Entropy Guided Adversarial Domain Adaptation for Aerial Image Semantic Segmentation

Aihua Zheng, Ming Wang, Chenglong Li, Jin Tang, and Bin Luo, Senior Member, IEEE

Abstract—Recent advances on aerial image semantic segmentation mainly employ the domain adaption to transfer knowledge from the source domain to the target domain. Despite the remarkable achievement, most methods focus on the global marginal distribution alignment to reduce the domain shift between source and target domains, leading to a wrong mapping of the well-aligned features. In this article, we propose an effective unsupervised domain adaptation approach, which relies on a novel entropy guided adversarial learning algorithm, for aerial image semantic segmentation. In specific, we perform local feature alignment between domains by learning a self-adaptive weight from the target prediction probability map to measure the interdomain discrepancy. To exploit the meaningful structure information among semantic regions, we propose to utilize the graph convolutions for long-range semantic reasoning. Comprehensive experimental results on the benchmark dataset of aerial image semantic segmentation and natural scenes demonstrate the superior performance of the proposed method compared to the state-of-the-art methods.

Index Terms—Aerial image, graph convolutional network (GCN), information entropy, semantic segmentation, unsupervised domain adaptation (UDA).

I. INTRODUCTION

Semantic segmentation [1] is a fundamental computer vision task that assigns a predictive category for each pixel of an input image, which has wide practical applications such as auto-driving [2] and image editing [3]. With potential applications such as land-cover monitoring and planning [4], climate monitoring and forecasting [5], urban management [6], building extraction [7], recent efforts devoted to aerial image semantic segmentation for better understanding the remote scenarios with reliable inferential knowledge. However, the aerial images collected by airborne satellites or unmanned aerial vehicles normally contain a wealth of detailed land information with redundant information and noise, which bring great challenges for semantic segmentation.

Manuscript received August 8, 2021; accepted September 13, 2021. Date of publication September 29, 2021; date of current version January 26, 2022. This work was supported in part by the National Natural Science Foundation of China under Grant 61976002, Grant 61976003, Grant 61860206004, and Grant U20B2068. (Corresponding author: Chenglong Li.)

Aihua Zheng and Chenglong Li are with the Information Materials and Intelligent Sensing Laboratory of Anhui Province, Anhui Provincial Key Laboratory of Multimodal Cognitive Computation, School of Artificial Intelligence, Anhui University, Hefei 230601, China (e-mail: ahzheng214@foxmail.com; lcl1314@foxmail.com).

Ming Wang, Jin Tang, and Bin Luo are with Anhui Provincial Key Laboratory of Multimodal Cognitive Computation, School of Computer Science and Technology, Anhui University, Hefei 230601, China (e-mail: wangming@stu.ahu.edu.cn; tangjin@ahu.edu.cn; ahzlb@163.com).

Digital Object Identifier 10.1109/TGRS.2021.3113581

Recent achievements on natural image semantic segmentation, such as FCN [8], U-Net [9], SegNet [10], PSPNet [11], ResNet [12], and DeepLab [13] significantly boost the development of aerial image semantic segmentation [14]–[16] due to the powerful capability of feature learning and representation. Despite their excellent success in aerial image semantic segmentation, the key issue is these data-driven supervised methods rely on a large amount of annotations, which are extremely time-consuming and labor-intensive, especially for the extraordinary high-resolution aerial image. Although weakly supervised methods [17]–[21] and synthetic data [22], [23] can overcome this issue to some extent, they still require extra human efforts or have a large discrepancy in texture, spatial layout, color, and lighting conditions between synthetic and real images which is also called domain shift. Domain adaptation, as a particular case of transfer learning, has been widely explored to simulate the human visual system to learn from one or more related source domain data distribution and transfer the learned knowledge to the target domain [24], [25]. Unsupervised domain adaptation (UDA) bridges the gap between the labeled source and the unlabeled target domains mainly through maximum mean discrepancy (MMD) [26], [27], adversarial learning [28]–[34], or self-training [35]–[37]. Recent efforts evidence the promising achievement of adversarial learning-based DA in UDA semantic segmentation.

Adversarial DA uses the discriminator to improve the ability of the generator while learning domain-invariant features. Early efforts [30], [38]–[40] mainly focus on generative adversarial learning methods by employing the generative adversarial network (GAN) [41] structure, which generates simulated images to mimic the appearance of the target domain and retain the semantic invariant. Recent efforts [28], [29], [33], [34] adopt the subspace adversarial learning method to learn the domain-invariant features, which maps images from both domains to a common subspace and then uses a discriminator to reduce distribution discrepancy due to the similarity of the feature distribution in the feature space [29] or the spatial layout in the output space [33], and [32] uses the differences between the two classifiers to promote feature alignment. Others methods [31], [42]–[44] make efforts to integrate both generative and subspace adversarial learning methods. More recent works devote to introduce curriculum learning [45] and entropy minimization [46] to adversarial learning to reduce the domain shift. However, most existing adversarial learning-based UDA methods suffer from a major limitation: they focus on aligning the global marginal distribution of the features from both domains while ignoring...
the local joint distribution shift that the discrepancy of regional feature similarity for multiple categories between domains, as so-called negative transfer [34].

Recent efforts [47]–[50] evidence the effectiveness of domain adaptation method on aerial image semantic segmentation. Benjdira et al. [47], [48] and Tasar et al. [49] translate images between domains to reduce style discrepancy and reuse labels as additional supervision based on the generative method. Fang et al. [50] apply feature generation and category-level subspace adversarial learning methods [34], via a geometry-consistent GAN embedded into a cotraining adversarial network to eliminate above negative transfer. It is difficult to train the discriminator to distinguish the transferred images or the generated high-dimensional features. At the same time, the above methods rarely take into account the domain-invariant feature alignment during the global adversarial learning, and the additional classifier increases the complexity of the model in the work of Fang et al. [50]. Furthermore, most of the above methods directly derive from natural images and ignore the specific characteristics of the aerial image, such as the existence of multiscale objects and the absence of regional semantic information in image preprocessing.

