Synthetic to Realistic Imbalanced Domain Adaption for Urban Scene Perception

Yining Hua and Dewei Yi, *Member, IEEE*

***Abstract*—Deep neural networks technique has achieved impressive performance on semantic segmentation, while its training process requires a large amount of pixel-wise labeled data. Domain adaptation, as a promising solution, can break the restriction by training the model on synthetic data, and generalizing it in real-world data. However, there is still a lack of attention paid to the imbalance problems on semantic segmentation adaptation, including the imbalance problem between 1) source and target data and 2) different classes. To solve these problems, a progressive hierarchical feature alignment method is proposed in this article. To alleviate the data imbalance problem, the network is progressively trained by the data from multisource domains, so as to obtain domain-invariant features. To address the class imbalance problem, the features are aligned hierarchically across domains. According to the experimental results, our method shows the competitive adapted segmentation performance on three benchmark datasets.**

***Index Terms*—Convolution neural networks, deep learning, domain adaptation, image segmentation.**

I. Introduction

Deep neural networks (DNNs) have achieved remarkable performance in computer vision, especially in semantic segmentation [1]. Semantic segmentation is to assign the prediction of each pixel in an image. However, a recent work [2] shows that DNNs cannot generalize well in unseen environments. One intuitive idea is to train a segmentation model with more labeled data from the unseen environment. This straightforward idea is not very realistic in practice due to the high cost of obtaining pixel-wise manual annotations. To tackle the issue, unsupervised domain adaptation algorithms are introduced into semantic segmentation tasks for moving one step closer to real-world practice. The purpose of domain adaptation for semantic segmentation is to train a segmentation network on the data and labels in the source domain and generalize well in the target domain.

Most of the works on semantic segmentation adaptation attempt to minimize the discrepancy of data distribution across domains. Two main-stream methods are identified for this task. In the first stream, many existing studies align two domains by minimizing the distribution discrepancy from various aspects. Pixel-wise alignment between source and target domains is investigated in [3]–[5]. Feature-level alignment across domains is explored in [6]–[8]. In addition, the work of [9], [10] is to align semantic classes from the source domain to the target domain. Despite the great success this stream has achieved so far, the work of this stream cannot guarantee an optimal solution due to the neglect of domain-specific knowledge. In the second stream, many methods attempt to extract the knowledge of unlabeled target domain data. Specifically, the methods of this stream usually adopt a two-step pipeline, which is similar to the traditional semi-supervised framework [11]. The first step is to predict pseudo-labels by utilizing the knowledge learnt from the labeled data, e.g., the model trained on the source domain. The second step is to minimize the loss on the pseudo-labels of the unlabeled target domain data. In the training process, pseudo-labels are usually regarded as accurate annotations to optimize the model. However, this arises one inherent problem. Pseudo-labels usually suffer from the noise caused by the model trained on different data distributions. To deal with this problem, Zou *et al.* [12] ignore pseudo-labels below a specific confidence threshold. Our method takes full advantage of both streams above. The discrepancy is minimized at different levels and domain-specific knowledge of unlabeled target data is fully exploited through self-learning learning.

As discussed in [13], although unsupervised domain adaptation algorithms do not need labels of target domain data, to achieve promising performance, it requires a large number of unlabeled data from the target domain for the training purpose. However, it is hard to guarantee that there are enough target domain data available. This can be formulated as a few-shot unsupervised domain adaptation problem, where there are a large amount of data in the source domain, and only a few shots of data are available in the target domain. To deal with this issue, we propose a progressive hierarchical feature alignment method on domain adaptation for semantic segmentation. The data from multisource domains are trained progressively to obtain domain-invariant features. A more comprehensive cross-domain alignment is realized by a hierarchical feature alignment scheme, where all the objects, categories, and images are taken into account to achieve better alignment from the source domain

Manuscript received July 2, 2021; accepted August 11, 2021. Date of publication August 25, 2021; date of current version February 2, 2022. This work was supported by the University of Aberdeen Internal Funding to Pump-Prime Interdisciplinary Research and Impact under Grant SF10206-57. Paper no. TII-21-2798. *(Corresponding author: Dewei Yi.)*

Yining Hua is with the School of Arts, University of Roehampton, SW15 5PH London, U.K. (e-mail: yining.hua@roehampton.ac.uk).

Dewei Yi is with the Department of Computing Science, University of Aberdeen, AB24 3UE Aberdeen, U.K. (e-mail: dewei.yi@abdn.ac.uk).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TII.2021.3107785.

Digital Object Identifier 10.1109/TII.2021.3107785

to the target domain. The main contributions of this article are summarized as follows.

1) To learn domain-invariant features, a progressive multi-source adversarial domain adaptation method is adapted to extract domain-invariant by using synthetic data from different simulators (e.g., GTA5, Unity).
2) In practice, it is difficult to guarantee the availability of sufficient data from the target domain, where there are a large number of data in the source domain while only a few shots of data are available for the target domain. To the best of our knowledge, in urban driving scene, this is the first attempt to tackle the few-shot unsupervised domain adaptation for semantic segmentation.
3) A hierarchical feature alignment scheme is proposed to align object-level, category-level, and image-level features across domains along with self-supervised learning to enhance the performance of adapted segmentation.
4) To evaluate the performance of our proposed method, extensive experiments are conducted to adapt from synthetic GTA5 and SYNTHIA datasets to real-world Cityscapes dataset. Many advanced methods are compared with our proposed method on the scenario that only a few shots of data are available for the target domain. The experimental results demonstrate the superiority of our proposed method along with competing with other existing methods.

