E. Xie, W. Wang, Z. Yu, A. Anandkumar, J. M. Alvarez, and P. Luo, "Segformer: Simple and efficient design for semantic segmentation with transformers," *Advances in Neural Information Processing Systems*, vol. 34, 2021.
[24] C. Yu, J. Wang, C. Peng, C. Gao, G. Yu, and N. Sang, "Bisenet: Bilateral segmentation network for real-time semantic segmentation," in *Proceedings of the European Conference on Computer Vision*. Springer, 2018, pp. 325–341.
[25] G. Dong, Y. Yan, C. Shen, and H. Wang, "Real-time high-performance semantic image segmentation of urban street scenes," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 6, pp. 3258–3274, 2020.
[26] M. Fan, S. Lai, J. Huang, X. Wei, Z. Chai, J. Luo, and X. Wei, "Rethinking bisenet for real-time semantic segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 9716–9725.
[27] Y. Yang and S. Soatto, "Fda: Fourier domain adaptation for semantic segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 4085–4095.
[28] H. Ma, X. Lin, Z. Wu, and Y. Yu, "Coarse-to-fine domain adaptive semantic segmentation with photometric alignment and category-center regularization," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 4051–4060.
[29] N. Araslanov and S. Roth, "Self-supervised augmentation consistency for adapting semantic segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 15 384–15 394.
[30] E. Romera, L. M. Bergasa, K. Yang, J. M. Alvarez, and R. Barea, "Bridging the day and night domain gap for semantic segmentation," in *2019 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2019, pp. 1312–1318.
[31] X. Wu, Z. Wu, H. Guo, L. Ju, and S. Wang, "Dannet: A one-stage domain adaptation network for unsupervised nighttime semantic segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 15 769–15 778.
[32] W. Zhou, Y. Wang, J. Chu, J. Yang, X. Bai, and Y. Xu, "Affinity space adaptation for semantic segmentation across domains," *IEEE Transactions on Image Processing*, vol. 30, pp. 2549–2561, 2020.
[33] Q. Wang, D. Dai, L. Hoyer, L. Van Gool, and O. Fink, "Domain adaptive semantic segmentation with self-supervised depth estimation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 8515–8525.
[34] L. Hoyer, D. Dai, and L. Van Gool, "Daformer: Improving network architectures and training strategies for domain-adaptive semantic segmentation," *arXiv preprint arXiv:2111.14887*, 2021.
[35] O. Chapelle and A. Zien, "Semi-supervised classification by low density separation," in *International Workshop on Artificial Intelligence and Statistics*. PMLR, 2005, pp. 57–64.
[36] V. Olsson, W. Tranheden, J. Pinto, and L. Svensson, "Classmix: Segmentation-based data augmentation for semi-supervised learning," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2021, pp. 1369–1378.
[37] W. Tranheden, V. Olsson, J. Pinto, and L. Svensson, "Dacs: Domain adaptation via cross-domain mixed sampling," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2021, pp. 1379–1389.
[38] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 2980–2988.
[39] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2016, pp. 770–778.

Weihao Yan received the B.S. degree in automation from Shanghai Jiao Tong University, Shanghai, China, in 2020. He is currently working toward the Ph.D. degree in Control Science and Engineering with Shanghai Jiao Tong University.
His main research interests include computer vision, image processing, deep learning and their applications in intelligent transportation systems.

Yeqiang Qian received the Ph.D. degree in Control Science and Engineering from Shanghai Jiao Tong University, Shanghai, China, in 2020. He is currently a postdoctoral fellow with University of Michigan-Shanghai Jiao Tong University Joint Institute in Shanghai Jiao Tong University.
His main research interests include computer vision, pattern recognition, machine learning and their applications in intelligent transportation systems.

Chunxiang Wang received the Ph.D. degree in mechanical engineering from the Harbin Institute of Technology, Harbin, China, in 1999.
She is currently an Associate Professor with the Department of Automation at Shanghai Jiao Tong University, Shanghai, China. Her research interests include robotic technology and electromechanical integration.



Ming Yang received the Master and Ph.D. degrees from Tsinghua University, Beijing, China, in 1999 and 2003, respectively.
He is currently the Full Tenure Professor at Shanghai Jiao Tong University, the deputy director of the Innovation Center of Intelligent Connected Vehicles. He has been working in the field of intelligent vehicles for more than 20 years. He participated in several related research projects, such as the THMR-V project (first intelligent vehicle in China), European CyberCars and CyberMove projects, CyberC3 project, CyberCars-2 project, ITER transfer cask project, AGV, etc.