To solve these problems, we propose an entropy guided adversarial (EGA) learning method in the output-space to enhance the local feature alignment for UDA aerial image semantic segmentation and encode the long-range semantic dependencies between regions for compensating the semantic absence base on graph convolutional network (GCN). Information entropy, which indicates the amount of information and measures uncertainty, has been widely explored in semantic segmentation [37], [46], [51], [52]. As shown in Fig. 1(a), the model trained on the labeled source domain tends to produce over-confident prediction with low entropy. And the target domain predictions based on a source-only trained model usually represent low-entropy on the source-like (domain-invariant) regions (the category regions of “Build.”) and high-entropy on the target-like (domain-variant) regions (the regions of “Imp. surf.” and “Tree”) due to the domain shift, as shown in Fig. 1(b) and (c). Furthermore, the information entropy map has a strong relationship with the error map, which is apparent in the labeled source domain. This relationship has also been studied in previous work [51], [52]. In particular, we have studied the entropy distributions of correct and incorrect predictions in the source and target domains in Fig. 1(c). The incorrect prediction usually corresponds to the high entropy, while the correct prediction corresponds to the low entropy. The traditional global feature alignment method ignores the local feature alignment, thus will cause prediction errors in domain-invariant regions, as shown in the third row of Fig. 1(d) (the category of “Build.”). Therefore, regions should be discriminatively treated in adversarial learning. We argue that the entropy map can represent the domain shift, which in turn means we can use the entropy value to enhance the local feature alignment in adversarial learning. By the EGA learning, we encourage the network to preserve the domain-invariant features while enforcing the alignment of the domain-variant features.

Fig. 1. Effectiveness of the proposed EGA domain adaptation for aerial image semantic segmentation. The first two rows show the results of source (a) and target (b) domain scene tested on the source-only trained model. In the third row (c), the left figure draws the relationship between the entropy and the predictions on both domains before adaptation, while the right one plots the relationship between the feature similarity between two domains for each class together with the entropy of the target domain. The last two rows show the results of the same target domain scene after traditional global feature alignment (d) and our method (e). Columns represent the images from both domains, the information entropy maps calculated on the prediction probability maps and the error maps, respectively. The error map is represented by a binary map to show the difference between the ground truth (GT) and the prediction map, where white and black pixels indicate the correct and incorrect predictions, respectively.

In addition, we note that the scene layout between the source and the target domain in aerial image presents a huge discrepancy comparing to the natural image, which brings more challenges to the segmentation task, as shown in Fig. 2(a) and (b). Due to the high resolution, aerial images are usually divided into multiblocks, which will result in missing region semantics, such as the “Build.” cutting from the middle significantly destroy the local region structure. Therefore, the region’s relationships are needed for long-range semantic dependence. As shown in Fig. 2(c), the feature response map after the long-range semantic reasoning (LSR) is enhanced by encoding more context information and refining object boundaries.

The main contributions of our work can be summarized as follows.

1) We propose a novel UDA network to handle the problem of local distribution shift for aerial image semantic segmentation.
that initializes the weights by using other aerial image to alleviate the scarcity of training dataset. Although the above methods achieve significant improvement, they mainly train segmentation networks through many annotated images, which are usually difficult to be collected in real scenarios. Meanwhile, they are based on the assumption that the training and testing sets fall into the same data distribution, which is also hard to be satisfied when the domain shift emerges.

Recent efforts [47]–[50] evidence the effectiveness of domain adaptation method for aerial image semantic segmentation. Benjdira et al. [47] proposed to generate images from the source domain to the target domain for reducing style discrepancy and reuse labels from the source domain. Tasar et al. [49] proposed a translating framework, ColorMapGAN framework to mimic the spectral distribution of the target image. Meanwhile, Benjdira et al. [48] proposed to use two GAN structures and a small set of labeled target images to improve the generalization ability of the model on the target domain. Fang et al. [50] proposed the category-sensitive domain adaptation via a geometry-consistent GAN embedded into a cotraining adversarial network. However, these generation-based methods are generally hard to train, and their segmentation performance is significantly restricted by the quality of the generated images or features.

B. UDA for Semantic Segmentation

Adversarial learning [41] is one of the most active techniques for UDA semantic segmentation due to its powerful ability in image or feature generation and the adversarial loss can be used to promote feature distribution alignment between domains. Adversarial learning can be considered into two parts, including generative adversarial learning and subspace adversarial learning methods.

The works [29], [32]–[34], [59] apply the subspace adversarial learning method to learn domain-invariant features. The map images from both domains to a common subspace (feature- or output-space) and then use discriminators to reduce domain discrepancies. Hoffman et al. [29] are the first to explore the adversarial learning based on feature subspace for UDA semantic segmentation, propose both global, and category-specific domain adaptation techniques. Tsai et al. [33] propose a multilevel adversarial network on different feature representation levels to realize output-space-based adversarial learning. However, such methods pursue global feature alignment and ignore the regional discrepancy between domains. Luo et al. [34] propose a category-level adversarial network using the cotraining mechanism and calculating the two distinct view-prediction local-aligned maps to guide the adversarial learning for keeping local semantic consistency. Zheng and Yang [59] propose an orthogonal method to exploit the intradomain knowledge and reduce the inherent uncertainty by minoring the discrepancy of two different classifiers. Saito et al. [32] minimize the difference from two classifiers to align feature distribution between domains. Some works [30], [31], [38]–[40], [42], [43] employ the generative adversarial learning to translate the image styles for decreasing the appearance difference in the

Fig. 2. Comparison of natural image and aerial image, and the feature response map with/without the LSR. The scene layout of the natural image in the source domain is more similar as in the target domain comparing with the aerial image by comparing (a) with (b). Furthermore, the divided blocks in the aerial image have lost/destroyed the local semantic information, as shown in (b). In (c), we show the deep feature response map with/without LSR.

II. RELATED WORK

We briefly review methods of aerial image semantic segmentation and related works on UDA for semantic segmentation.

A. Aerial Image Semantic Segmentation

The powerful ability of deep models and the availability of large-scale aerial images with high visual quality provides an opportunity to enhance segmentation performance. Hence, many representative deep learning-based methods have been applied to aerial image semantic segmentation. Existing efforts mainly concern to modify the FCN framework for segmentation work [14]–[16], [51]–[58]. For example, Sherrah [53] incorporate the dilated convolution into FCN without extra down-sampling operation. To fuse high-level semantic information with low-level detail information, Wang et al. [51] propose a gate mechanism implemented by entropy maps to fuse feature maps from different levels. Furthermore, Panboonyuen et al. [57] employ a global convolutional network to capture complex features and produce weights for the feature in different layers by the channel attention. They also introduce domain-specific transfer learning

2) An EGA domain adaption approach is proposed, which learns the adaptive weight of the predictive entropy map in the target domain to guide the adversarial learning for local feature alignment.