II. Related Work

In this section, important work about the three most related tasks are broadly discussed, i.e., 1) domain adaptation for semantic segmentation, 2) the imbalanced problems in domain adaptation, and 3) multisource domain adaptation.

A. Domain Adaptation for Semantic Segmentation

Since labeling a large amount of pixel-level data is a labor-intensive work, training networks on automatically labeled virtual data become a promising solution to alleviate the efforts of manual annotation. However, there exists a gap between virtual data and real-world data, which makes the segmentation networks trained on the virtual data cannot generalize well in real-world data. To this end, domain adaptation is introduced to semantic segmentation, to obtain better generalization ability when human intervention is reduced. By minimizing the discrepancy between source and target domains, adapted semantic segmentation can achieve promising performance in the target data after training a model with labeled source data along with unlabeled target data.

Recently, methods related to adversarial learning are treated as a promising way to bridge the gap across domains, such as [3], [5], [14]–[17]. Studies on [14] and [3] achieve the alignment of feature space latent representations across domains. In [5] and [15], input-level adaption is enforced for minimizing the visual difference of different domains. The work of [9] adapts the semantic predictions across domains by using output-feature space discriminators. As mentioned [18], previous GAN-style methods focus on minimizing the appearance difference between generated features and target domain features. One insight is observed that the appearance of background classes is similar to each other. This should be noticed during adversarial domain adaptation. The naïve combination of the image-transferring model and segmentation model is insufficient to minimize the gap across domains. This is because the quality of segmentation is impaired a lot when there exist the failures of image style translation across domains. To further improve the generalization of cross-domain semantic segmentation, Li *et al.* [18] introduce bidirectional learning to help CycleGAN retain local semantic information when carrying out the unpaired image style translation and also propose a self-training approach to generate pseudo-labels for target data. The work of [19] attempts to align different domains by both considering the local regions of an image and the entire image. Tao *et al.* [20] enhance the performance of semantic segmentation by taking the predictions of multiscales into account. However, the alignments of different feature levels, e.g., object-level, category-level, and image-level, are not being paid enough attention. Therefore, we propose a hierarchical feature alignment scheme to generalize well from source to target data.

B. Imbalanced Problems in Domain Adaptation

In this article, we focus on two kinds of imbalanced problems: 1) the class imbalance problem and 2) the imbalance between source and target data. The class imbalance problem occurs when the respective numbers of data for different classes are imbalanced. The imbalanced dataset is with a long-tailed class distribution. This problem is more severe for the pixel-level category prediction of an image, which is known as semantic segmentation. For the perception of urban scenes, Cityscapes [21] is a commonly used dataset to assess the performance of semantic segmentation. This dataset has many samples on the classes of road and sky, which are defined as head classes in the class distribution, while there are significantly fewer samples for traffic signs, which are defined as one of the tail classes in the class distribution. If a model is trained on an imbalanced dataset, it will be skewed to the head classes [22]. When there is a large amount of data available from the source domain but much fewer from the target domain, the imbalance between source and target data occurs, which would bring a big challenge in source to target domain alignment.

To handle the class imbalance issue, some approaches have been proposed to rebalance the classes [22], [23]. For example, for each class, Cui *et al.* [22] use an effective number of samples to calculate the class-balanced loss and then rebalance such a loss. In [23], a cut-and-paste approach is proposed to increase the amount of tail-class training data. However, without target labels, these methods cannot be applied directly to unsupervised semantic segmentation. In our method, the class imbalance problem is alleviated by introducing maximum square loss. Moreover, we attempt to make pioneering efforts on the cross-domain data imbalance issue. It is because there is not much related work done in the literature.

C. Multisource Domain Adaptation

By using multisource data, multisource to single target adaption can achieve better generalization in the target domain data. For example, in the literature [24], the distribution shifts are adjusted by a generalization bound, which is found by leveraging heuristic algorithms. Zhao *et al.* [25] propose a certain *ad hoc* scheme, which combines coefficients α to implement multisource domain adaption. Additionally, multiple domain-matching network in [26] computes domain similarities on both source-to-target domain and within-the-source domain based on the Wasserstein-like measure. Nonetheless, calculating such pairwise weights can be computationally demanding, especially when there are a lot of source domains. Their bound requires additional smooth assumptions on the labeling functions f_{S_i} and f_T. Thus, unlike existing work, multisource data in this article are learnt progressively to obtain auxiliary information, which helps extract the domain-invariant features.

III. Progressive Hierarchical Feature Alignment

This section introduces the details of our proposed progressive hierarchical feature alignment method on the domain adaptation for semantic segmentation. To bridge the gap between the source domain and target domain, we first enforce the progressive adversarial domain adaptation by multisource data. Consequently, auxiliary information can be obtained to enable preliminary domain alignment and extract domain-invariant features. A detailed description of progressive adversarial domain adaptation is provided in Section III-B. Second, features from different domains are aligned hierarchically to carry out a more subtle alignment. Cross-domain features are aligned from low level to high level. In Section III-C, a hierarchical feature alignment scheme is elaborated. Third, Section III-D describes a self-guidance framework, which is introduced into our proposed method to achieve the label-level transferring. Finally, the full objective and the entire framework of our proposed method are presented in Section III-E and Fig. 1, respectively.