3) A LSR model is designed to explore local regional dependencies and compensate for the lack of semantic structure information.

4) Experiments on benchmark aerial segmentation dataset ISPRS and natural synthetic-to-real dataset demonstrate the effectiveness and robustness of our proposed method, which yields a new state-of-the-art performance.
image-level and make feature invariant to style information. Bousmalis et al. [30], Li et al. [39], and Wu et al. [40] apply style transfer approaches to reduce appearance discrepancy.

Hoffman et al. [31] propose the cycle-consistent adversarial domain adaptation, based on the CycleGAN [38], to generate images and reuse corresponding labels by the constraint of the cycle-consistency and semantic losses. Li et al. [43] jointly consider subspace and generative adversarial learning methods, apply bidirectional learning for domain adaptation, and introduce a self-supervised learning method to alternatively learn the image generator and the segmentation model by perceptual loss. Vu et al. [46] propose two complementary losses, i.e., entropy loss and adversarial loss, based on information entropy to reduce the data distribution gap between domains. Zhang et al. [45] design curriculum-style learning by considering the consistency of the local and global label distributions for UDA semantic segmentation. Other works introduce the idea of self-training. Zou et al. [35], [36] and Saporta et al. [37] propose to generate high-confidence pseudo-labels on the target domain, and then use the pseudo-labels to train the target model.

III. METHODS

In this section, we shall elaborate on the proposed LSR and EGA domain adaptation for aerial image semantic segmentation.

A. Problem Settings

Given the source aerial image set $X_S$ with the paired pixel-level labels set $Y_S$, and the target unlabeled aerial image set $X_T$, the ultimate goal of unsupervised domain adaptation in aerial image semantic segmentation is to learn a model $G$ producing the pixel-level labels for the target aerial image. The traditional adversarial network employs a generative model $G$ to transfer knowledge from the source domain to the target domain by minimizing a supervised segmentation loss in the source domain. The supervised segmentation loss defined as

$$L_{seg}(G) = \mathbb{E}_{(x_i, y_i) \sim (X_S, Y_S)}[\ell(G(x_i), y_i)]$$

where $x_i \in X_S$, $y_i \in Y_S$, $\mathbb{E}[]$ represents the statistical exception and $\ell$ indicates the commonly employed cross-entropy loss.

Meanwhile, to generalize the model $G$ to the target domain, adversarial-based methods encourage $G$ to learn the domain-invariant features by confusing the domain discriminator $D$ which tries to distinguish the outputs from the source or target domain. This process is achieved by minimizing an adversarial loss

$$L_{adv}(G, D) = -\mathbb{E}_{x_i \sim X_S}[\log(D(G(x_i)))] - \mathbb{E}_{x_i \sim X_T}[\log(1 - D(G(x_i)))]$$

where $x_i \in X_T$, represents an unlabeled sample from target set.

As mentioned above, traditional adversarial learning methods [29], [31], [33], [42], [46] focus on the global feature alignment, which will cause the negative transfer. The direct effect is, some already aligned features may be destroyed by the global margin distribution alignment. Meanwhile, the aerial image has specific characteristics, such as the multiscale variations and the loss of structural continuity during data preprocessing. To relieve these problems, we proposed an EGA learning method to compute a self-adaptive weight that suppresses the interdomain shift during the adversarial learning by information entropy, and the LSR by GCN to explore the structure information.

B. Overview of the Proposed Model

As shown in Fig. 3, there are two main components in the proposed model, i.e., a generator $G$ and a discriminator $D$. The generator $G$ aims to generate feature representation and prediction map, which consists of a segmentation backbone (DeepLab-V2 [13] with ResNet-101 [12] in this article) and an embedding LSR model. The discriminator $D$ is a convolutional neural network (CNN)-based binary classifier with a fully convolutional output. In our framework, $G$ consists of three parts: 1) feature extractor $F$, that extracts CNN features $F$ from both domains, 2) embedded LSR model, that explores the image structure information and enhances the local feature representation on $F$, and 3) classifier $M$, that produces the prediction probability map. The discriminator $D$ works on the output-space to align the distributions of two domain predictions under the guidance of the information entropy map of the target domain.

For the source domain aerial image $x_s$, we first generate the deep feature map $F_s$ by feature extractor $F$, followed by the LSR model to aggregate the node information through the information passing mechanism of the graph structure. Then, the aggregated node features are mapped to the original deep feature space and merged with deep features by summation operation, to enhance the information of the features without increasing the model parameters comparing to the concatenation. After that, we obtain the prediction map $P_s$ via the classifier $M$, followed by the supervised segmentation loss with the corresponding label $y_s$. In the same manner, we can obtain the prediction map $P_t$ for the target domain. In addition, based on the probability prediction map, we further calculate the entropy map as a self-adaptive weight $W_g$, which can guide the adversarial learning to perform feature alignment. Meanwhile, to pull the distribution of target prediction to the source domain, the discriminator $D$ is introduced to distinguish the prediction map $P_t$ from the source or target domain. Then we calculate the adversarial loss weighted by $W_g$ to learn the diverse attention for each pixel.

C. LSR Model

Our proposed LSR model aims to explore the regional dependence relations in the aerial image by the information passing mechanism of the graph structure.

The GCN [60] operates on the instance graph to explore the data structure information. In this article, we propose to aggregate the meaningful structure information in aerial image segmentation via GCN due to its capability of extracting
We directly construct the graph structure and aggregating node features by nodes relations from the graph structure. We directly perform graph reasoning on the original feature map to explore the long-range dependence and structure information. We formulate the graph model as

\[ X^{(l+1)} = \sigma \left( \tilde{D}^{-\frac{1}{2}} \tilde{A} \tilde{D}^{-\frac{1}{2}} X^{(l)} \Theta^{(l)} \right) \]  

where \( \tilde{A} = A + I \), \( I \) is an identity matrix, and \( \tilde{D} \) is a degree matrix, represents the degree of each node, is used to normalize the new features of each node aggregated from the connected nodes. \( \tilde{D}_{ij} = \sum_j \tilde{A}_{ij}, \) \( i \) and \( j \), respectively, represent the \( i \)th row and the \( j \)th column of \( \tilde{A} \). The value of degree matrix \( \tilde{D} \) is used to calculate the symmetric normalized Laplacian matrix \( \tilde{D}^{-(1/2)} \tilde{A} \tilde{D}^{-(1/2)} \) to measuring the relations of nodes and compute the new features of each node as the average of itself and connected nodes. \( X^{(l)} \) is the vertex feature matrix in \( l \)th layer, \( \sigma \) is the nonlinear activation function (rectified linear unit (ReLU) used in our experiments), and \( \Theta^{(l)} \) is a trainable weight matrix.