A. Problem Formulation

We consider the unsupervised multisource domain adaptation with only a few shots of data available in the target domain. In this case, there are multiple labeled source domains. Source domains $X_S^1, X_S^2, \ldots, X_S^M$ and the corresponding ground truth Y_S are given, where the ith image of the source domain X_S is defined as x_s^i. In addition, only a few shots of target domain images are given without labels from a small target set X_T. The aim is to learn a generative model G for transferring knowledge from the source domain to the target domain so that G can correctly predict semantic labels (e.g., road, building, sign, etc.) at the pixel level in the target domain. That is, an adaptation model trained on X_S^M, Y_S^M, and X_T can assign the correct labels for the target domain data.

B. Progressive Adversarial Domain Adaptation With Auxiliary Information

The progressive adversarial domain adaptation targets on learning domain-invariant feature representations. To achieve this, multisource domain data are involved in the adversarial learning of domain adaptation. First, the preliminary source data are used to train the domain adaptation model and proceed initial alignment from synthetic data to realistic data. In this way, basic feature representations are extracted, and they are used as the auxiliary information for more subtle alignment. Second, the primary source data are used to learn domain-invariant feature representation. This is a sequential learning process, so named “progressive learning.”

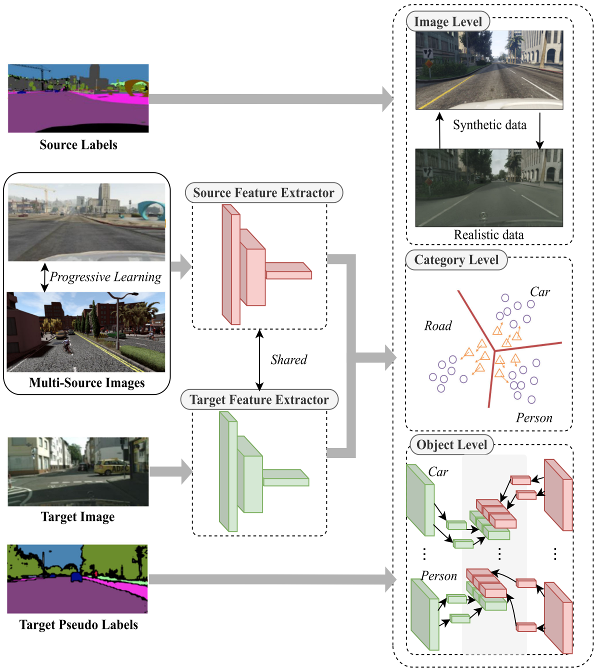


Fig. 1. Framework of the proposed progressive hierarchical feature alignment method.

C. Hierarchical Feature Alignment Scheme

Due to the diversity and complexity of data distribution of different domains, cross-domain feature alignment is a challenging task. To achieve a holistic representation of the mapping from the source domain to the target domain, it is not enough if we only conduct global-level alignment. The cross-domain mapping is required to be in different levels, e.g., object level for objects and foreground, category level for semantic classes and background, and image level for image style translation. Thus, we propose a novel hierarchical feature alignment scheme to map feature representation from source domain to target domain in a better manner, where the bottom-top alignment is conducted. In the low level, objects (e.g., cars or persons) of source and target domains are aligned individually. In the medium level, different categories of semantic classes are aligned to achieve better probability balance, the so-called category-level alignment, where class-balanced weighting factor is adopted for the sake of balancing the number of classes. In the high level, the image of source domains is transferred to the target domain image style through CycleGAN [18].

1) Object-Level Alignment: The object-level alignment focuses on the objects of foreground classes. These objects come from the classes of cars, persons, etc. Since the object-level annotations are not available, we follow [27] to generate the foreground object mask. In a label map, the foreground objects are found through identifying the disconnected regions of each foreground class. By using such a coarse segmentation, objects can be identified from intraclass semantic regions. Subsequently, various object-level feature representations can be extracted from an image by using (1).

$$L_{obj} = \sum_{i}\sum_{k\in K}\frac{1}{|T_k^t|}\sum_{r^t\in T_k^t}\min_{j}\left\|\frac{\sum\limits_{h,w} I_t^{(h,w)}F(x_i^t)^{(h,w)}}{\max(\epsilon,\sum_{(h,w)} I_t^{(h,w)})} - x_{j.k}^s\right\|_1^1 \tag{1}$$

where $i \in 1,\ldots,|X_T|$ and $T_k^t = \{I_{k_1}, I_{k_2}, I_{k_i}, \ldots, I_{k_m}\}$. I_{k_i} is the binary mask of the connected region with regard to ith target domain image x_i^t on class k, $k \in K$. T_k^t is the set of objects on kth class in the target domain. $x_{j,k}^s$ is the averaged features of kth class denoted in (1). With minimizing the loss of in (1), the object features of the closest intraclass sample in the source domain can be pushed to get closer to the object features of the target domain.

2) Category-Level Alignment and Balancing: Category-level alignment is to align the various semantic classes across domains. Entropy minimization method is one of the most popular approaches in semi-supervised learning, which is promising to be used in the semantic segmentation adaptation. However, conventional entropy minimization method has a problem that the gradient of entropy is overly concerned with easy-to-transfer classes. Adequate attention is not paid to the hard-to-transfer classes. As a result, the gradients of easy-to-transfer classes are much larger than the hard-to-transfer classes during the training process. To avoid the training process dominated by easy-to-transfer classes, the maximum square loss is adopted for balancing the probabilities of different classes. The maximum square loss has linear growth of gradient, which makes areas with higher confidence keep larger gradients while their dominant effects are suppressed for letting hard-to-transfer classes obtain training gradients. As a consequence, the alignment of various classes is conducted in a more balanced manner. In addition, there are more pixels about the easy-to-transfer classes on the label map and this situation causes an imbalance in quantity. Since labels are not available for target domain data in the unsupervised domain adaptation task, the class frequency of the target domain cannot be obtained, so the conventional weighting-based methods are not appropriate for this case.