To apply the forward propagation rule in (3) on the CNN feature map to explore the long-range dependence and structure information. We directly perform graph reasoning on the original feature map. For an input deep CNN feature map \( F \) with the shape of \( h \times w \times n \), where \( h, w, n \) represent the height, width, and channel number, respectively. To reduce parameters and computational costs, we use a \( 1 \times 1 \) convolution reducing the channel into \( m \). Therefore, the graph node input can be represented by transferring features after reduced dimension into \( X \in \mathbb{R}^{hw \times m} \). As shown in Fig. 3, we calculate the adjacency matrix \( A \) by the dot-product distance of the dimension-reduced features. In this way, it will increase the computational cost in the LSR module when aggregating features. As we evaluated in the ablation study in Table V, it significantly boosts the performance by enhancing the contextual features without introducing significant computational cost. The similarity between position \( i \) and \( j \) is expressed as

\[ A_{ij} = \phi(F)_i \phi(F)_j^T \]  

where \( F \) is deep feature map extracted from CNN, \( \phi(\cdot) \) is a \( 1 \times 1 \) convolution, and \( \phi(F) \in \mathbb{R}^{hw \times m} \) is a linear embedding followed by ReLU(\( \cdot \)). The \( A \) can be calculated as follows:

\[ A = \phi(F; W_{\phi}) \phi(F; W_{\phi})^T \]

where \( W_{\phi} \) is the learnable parameters for the linear transformations. Meanwhile, \( A = A + I \) and \( \tilde{D}_{ii} = \sum_j \tilde{A}_{ij} \). We can formulate the graph model as

\[ Z = \sigma \left( \tilde{D}^{-\frac{1}{2}} \tilde{A} \tilde{D}^{-\frac{1}{2}} X \Theta \right) \]  

where \( \Theta \) is a trainable weight matrix, \( \sigma \) is the ReLU activation function, and \( Z \) is the output features.

After the LSR model, the output \( Z \) projects back into the original deep feature space and then fused with the original deep CNN feature map \( F \). By this means, the refined feature \( \tilde{F} \) encodes long-range semantic dependencies and aggregates the image structure information.

Fig. 3. Pipeline of our proposed EGA learning and the LSR model base on GCN. It contains two components, generator (\( G \)) and discriminator (\( D \)). The generator consists of a feature extractor (\( E \)), an embedded LSR model, and a classifier (\( M \)). For the source domain input, we use the prediction after the classifier to calculate a supervised segmentation loss. As for the target input, based on the output of the classifier, we first calculate a self-adaptive weight and aggregating node features by nodes relations from the graph structure. We directly perform graph reasoning on the original CNN feature maps to preserve the spatial relationship. The feature map is one-eighth of the input image size, therefore the computational cost is not extremely large to handle the most existing cropped aerial or natural images. Given an undirected graph \( g = (v, e) \), the nodes \( v \) indicate each pixel node of the deep feature map, the edges \( e \) represent the similarity between nodes, and \( A \in \mathbb{R}^{n \times n} \) is an adjacency matrix. Unlike the standard convolution, which operates on a local regular-grid region, the graph can encode the long-range dependencies through the relation between nodes represented by the adjacency matrix. Therefore, it has a larger receptive field and can explore more context information. Formally, the graph convolution can be defined as:

\[ X^{(l+1)} = \sigma \left( \tilde{D}^{-\frac{1}{2}} \tilde{A} \tilde{D}^{-\frac{1}{2}} X^{(l)} \Theta^{(l)} \right) \]
D. EGA Model

To relieve the negative transfer during domain adaptation, we propose an EGA model, which can learn a self-adaptive weight from the information entropy map to guide the adversarial process.

In this work, we employ the widely used Shannon entropy [61]. For a target prediction $P_t$ with the shape of $C \times H \times W$, the entropy map $\text{Ent} \in [0, 1]^{H \times W}$ can be calculated as follows:

$$\text{Ent}^{(i,j)} = -\sum_{c=1}^{C} P_t^{(i,j,c)} \log P_t^{(i,j,c)}$$  \hspace{1cm} (7)

where $P_t$ is a prediction probability map in target domain flow, $C$, $H$, and $W$ correspond to the class numbers, height, and width, respectively, and $i \in \{0, 1, \ldots, H - 1\}$, $j \in \{0, 1, \ldots, W - 1\}$. $P_t^{(i,j,c)}$ represents the predicted probability in the class $c$ (not the real label) at the position $(i, j)$.

As discussed above, information entropy can be used to measure the domain shift. Due to the shift between domains, the predictions of the regions with the large discrepancy on the target domain image may be under-confident, corresponding to high-entropy. While the predictions of similar regions are generally with low-entropy. Therefore, it is meaningful to use entropy weight to measure the discrepancies between domains via weighting the adversarial loss. The EGA loss can be defined as

$$\mathcal{L}_{\text{ega}}(G, D) = -\mathbb{E}_{x_t \sim X_t} [\log(D(G(x_t)))]$$

$$-\mathbb{E}_{x_t \sim X_t} [W_g \times \log(1 - D(G(x_t)))]$$  \hspace{1cm} (8)

where the weight map is a tensor with the size of $1 \times H \times W$, and $W_g^{(i,j)} = (1/C) \times \text{Ent}^{(i,j)}$. The domain adversarial loss map produced by $D$ is upsampled to the size of the input image. An element-wise multiplication is operated between the weight map and domain adversarial loss map. After this operation, each location in the domain adversarial loss map has different attention, and the $\mathcal{L}_{\text{ega}}(G, D)$ will focus on the low similarity regions and decrease the attention on the high similarity regions on the target sample, which can push local feature alignment between domains.

E. Training Objective

As the traditional adversarial network, our method is optimized by two loss functions, segmentation loss, and feature alignment between domains.

1) Segmentation Loss: Given an aerial image $x_t$ of shape $3 \times H \times W$ and correspond label map $y_t$ of shape of $C \times H \times W$, $C$ is the number of object classes in the domain.

The segmentation loss can be defined as

$$\mathcal{L}_{\text{seg}}(G) = \sum_{i=1}^{H} \sum_{j=1}^{W} \sum_{c=1}^{C} -y_t^{(i,j,c)} \log P_s^{(i,j,c)}$$  \hspace{1cm} (9)

where $P_s^{(i,j,c)}$ represents the predicted probability belonging to class $c$ at the location pixel $(i, j)$. $y_t^{(i,j,c)}$ denotes the corresponding GT probability belonging to class $c$ at the location $(i, j)$. If pixel $(i, j)$ belongs to class $c$, $y_t^{(i,j,c)} = 1$, else $y_t^{(i,j,c)} = 0$.

2) Self-Adaptive EGA Loss: In the traditional adversarial learning process, due to the negative shift in global feature alignment, we jointly consider local and global features alignment. In the DA process, we consider generating self-adaptive attention to different regions to promote local region feature alignment. By the EGA learning, we encourage the network to preserve the domain-invariant features while enforcing the alignment of the domain-variant features. And the self-adaptive weight $W_g$ is calculated by the prediction probability of the target domain input. Given an aerial image $x_t$ with the shape of $3 \times H \times W$, $P_t$ represents the prediction after generator $G$. Thus, the self-adaptive weight map $W_g$ can be calculated as follows:

$$W_g^{(i,j)} = -\frac{1}{C} \sum_{c=1}^{C} P_t^{(i,j,c)} \log P_t^{(i,j,c)}$$  \hspace{1cm} (10)

where dividing by $C$ is to reduce the calculated information entropy value and prevent misalignment of features caused by high entropy. The local feature alignment loss formula is mentioned has mentioned in (8).

Furthermore, to balance local and global features alignment, we design $\lambda_w$ and $\epsilon$ to promote the feature alignment. By considering both global and local feature alignment, our method can mitigate the domain shift and encourage the generator to learn more domain-invariant knowledge. The self-adaptive EGA loss can be formulated as

$$\mathcal{L}_{\text{EGA}}(G, D) = -\mathbb{E}_{x_t \sim X_t} [\log(D(G(x_t)))]$$

$$-\mathbb{E}_{x_t \sim X_t} [\lambda_w W_g + \epsilon \log(1 - D(G(x_t)))]$$  \hspace{1cm} (11)

where $\lambda_w$ effects on the self-adaptive weight $W_g$, and the $\epsilon$ is a decimal to stabilize the training process.

With the above descriptions, the overall loss objective can be summarized as

$$\mathcal{L}(G, D) = \mathcal{L}_{\text{seg}}(G) + \lambda_{\text{adv}} \mathcal{L}_{\text{EGA}}(G, D)$$  \hspace{1cm} (12)

where $\lambda_{\text{adv}}$ denotes the hyperparameters that control the influence of the $\mathcal{L}_{\text{EGA}}(G, D)$, whose training objective is shown as follows:

$$G^*, D^* = \arg \min_{G} \max_{D} \mathcal{L}(G, D).$$  \hspace{1cm} (13)

F. Analysis

According to the theory of domain adaptation proposed by Ben-David et al. [24], let $\mathcal{H}$ be the hypothesis class, $S$ and $T$, respectively, express the source and the target domains, the theory defines the expected error on the target samples $\epsilon^T(h)$ as follows:

$$\forall h \in \mathcal{H}, \quad \epsilon^T(h) \leq \epsilon^S(h) + \frac{1}{2} d_{H\Delta T}(S, T) + \lambda$$  \hspace{1cm} (14)

where $\epsilon^S(h)$ represents the expected error on the source domain which can be minimized easily by predictions of the generator $G$ based on the source labels, $d_{H\Delta T}(S, T)$ measures the discrepancy distance between the source and the target distributions, and $\lambda$ is the shared expected loss expected to be negligibly small.
Followed by Ben-David et al. [24], \( d_{\text{H} \Delta \text{H}}(S, T) \) is minimized by the global domain adversarial learning. Recently, UDA semantic segmentation commonly use global margin alignment to narrow the discrepancy between domains [28]

\[
d_{\text{H} \Delta \text{H}}(S, T) = 2 \sup_{h \in \Gamma \triangle \Lambda} |P_{f \rightarrow S}[h(f) = 1] - P_{f \rightarrow T}[h(f) = 1]|
\]

As we have mentioned, the global feature alignment chases the global marginal distribution without considering the local discrepancies between domains. As shown in Fig. 1(c), the higher similarity of the class features between domains, the smaller the entropy value. Furthermore, the lower similarity corresponds to the higher entropy. Therefore, in the feature alignment process, it is essential to discriminatively treat different classes. Thus, we propose to use the entropy value to reflect the discrepancies between domains, and adaptively learn the attention of distinctive regions to better promote the alignment of features between domains.

Our method consists of two parts: LSR and EGA learning. LSR aims to enhance the dependence of local regions through the message propagation mechanism and aggregate context information, which enhances domain-invariant feature extraction ability of the generator. Meanwhile, EGA extends the traditional adversarial loss with a self-adaptive attention \( W_g^{(i,j)} = -(1/C) \sum_{c=1}^{C} P^{(i,j,c)} \log P^{(i,j,c)} \). \( W_g \) indicates the amount of uncertain information extracted from the local regions, while the entropy output corresponding to the region discrepancy compared to the source domain. Therefore, larger \( W_g \) encourages the generator \( G \) to fool discriminator \( D \) and pays more attention to the local regions while adversarial learning. Smaller \( W_g \) indicates that the generator \( G \) has learned enough domain-invariant knowledge in this region, thus weakens the adversarial learning process to remain the local feature alignment.

IV. EXPERIMENTS

To verify the effectiveness of the proposed method, we evaluate our method in comprehensive scenarios in both aerial and natural image segmentation datasets.

A. Setup

1) Datasets: We evaluate our method on a 2-D semantic segmentation benchmark dataset ISPRS. ISPRS is offered by the International Society for Photogrammetry and Remote Sensing 2-D Semantic Labeling Contest, which currently provides the best evaluation platform for aerial image semantic segmentation. It contains two subsets the Potsdam and the Vaihingen.\(^2\)

\(^2\)http://www2.isprs.org/commissions/comm3/wg4/2d-sem-label-potsdam.html

\(^2\)http://www2.isprs.org/commissions/comm3/wg4/2d-sem-label-vaihingen.html

Potsdam subset contains 38 aerial images covering 3.42 km\(^2\) area of the Potsdam city with a resolution of 5 cm. Those images are fixed with a size of 6000 × 6000 pixels in three channels: red, green, and blue. About 24 out of 38 images in Potsdam have pixel-annotation, which spans six categories: Building, Tree, Car, Impervious surfaces, Low vegetation, and Clutter/background.