To tackle the problem of missing the target domain label, each target image is used to compute the class frequency rather than using the whole data of target domain as given in (2).

$$L_{t,P^*}^{(h,w)} = \begin{cases} 1 & if\ P^* = \arg_c \max L_{t,P_i}^{(h,w)} \\ 0 & \text{otherwise} \end{cases}$$

$$N^c = \sum_{w=1}^{W}\sum_{h=1}^{H} L_{t,P^*}^{(h,w)}. \tag{2}$$

Taking the inaccurate predictions into account, the average loss of a target image relies on both the total number of pixel samples $(W \times H)$ and the number of classes N^c as shown in (3).

$$L_{\text{class}}(x_t) = -\sum_{w=1}^{W}\sum_{h=1}^{H}\sum_{c=1}^{C}\frac{(L_{t,P^*}^{(h,w)})^2}{2(N^c)^{\alpha} \times (W \times H)^{(1-\alpha)}} \tag{3}$$

where α is a hyperparameter and set as 0.2 as suggested in [28].

In addition, the feature representations of background semantic classes are extracted for enforcing category-level alignment. In contrast to foreground classes, the appearance of background classes is inclined to be invariant and occupying a big part of an image. The overlap of the predictions and ground truth is leveraged to generate the label map with corrected predictions, which is given in (4).

$$L_{C_i}^s = L_{G_i}^s \cap \{\arg_{k\in N}\max(G(x_i^s)^{(h,w)})^{(k)}\} \tag{4}$$

where the map of correct predictions is denoted by $L_{C_i}^s$. It is calculated by the overlap between the ground truth label map $L_{G_i}^s$ and the predicted label map, where the prediction of each pixel in an image is obtained by $G(x_i^s)$. The height and width of the feature map are denoted as h and w, respectively. k is the class type for the corresponding position in the feature map. The averaged features of the same background class are defined as the representations of background classes in (5)

$$x_{j,b}^s = \frac{\sum_{h,w}\Delta(L_{s,C_i}^{(h,w)} - b)F(x_i^s)^{(h,w)}}{\max(\tau, \sum_{h,w}\Delta(L_{s,C_i}^{(h,w)} - b))} \tag{5}$$

where $x_{j,b}^s$ is the jth semantic feature sample of class b in the source domain. The Dirac delta function is denoted as $\Delta(\bullet)$. If $x_{j,b}^s \neq 0$, then $j = i \bmod \zeta, b \in B, i \in \{1,\ldots,|X_S|\}$. The ζ represents the number of stored feature samples of each class and τ is the regularizing term. We minimize the distance between the features of each background class in the target domain and its closest intraclass features in the source domain. The feature representation of each background class is obtained by using the predicted label map due to the lack of the ground truth on the target images. The source-to-target domain adaptation of background class feature representations is realized by minimizing the loss function defined in (6) during the training process.

$$L_{\text{back}} = \sum_{i}\sum_{b}\min_{j}\left\|\frac{\sum_{h,w}\delta(L_{t,P_i}^{(h,w)} - b)F(x_i^s)_{(t,i)}{}^{(h,w)}}{\max(\epsilon,\sum_{h,w}\delta(L_{t,P_i}^{(h,w)} - b))} - x_{j,b}^s\right\|_1^1 \tag{6}$$

where $i \in 1,\ldots,|X_T|$ and $b \in L^{(h,w)_{t,P_i}} \cap B$.

3) Image-Level Alignment: The image-level alignment is to transfer the image style from the source domain to the target domain. To alleviate the effect of failing alignment on image-to-image translation, this article adopts bidirectional learning to retain local semantic information when carrying out the unpaired image style translation. The observation from [18] clarifies that "source data and image translated source data" or "target data and image translated target data" have the same labels when we obtain an ideal segmentation adaptation model. The perceptual loss is introduced to measure the difference between source data and image-translated source data or target data and image-translated target data, which is used to guide the training process for obtaining an ideal segmentation adaptation model. The perceptual loss (l_{ppl}) is given in (7).

$$\begin{aligned} L_{ppl} = \lambda_{ppl} E_{X_S} ||I(X_S) - I(G(X_S))||_1 \\ + \lambda_{ppl_r econ} E_{X_S} ||I(F(X_S)) - I(X_S)||_1 \end{aligned} \quad (7)$$

where I is the segmentation network. G is the image-to-image translation network from $X_S \rightarrow X_T$. F is the translation network from $X_T \rightarrow X_S$. L_{ppl} and $\lambda_{ppl_r econ}$ are the weighted factors for constructing and reconstructing paths. Due to the symmetry, the L_{ppl} of X_T and $F(X_T)$ is similar as shown above.

D. Self-Guidance With Our Proposed Method

In the semantic segmentation adaptation, the labels of target data are not available. The segmentation loss is computed by using the ground truth annotations from source domains. Such a manner neglects the discrepancy of the distribution for ground truth labels in the source and target domains. Taking this into account, our proposed method is combined with a self-supervised learning framework to alleviate misalignment of ground truth labels from source and target domains.