Vaihingen subset contains 33 aerial images covering 1.38 km\(^2\) area of the Vaihingen city with a resolution of 9 cm. The size of each image is approximately 2000 × 2000 pixels in three different channels: near-infrared, red, and green. About 16 out of 33 images provide pixel-annotation, with the same categories classification as in Potsdam.

Following the protocol in [47], [48], and [50], for Potsdam (source domain)-to-Vaihingen (target domain) task, we use 24 labeled images with the paired labels in Potsdam and 17 unlabeled Vaihingen images as the training data, while the remaining 16 Vaihingen images with the paired labels are as the testing data. On the contrary, for the Vaihingen-to-Potsdam task, we use 16 Vaihingen labeled images with the paired labels and 14 Potsdam unlabeled images as training data, while the remaining 24 Potsdam labeled images for testing.

As preprocessing, we crop all the high-resolution images in the testing data. On the contrary, for the Vaihingen-to-Potsdam task, we use a training of 16 Vaihingen labeled images with the paired labels and 14 Potsdam unlabeled images as training data, while the remaining 24 Potsdam labeled images for testing.

As preprocessing, we crop all the high-resolution images in three different channels: near-infrared, red, and green. The size of each image is approximately 2000 × 2000 pixels in three different channels: near-infrared, red, and green. About 16 out of 33 images provide pixel-annotation, with the same categories classification as in Potsdam.

Following the protocol in [47], [48], and [50], for Potsdam (source domain)-to-Vaihingen (target domain) task, we use 24 labeled images with the paired labels in Potsdam and 17 unlabeled Vaihingen images as the training data, while the remaining 16 Vaihingen images with the paired labels are as the testing data. On the contrary, for the Vaihingen-to-Potsdam task, we use 16 Vaihingen labeled images with the paired labels and 14 Potsdam unlabeled images as training data, while the remaining 24 Potsdam labeled images for testing.

As preprocessing, we crop all the high-resolution images in both domains into 512 × 512 pixels by a sliding window with an overlap of 256 pixels.

2) Implementation Details: In the same manner as state-of-the-art methods, we employ the widely used DeepLab-V2 [13] framework with ResNet-101 [12] model pretrained on ImageNet [62] as the backbone. Then 1 × 1 convolution layer is used to reduce the channel \( m \) of the deep feature map from 2048 to 64. In our LSR model, we use a three-layer GCN network with the feature dimension as \{64, 512, 1024\} and insert the dropout layer between the GCN layer with a rate of 0.5, following a 1 × 1 convolution layer to recover the channel and match the size with the original deep feature map. Once obtaining the refined feature map, we combine the refined feature with the input feature by summation with a 1 × 1 convolution layer. Subsequently, Atrous Spatial Pyramid Pooling (ASPP) with dilated rates \{6, 12, 18, 24\} is employed to predict segmentation maps. Finally, an up-sampling layer with the softmax function output the prediction probability map with the size of the input image. Meanwhile, for the discriminator, we adopt the similar structure as in [63], which contains 5 convolution layers with kernel of 4 × 4, channel numbers of \{64, 128, 256, 512, 1\} and stride of 2. Each convolution layer follows a Leaky-ReLU parameterized by 0.2 except the last layer. Finally, the prediction of the discriminator is upsampled to the same size of the input for matching the size of the weight map. Empirically, following by our baseline [33] and other state-of-the-art domain adaption works [46], [50], we use different optimizers to optimize generator and discriminator. We use stochastic gradient descent (SGD) [64] with the initial learning rate as \( 2.5 \times 10^{-4} \) and a momentum of 0.9 as the optimizer for the generator. The discriminator optimizer employs Adaptive Moment Estimation (Adam) [65] with the initial learning rate \( 5 \times 10^{-3} \) and \( \beta_1 = 0.9, \beta_2 = 0.99 \). Theoretically, SGD [64] is a mainstream algorithm to optimize the generator [28]. SGD can speed up the convergence with...
a strong generalized optimal solution while avoiding local minimums. Adam [65] can speed up the convergence of the discriminator, thereby promoting the feature extraction of the generator for the domain-invariance. We set both optimizers a weight decay of $5 \times 10^{-4}$. Meanwhile, the learning rate decays by a poly learning rate policy. The initial learning rate is multiplied by $(1 - \frac{iter}{max\_iter})^{power}$ with power of 0.9. In all experiments, the maximum iterations are set to 100 k, and the batch size is 1. In our best model, the hyperparameters $\lambda_{adv}$, $\lambda_w$, and $\epsilon$ are set to 0.001, 5, and 0.6, respectively, on Potsdam-to-Vaihingen domain adaptation task. Our experiments are implemented in a PyTorch environment with a single NVIDIA Tesla V100-16G GPU.

Followed by the state-of-the-art methods [47], [48], we use four metrics to evaluate the performance of our method, pixel accuracy (PA), mean F1 score (mF1), intersection over union (IoU), and mean IoU (mIoU).

### B. Quantitative Comparison

1) Potsdam-to-Vaihingen: Table I reports the experimental results of our method comparing with the advanced methods on Potsdam to Vaihingen datasets. Due to the domain shift between the two datasets, the source-only method NoAdapt presents stumbling performance. Although the generative methods improve the results by a larger margin, they are hard to preserve the semantic information during the generation, which leads to limited performance. Generally speaking, the domain adaptation methods significantly boost the performance of our method due to their limited capability in some challenging categories such as “Low veg.” and “Clu./Back.”, which evidences the robustness of our method on challenging scenarios for aerial image semantic segmentation. In addition, summation operation outperforms concatenation operation in our model, which verifies the summation can enhance the information of the features without increasing the model parameters. We use the summation in the following experiments if not specified.