There are two stages for self-guidance training. First, a model is trained on the source domain images X_S and their corresponding ground-truth annotations Y_S along with the target domain images X_T. Second, the model obtained from the first step is applied to produce pseudo-labels. More specifically, the pseudo-labels of the training set images X_T are generated by using the pixels with high predicted confidence scores as shown in (8).

$$\sigma_k(G(x_i^t)) > y_t^k \Rightarrow \hat{y}_i^t = \arg\max_{k \in N} G(x_i^t)^{(k)} \quad (8)$$

where $\sigma_k(\bullet)$ returns the confidence score of class k, which is generated by generative network $G(x_i^t)$. The confidence threshold of class k is denoted as y_t^k. Then, our model is retrained by using the semantic segmentation loss of the target domain images, which is given in (9).

$$L_{\text{seg}}^T(F(x^t)) = -\sum_{i,h,w} \sum_{k \in K} \hat{y}_i^{(h,w)} \log(\sigma_k(G(x_i^t)^{(h,w)})). \quad (9)$$

With the help of pseudo-labels, the generated features of corresponding classes are pushed closer to the corresponding intraclass features of the source domain. As a result, adapted segmentation performance can be further enhanced for hierarchical feature alignment.

E. Full Objective

Following [18], [27], a two-stage training pipeline is enforced to make trained model generalize better in the target domain dataset on semantic segmentation. The former step is to train our model without the pseudo-labels. The target function is optimized through an adversarial training strategy given in (10).

$$\begin{aligned} \min_{G,D} L_{\text{former}} = \min_{G}(\lambda_{\text{seg}} L_{\text{former}}^S + \lambda_{\text{adv}} L_{\text{adv}} \\ + L_{\text{hierarchy}} + \min_{D} \lambda_D L_D) \end{aligned} \quad (10)$$

where λ_{seg}, λ_{adv}, and λ_D are the weights of segmentation loss, adversarial loss, and discriminator loss. After obtaining the pseudo-labels of the target domain from the former step, latter step is to repeat the training process with reinitializing the weights of the network and using pseudo-labels to guide the optimization of minimizing the loss function in (11)

$$\begin{aligned} \min_{G,D} L_{\text{latter}} = \min_{G}(\lambda_{\text{seg}}^S (L_{\text{seg}}^S + L_{\text{seg}}^T) + \lambda_{\text{adv}} L_{\text{adv}} \\ + \tilde{L}_{\text{hierarchy}} + \min_{D} \lambda_D L_D) \end{aligned} \quad (11)$$

where $\tilde{L}_{\text{hierarchy}}$ is augmented with predicted $\hat{y}_i^t$ according to (8).

IV. Experimental Evaluation

A. Datasets

To evaluate our proposed method, three benchmark datasets, i.e., GTA5, SYNTHIA, and Cityscapes [9], are used in the experiments. In specific, GTA5 and SYNTHIA are chosen as the source domain data since they are both synthetic datasets and easy to be collected and labeled. The Cityscapes dataset is chosen as the target domain since it is a realistic dataset and difficult to be labeled due to the large data scale.

1) GTA5 to Cityscapes: The GTA5 dataset consists of 24,966 fine annotated synthetic images with the resolutions of 1914 × 1052. All these images are captured from a photo-realistic open-world computer game called "Grand Theft Auto V." Similar to [7], [9], [18], [27], GTA5 images are resized into the resolution of 1280 × 720 for saving GPU memory. In the GTA5 dataset, there are 19 classes shared with the Cityscapes dataset. Therefore, all the 19 classes can be used to evaluate the performance of semantic segmentation. The images in the Cityscapes dataset are resized to the resolutions of 1024 × 512 for training and validating purposes.

2) SYNTHIA to Cityscapes: The SYNTHIA [9] dataset consists of 9400 images with the resolutions of 1280 × 760, and also the dense pixel-level annotations. Following [7], [9], [18], [27], we evaluate our models on Cityscapes validation set with the 13 common classes between SYNTHIA and Cityscapes. Similar to the adaption from GTA5 to Cityscapes, the models are trained and tested on Cityscapes images with the resolution of 1024 × 512.

B. Implementation Details

To train the segmentation and discriminator networks, Pytorch is used on a single RTX2080ti GPU. As emphasized in [9], a strong baseline model is helpful to obtain better understanding on the effect of different adaptation approaches, and enhance the performance for the practical applications. Thus, according to the conventional literature, the backbone of pretrained ResNet-101 on ImageNet is chosen as our baseline model [7], [9], [18], [27]. Specifically, the backbone network is with five

TABLE I
CONFIGURATION OF NETWORKS. ILR, INITIAL LEARNING RATE; DP, DECAY POWER, AND WD, WEIGHT DECAY

Network	Optimiser	ILR	DP	β_1	β_2	λ_{seg}	λ_{adv}	Momentum	WD	λ_{class}	λ_D	λ_{obj}	w
Segmentor	SGD	1×10^{-4}	0.9	-	-	1	-	0.9	5×10^4	0.003	-	0.01	50
Discriminator	Adam			0.9	0.99	-	0.01	-	-	-	1	-	-

TABLE II
QUANTITATIVE COMPARISON RESULTS FROM GTA5 TO CITYSCAPES (UNIT %)

Semantic Class	FCN (baseline) [7]	AdaptSegNet [9]	CLAN [29]	SIM [27]	BDL [18]	SIM with SSL [27]	Ours (PHEA)
road	75.8	82.0	85.8	86.3	89.8	87.9	89.7
sidewalk	16.8	29.7	16.1	26.1	41.0	29.5	41.6
building	77.2	78.8	79.6	78.0	82.8	79.1	83.4
wall	12.5	21.9	25.3	25.6	28.2	29.2	29.8
fence	21.0	18.0	22.4	19.8	20.9	17.9	26.1
pole	25.5	26.4	28.7	21.5	29.7	23.8	30.7
light	30.1	28.5	32.9	23.1	31.4	31.6	34.2
sign	20.1	26.6	20.5	13.2	28.5	17.5	32.3
vegetation	81.3	80.0	82.5	81.5	83.3	82.7	83.5
terrain	24.6	26.7	32.3	31.6	36.2	34.1	39.0
sky	70.3	72.8	72.8	77.7	82.5	75.2	81.6
person	53.8	55.4	57.8	48.2	58.6	54.3	58.9
rider	26.4	25.1	28.1	22.6	27.6	27.9	28.2
car	49.9	72.5	80.0	75.1	83.2	79.0	84.1
truck	17.2	30.2	33.5	29.3	36.7	28.6	33.5
bus	25.9	13.2	28.4	28.5	42.0	37.1	42.7
train	6.5	11.4	0.5	2.0	1.6	1.4	2.7
motorcycle	25.3	29.3	29.0	29.3	25.0	22.3	28.1
bicycle	36.0	10.5	25.3	23.2	38.0	29.1	38.2
mIoU	36.6	38.9	41.1	39.1	45.6	41.5	**46.8**

convolutional layers. The final layer is used to obtain a high-quality feature map, and atrous spatial pyramid pooling (ASPP) is applied for classification modules with controlling the weight by hyperparameter λ_{adv}. In agreement with [7], [9], [18], [27], discriminator network contains five convolutional layers with channel number $\{64, 128, 256, 512, 1\}$. The kernel size and stride are set to 4×4 and 2, respectively. To train the discriminator network, the output of ASSP head on the final conventional layer is upsampled with weights λ_{adv} and λ_D. The detailed configuration of training process is given in Table I. But unlike the conventional literature, we only use 1% of the data from Cityscapes training dataset, which are 30 images, instead of using all unlabeled images of the training set as in [7], [9], [18], [27], which are 2,975 images. Thus, in our evaluation scenario, the amount of data in source and target domain is extremely imbalanced, which is more realistic and closer to the practical real-world applications.

C. Evaluation Metrics

This section provides the metrics for evaluating the performance of adapted semantic segmentation. To quantitatively evaluate the results of semantic segmentation, interaction-over-union (IoU) is used to assess the performance for each semantic class. The definition of IoU is given by

$$IoU(Y, \hat{Y}) = \frac{Y \cap \hat{Y}}{Y \cup \hat{Y}} = \frac{t_p}{t_p + f_n + f_p} \tag{12}$$

where Y are pixel-wise labels of ground-truth and $\hat{Y}$ are the predictions of each pixel. t_p, f_n, and f_p represent the true positives, false negative, and false positives, respectively. IoU is used to measure the performance of a specific semantic class and it is not affected by the class imbalances. In addition, the overall performance of different methods is measured through the mean value of IoU for all semantic classes, which is defined by mIoU.

D. Quantitative and Qualitative Analysis

Tables II and III provide the semantic segmentation results on GTA5-to-Cityscapes and SYNTHIA-to-Cityscapes adaptation, respectively. FCN is the baseline for domain adaptation, where a segmentation model is only trained by the source dataset and then assessed on the target dataset. For the GTA5-to-Cityscapes adaptation, the mean IoU is used to exhibit the performance of 19 common classes shared between the two datasets. To keep consistency with [7], [9], [18], [27], the mean IoU for SYNTHIA-to-Cityscapes adaptation is evaluated in 13 categories, and 6 categories (i.e. fence, wall, pole, terrain, truck, and train) are not taken into account. Compared with the FCN baseline model, our proposed method outperforms on all classes in class-wise IoU and mIoU. Specifically, the mIoU score of our proposed method can be boosted to 46.8 and 51.0 on the GTA5-to-Cityscapes and SYNTHIA-to-Cityscapes adaptation, respectively. In addition, to alleviate the difficulty of maintaining the category and spatiality information for aligning the marginal distributions across domains, we not only enforce output space feature (as in AdaptSegNet [9]), but also introduce reliable target-style images into the training process to prevent the spatial information being interrupted in the segmentation network. According to comparative results provided in Tables II and III, our method can achieve a significant improvement compared with other methods. The best performance of a specific semantic class is highlighted in blue and the best performance of mIoU

TABLE III
QUANTITATIVE COMPARISON RESULTS FROM SYNTHIA TO CITYSCAPES (UNIT %)

Semantic Class	FCN (baseline) [7]	AdaptSegNet [9]	CLAN [29]	SIM [27]	BDL [18]	SIM with SSL [27]	Ours (PHEA)
road	55.6	61.2	56.3	61.9	72.6	61.7	77.2
sidewalk	23.8	26.5	22.2	23.9	34.4	24.0	36.3
building	74.6	65.5	75.7	75.6	71.7	77.7	79.6
light	6.1	17.8	19.9	10.7	21.9	12.0	26.9
sign	12.1	22.7	15.4	9.5	23.7	11.7	35.6
vegetation	74.8	75.5	77.6	74.1	77.6	77.7	81.3
sky	79.0	77.5	78.0	76.6	75.6	78.5	82.2
person	55.3	45.0	51.5	48.8	47.1	52.2	48.3
rider	19.1	21.0	22.9	18.0	22.6	22.4	23.4
car	39.6	72.1	60.1	66.4	74.0	68.1	74.1
bus	23.3	24.7	30.2	22.4	27.1	20.6	30.0
motorcycle	13.7	15.3	13.0	15.6	14.1	19.6	20.9
bicycle	25.0	37.8	31.6	30.1	47.7	41.3	47.7
mIoU	38.6	43.3	42.6	41.1	46.9	43.6	**51.0**