<table>
<thead>
<tr>
<th>Methods</th>
<th>adversarial learning</th>
<th>Reference</th>
<th>PA(%)</th>
<th>mF1(%)</th>
<th>Imp. surf.</th>
<th>Build</th>
<th>Low veg.</th>
<th>Tree</th>
<th>Car</th>
<th>Clu./Back</th>
<th>mIoU(%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>NoAdapt</td>
<td>generative</td>
<td>ICCV2017</td>
<td>46.26</td>
<td>36.19</td>
<td>26.86</td>
<td>41.67</td>
<td>12.56</td>
<td>44.48</td>
<td>16.70</td>
<td>24.01</td>
<td></td>
</tr>
<tr>
<td>CycleGAN [38]</td>
<td>generative</td>
<td>Remote Sens.2019</td>
<td>58.44</td>
<td>43.38</td>
<td>34.91</td>
<td>52.73</td>
<td>20.65</td>
<td>42.28</td>
<td>21.45</td>
<td>29.60</td>
<td></td>
</tr>
<tr>
<td>Benjida’s [47]</td>
<td>generative</td>
<td>Remote Sens.2019</td>
<td>53.50</td>
<td>43.09</td>
<td>31.87</td>
<td>53.26</td>
<td>23.67</td>
<td>41.76</td>
<td>21.70</td>
<td>29.31</td>
<td></td>
</tr>
<tr>
<td>CsDA [50]</td>
<td>generative &amp; subspace</td>
<td>Remote Sens.2019</td>
<td>55.98</td>
<td>45.65</td>
<td>44.82</td>
<td>53.39</td>
<td>20.74</td>
<td>42.04</td>
<td>23.81</td>
<td>32.56</td>
<td></td>
</tr>
</tbody>
</table>

2) Vaihingen-to-Potsdam: To further verify the effectiveness of our method, we evaluate our method by exchanging the source domain and target domain data as the Vaihingen-to-Potsdam. As shown in Table II, first of all, our method consistently beats the state-of-the-art methods as in the Potsdam-to-Vaihingen task as shown in Table I. Note that, the images in Vaihingen dataset cover a narrow area with a lower resolution with relatively rough annotation, while Potsdam includes images from a larger area with much higher resolution and more refined annotation. This brings more challenges in the Vaihingen-to-Potsdam domain adaptation task.

3) Synthetic-to-Real: To verify the generalization ability of our method, we conduct experiments of domain adaptation from two synthetic datasets GTA5 [22] and Synthia [23] to the real dataset Cityscapes [66], as shown in Table III (GTA5-to-Cityscapes) and Table IV (Synthia-to-Cityscapes). First, our method achieves promising performance against the state-of-the-art domain adaptation methods on natural images, which verifies the effectiveness of the proposed method. Second, our method focuses on local feature alignment and LSR thus can effectively reduce the impact of negative transfer, such as the infrequent classes “fence,” “pole,” “bike” in GTA5-to-Cityscapes, and “sign” in Synthia-to-Cityscapes. Third, the improvement on the natural dataset is not as significant as the infrequent classes “fence,” “pole,” “bike” in GTA5-to-Cityscapes, and “sign” in Synthia-to-Cityscapes.
TABLE II

<table>
<thead>
<tr>
<th>Methods</th>
<th>adversarial learning</th>
<th>Reference</th>
<th>PA(%)</th>
<th>mIoU(%)</th>
<th>Imp. surf. Build. Low veg. Tree Car Clu./Back. mIoU(%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>NoAdapt</td>
<td></td>
<td></td>
<td>44.33</td>
<td>37.20</td>
<td>35.78 37.68 12.90 14.60 33.33 9.35 23.97</td>
</tr>
<tr>
<td>CycleGAN [38]</td>
<td>generative</td>
<td>ICCV2017</td>
<td>51.69</td>
<td>44.94</td>
<td>36.08 30.14 23.84 31.24 25.65 12.95 29.99</td>
</tr>
<tr>
<td>Benjdira’s [47]</td>
<td>generative</td>
<td>Remote Sens 2019</td>
<td>52.26</td>
<td>47.24</td>
<td>36.69 42.73 30.64 33.41 34.56 14.38 32.07</td>
</tr>
<tr>
<td>CsDA [50]</td>
<td>generative &amp; subspace</td>
<td>Remote Sens 2019</td>
<td>45.25</td>
<td>40.36</td>
<td>32.33 37.25 22.40 26.60 43.29 0.01 27.00</td>
</tr>
<tr>
<td>Baseline</td>
<td></td>
<td></td>
<td>55.13</td>
<td>46.15</td>
<td>42.77 48.75 35.42 28.96 36.70 0.26 32.14</td>
</tr>
<tr>
<td>AdapSegNet [33]</td>
<td>subspace</td>
<td>CVPR2018</td>
<td>57.02</td>
<td>47.95</td>
<td>46.58 52.68 32.16 32.95 39.42 0.19 34.00</td>
</tr>
<tr>
<td>CLAN [34]</td>
<td></td>
<td>CVPR2019</td>
<td>60.03</td>
<td>50.20</td>
<td>49.80 54.55 40.19 26.94 46.71 0.21 36.45</td>
</tr>
<tr>
<td>MRNet [59]</td>
<td></td>
<td>JICA2020</td>
<td>59.33</td>
<td>48.54</td>
<td>47.96 49.95 42.25 26.50 41.00 0.17 34.64</td>
</tr>
<tr>
<td>Ours</td>
<td>subspace</td>
<td></td>
<td>63.01</td>
<td>51.01</td>
<td>47.47 63.19 42.57 30.93 39.42 0.36 37.32</td>
</tr>
</tbody>
</table>