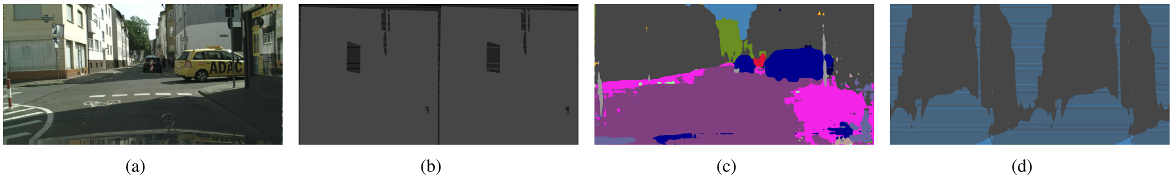

(a)
(b)
(c)
(d)

Fig. 2. Visualization of segmentation results. (a) Target domain image. (b) Segmentation ground-truth of the corresponding target image. (c) Generated images from AdaptSegNet. (d) Generated images from our model.

TABLE IV
ABLATION STUDY ON THE VARIOUS COMPONENTS OF OUR METHOD FOR DOMAIN ADAPTATION (UNIT %)

FCN(baseline)	AA	IT	SG	HFA	mIoU
✓					36.6
✓	✓				38.9
✓	✓	✓			44.0
✓	✓	✓	✓		45.6
✓	✓	✓	✓	✓	46.8

is highlighted in bold black. The qualitative results of adapted segmentation are illustrated in Fig. 2, where we can visually inspect the effectiveness of our method on imbalanced adapted segmentation.

E. Ablation Study

Table IV identifies the contributions of different components to the overall performance in our proposed model, where AA is adversarial adaptation, IT is image transferring, SG is self-guidance, and HFA is hierarchical feature alignment. The mIoU can achieve 36.6 when purely trained on the source domain dataset. Then, the adversarial training of output space is proceeded as [9]. The mIoU can be improved to 38.88. As argued in [18], image-level adaption also provides a significant contribution to minimizing the discrepancy of the data distribution. In light of this, we adopt a bidirectional translation to change the image style from GTA5 to Cityscapes images by utilizing a CycleGAN structure. This further improves the mIoU to 44.0.

Next, we add our proposed hierarchical alignment scheme to the training framework. Alignment is enforced hierarchically in the object, category, and image levels. In the object level, foreground classes, e.g., car, truck, person, are aligned across domains with $\lambda_{\text{obj}} = 0.01$ and $w = 50$ semantic source domain feature samples. In the category level, we achieve the alignment of semantic classes from the source domain to the target domain by a maximum square loss, which is defined in (6).

Finally, a self-guidance framework is introduced into our hierarchical feature alignment scheme to further improve the segmentation performance, where our model is retrained by using the given pseudo-labeled target dataset. The pseudo-labels of the target dataset are obtained by choosing the confidence threshold for each class, respectively. The pixel-level pseudo-labels are identified for each target image. A confidence score map is produced based on the pseudo-labels for the corresponding image in the target domain. Subsequently, each pixel-level label is mapped with a confidence score. We rank the confidence scores of the same class for the entire target dataset and set the threshold of confidence score to 0.9 for a specific class when the median of confidence scores for the class is higher than 0.9. Otherwise, the threshold of confidence score for the class is set to the median of confidence scores. After this, the new y_t^k is set and we can follow (8) to generate the pseudo-labels with neglecting the target dataset. To this end, the model is retrained through optimizing (11). The combination of our hierarchical feature alignment scheme and self-guided framework can improve mIoU to 46.8.

V. CONCLUSION

In this article, we proposed a progressive hierarchical feature alignment method on imbalanced domain adaptation for semantic segmentation. The imbalance was from two aspects: 1) class imbalance from source to target domain and 2) data imbalance between source data and target data. To alleviate the negative effects of the imbalanced data issue, we make fully use of multi-source domain data to learn domain-invariant features. To remit the class imbalance problem, we aligned the features across domains hierarchically from bottom to top, named "hierarchical feature alignment scheme," in order to maintain the category and spatial information when aligning the marginal distributions of two domains. Our proposed method was evaluated by transfer learning tasks on synthetic datasets, GTA5 and SYNTHIA, and a realistic dataset, Cityscapes. According to the experimental results, our proposed method achieved competitive performance on imbalanced semantic segmentation adaptation. We also conducted an ablation study to investigate the contribution of various components in our method. In the future, this work can be further extended from two aspects. On the one hand, more advanced pseudo-labeling algorithms, such as DCBT-Net [30], can be integrated into the proposed method to achieve better self-guidance. On the other hand, the region-level feature [19] and the diverse characteristics of target domain [20] can be combined into our framework for providing a better holistic alignment.

References

[1] C.-C. Wong, Y. Gan, and C.-M. Vong, "Efficient outdoor video semantic segmentation using feedback-based fully convolution neural network," *IEEE Trans. Ind. Informat.*, vol. 16, no. 8, pp. 5128–5136, Aug. 2020.

[2] X. Wang, H. He, and L. Li, "A hierarchical deep domain adaptation approach for fault diagnosis of power plant thermal system," *IEEE Trans. Ind. Informat.*, vol. 15, no. 9, pp. 5139–5148, Sep. 2019.

[3] Z. Wu *et al.*, "Dcan: Dual channel-wise alignment networks for unsupervised scene adaptation," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 518–534.

[4] Z. Wu, X. Wang, J. E. Gonzalez, T. Goldstein, and L. S. Davis, "ACE: Adapting to changing environments for semantic segmentation," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 2121–2130.

[5] J. Hoffman *et al.*, "Cycada: Cycle-consistent adversarial domain adaptation," in *Proc. Int. Conf. Mach. Learn.*, 2018, pp. 1989–1998.

[6] X. Yue, Y. Zhang, S. Zhao, A. Sangiovanni-Vincentelli, K. Keutzer, and B. Gong, "Domain randomization and pyramid consistency: Simulation-to-real generalization without accessing target domain data," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 2100–2110.

[7] Y. Luo, P. Liu, T. Guan, J. Yu, and Y. Yang, "Significance-aware information bottleneck for domain adaptive semantic segmentation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 6778–6787.

[8] L. Zhang, X. Li, A. Arnab, K. Yang, Y. Tong, and P. H. Torr, "Dual graph convolutional network for semantic segmentation," in *Proc. Brit. Mach. Vis. Conf.*, 2019, pp. 1–18.

[9] 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. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 7472–7481.

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

[11] D.-H. Lee *et al.*, "Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks," in *Proc. Workshop Challenges Representation Learn.*, 2013, pp. 896–901.

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

[13] S. Motiian, Q. Jones, S. M. Iranmanesh, and G. Doretto, "Few-shot adversarial domain adaptation," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 6670–6680.

[14] Y. Zhang, Z. Qiu, T. Yao, D. Liu, and T. Mei, "Fully convolutional adaptation networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 6810–6818.

[15] L. Du *et al.*, "SSF-DAN: Separated semantic feature based domain adaptation network for semantic segmentation," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 982–991.

[16] E. Tzeng, J. Hoffman, K. Saenko, and T. Darrell, "Adversarial discriminative domain adaptation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 7167–7176.

[17] B. Sun and K. Saenko, "Deep coral: Correlation alignment for deep domain adaptation," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 443–450.

[18] Y. Li, L. Yuan, and N. Vasconcelos, "Bidirectional learning for domain adaptation of semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 6936–6945.

[19] J. Huang, D. Guan, S. Lu, and A. Xiao, "Mlan: Multi-level adversarial network for domain adaptive semantic segmentation," 2021, *arXiv:2103.12991*.

[20] A. Tao, K. Sapra, and B. Catanzaro, "Hierarchical multi-scale attention for semantic segmentation," 2020, *arXiv:2005.10821*.

[21] M. Cordts *et al.*, "The cityscapes dataset for semantic urban scene understanding," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 3213–3223.

[22] Y. Cui, M. Jia, T.-Y. Lin, Y. Song, and S. Belongie, "Class-balanced loss based on effective number of samples," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 9268–9277.

[23] D. Dwibedi, I. Misra, and M. Hebert, "Cut, paste and learn: Surprisingly easy synthesis for instance detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 1301–1310.

[24] Q. Sun, R. Chattopadhyay, S. Panchanathan, and J. Ye, "A two-stage weighting framework for multi-source domain adaptation," in *Proc. Adv. Neural Inf. Process. Syst.*, 2011, pp. 505–513.

[25] H. Zhao, S. Zhang, G. Wu, J. M. Moura, J. P. Costeira, and G. J. Gordon, "Adversarial multiple source domain adaptation," in *Proc. Adv. Neural Inf. Process. Syst.*, 2018, pp. 8559–8570.

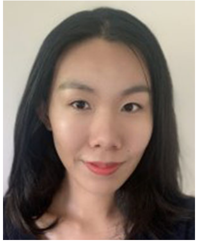
[26] Y. Li, M. Murias, S. Major, G. Dawson, and D. E. Carlson, "Extracting relationships by multi-domain matching," in *Proc. Adv. Neural Inf. Process. Syst.*, 2018, pp. 6799–6810.

[27] Z. Wang *et al.*, "Differential treatment for stuff and things: A simple unsupervised domain adaptation method for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 12635–12644.

[28] M. Chen, H. Xue, and D. Cai, "Domain adaptation for semantic segmentation with maximum squares loss," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 2090–2099.

[29] 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. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 2507–2516.

[30] B. Olimov, J. Kim, and A. Paul, "DCBT-Net: Training deep convolutional neural networks with extremely noisy labels," *IEEE Access*, vol. 8, pp. 220482–220495, 2020.



Yining Hua received the B.Eng. degree in information security from Northeastern University, Shenyang, China, in 2016, and the Ph.D. degree in computer science from the Department of Computer Science, Loughborough University, Loughborough, U.K., in 2020.

She was a Postdoctoral Research Assistant with the School of Computer Science, University of Glasgow, Glasgow, U.K., in 2021. She is currently a Lecturer in computer science with the School of Arts, University of Roehampton, London, U.K. Her research interests include autonomous systems, Internet-of-Things, edge/fog computing, and next-generation networks.

Dewei Yi (Member, IEEE) received the B.Eng. degree in software engineering from the Zhejiang University of Technology, Zhejiang, China, in 2014, and the M.Sc. degree in computer science and the Ph.D. degree in cognitive vehicles from the Department of Computer Science and Department of Aeronautical and Automotive Engineering, Loughborough University, Loughborough, U.K., in 2015 and 2018, respectively.

He was a Research Fellow with the Warwick Manufacturing Group (WMG), University of Warwick, Warwick, U.K., in 2019. He is currently a Lecturer with the School of Natural and Computing Sciences, University of Aberdeen, Aberdeen, U.K. His research interests include trustworthy autonomous systems, personalized driving assistance, autonomous vehicles, and vehicular networks.