TABLE III

<table>
<thead>
<tr>
<th>Methods</th>
<th>Reference</th>
<th>road</th>
<th>side</th>
<th>bul.</th>
<th>wall</th>
<th>fence</th>
<th>pole</th>
<th>light</th>
<th>sign</th>
<th>veg.</th>
<th>text</th>
<th>sky</th>
<th>pers.</th>
<th>rid.</th>
<th>car</th>
<th>bus</th>
<th>train</th>
<th>motor</th>
<th>bike</th>
<th>mIoU</th>
</tr>
</thead>
<tbody>
<tr>
<td>NoAdapt</td>
<td></td>
<td>75.8</td>
<td>16.5</td>
<td>77.7</td>
<td>12.5</td>
<td>21.0</td>
<td>23.5</td>
<td>30.1</td>
<td>20.1</td>
<td>83.1</td>
<td>24.6</td>
<td>70.3</td>
<td>53.8</td>
<td>26.4</td>
<td>49.9</td>
<td>17.2</td>
<td>25.7</td>
<td>6.5</td>
<td>25.3</td>
<td>36.0</td>
</tr>
<tr>
<td>Baseline</td>
<td></td>
<td>86.5</td>
<td>25.9</td>
<td>79.7</td>
<td>22.1</td>
<td>20.2</td>
<td>23.2</td>
<td>33.1</td>
<td>21.8</td>
<td>81.7</td>
<td>25.7</td>
<td>75.9</td>
<td>57.3</td>
<td>26.2</td>
<td>76.3</td>
<td>29.8</td>
<td>32.1</td>
<td>7.2</td>
<td>28.6</td>
<td>32.5</td>
</tr>
<tr>
<td>AdapSegNet [33]</td>
<td>CVPR2018</td>
<td>86.5</td>
<td>36.0</td>
<td>79.9</td>
<td>23.4</td>
<td>23.3</td>
<td>23.9</td>
<td>35.2</td>
<td>14.8</td>
<td>83.4</td>
<td>33.3</td>
<td>75.6</td>
<td>58.5</td>
<td>27.6</td>
<td>73.7</td>
<td>32.5</td>
<td>35.4</td>
<td>3.9</td>
<td>30.1</td>
<td>28.1</td>
</tr>
<tr>
<td>CLAN [34]</td>
<td>CVPR2019</td>
<td>89.4</td>
<td>33.1</td>
<td>81.0</td>
<td>26.6</td>
<td>35.8</td>
<td>27.2</td>
<td>33.5</td>
<td>24.7</td>
<td>83.9</td>
<td>36.7</td>
<td>78.5</td>
<td>58.7</td>
<td>30.5</td>
<td>83.8</td>
<td>38.5</td>
<td>34.3</td>
<td>1.7</td>
<td>31.6</td>
<td>32.4</td>
</tr>
<tr>
<td>MRNet [59]</td>
<td>JICA2020</td>
<td>88.4</td>
<td>36.5</td>
<td>81.2</td>
<td>25.9</td>
<td>22.6</td>
<td>32.0</td>
<td>35.2</td>
<td>23.6</td>
<td>82.3</td>
<td>27.3</td>
<td>73.5</td>
<td>61.8</td>
<td>31.3</td>
<td>83.5</td>
<td>32.2</td>
<td>38.5</td>
<td>8.1</td>
<td>30.7</td>
<td>39.2</td>
</tr>
<tr>
<td>Ours</td>
<td></td>
<td>88.4</td>
<td>36.5</td>
<td>81.2</td>
<td>25.9</td>
<td>22.6</td>
<td>32.0</td>
<td>35.2</td>
<td>23.6</td>
<td>82.3</td>
<td>27.3</td>
<td>73.5</td>
<td>61.8</td>
<td>31.3</td>
<td>83.5</td>
<td>32.2</td>
<td>38.5</td>
<td>8.1</td>
<td>30.7</td>
<td>39.2</td>
</tr>
</tbody>
</table>

TABLE IV

<table>
<thead>
<tr>
<th>Methods</th>
<th>Reference</th>
<th>road</th>
<th>side</th>
<th>bul.</th>
<th>wall</th>
<th>fence</th>
<th>pole</th>
<th>light</th>
<th>sign</th>
<th>veg.</th>
<th>text</th>
<th>sky</th>
<th>pers.</th>
<th>rid.</th>
<th>car</th>
<th>bus</th>
<th>train</th>
<th>motor</th>
<th>bike</th>
<th>mIoU</th>
</tr>
</thead>
<tbody>
<tr>
<td>NoAdapt</td>
<td></td>
<td>55.6</td>
<td>23.8</td>
<td>74.6</td>
<td>6.1</td>
<td>12.1</td>
<td>74.8</td>
<td>79.0</td>
<td>55.3</td>
<td>19.1</td>
<td>39.6</td>
<td>23.3</td>
<td>13.7</td>
<td>25.0</td>
<td>38.6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Baseline</td>
<td></td>
<td>79.2</td>
<td>37.2</td>
<td>78.8</td>
<td>9.9</td>
<td>10.5</td>
<td>78.2</td>
<td>80.5</td>
<td>53.5</td>
<td>19.6</td>
<td>67.0</td>
<td>29.5</td>
<td>21.6</td>
<td>31.3</td>
<td>45.9</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AdapSegNet [33]</td>
<td>CVPR2018</td>
<td>84.3</td>
<td>42.7</td>
<td>77.5</td>
<td>4.7</td>
<td>7.0</td>
<td>77.9</td>
<td>82.5</td>
<td>54.3</td>
<td>21.0</td>
<td>72.3</td>
<td>32.2</td>
<td>18.9</td>
<td>32.3</td>
<td>46.7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CLAN [34]</td>
<td>CVPR2019</td>
<td>85.6</td>
<td>42.2</td>
<td>79.7</td>
<td>5.4</td>
<td>8.1</td>
<td>80.4</td>
<td>84.1</td>
<td>57.9</td>
<td>23.8</td>
<td>35.3</td>
<td>36.4</td>
<td>14.2</td>
<td>33.0</td>
<td>48.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MRNet [59]</td>
<td>JICA2020</td>
<td>81.3</td>
<td>37.0</td>
<td>80.1</td>
<td>16.1</td>
<td>16.1</td>
<td>78.2</td>
<td>83.3</td>
<td>53.4</td>
<td>21.2</td>
<td>73.9</td>
<td>32.9</td>
<td>22.6</td>
<td>30.7</td>
<td>47.8</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Ours</td>
<td></td>
<td>85.4</td>
<td>39.0</td>
<td>79.0</td>
<td>15.5</td>
<td>20.6</td>
<td>77.6</td>
<td>82.7</td>
<td>55.2</td>
<td>21.8</td>
<td>77.3</td>
<td>30.8</td>
<td>12.1</td>
<td>31.4</td>
<td>50.3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

as on the aerial image dataset. The main reason is, compared with natural images, the large-scale objects in aerial images tend to lose more semantic information during cropping. Meanwhile, there are larger discrepancies between classes in true orthophoto of the aerial image. Our method is more suitable for aerial image segmentation through LSR to enhance feature representation and information entropy to promote local feature alignment.

C. Qualitative Comparison

Fig. 4 visualizes some qualitative segmentation examples obtained by our method compared to the state-of-the-art competitors on Potsdam-to-Vaihingen. Due to the domain shift between the Potsdam and the Vaihingen, the predictions of NoAdapt [Fig. 4(a)] introduce much noise and lose the object boundary and structure information. The segmentation results of the generative adversarial learning methods, CyCleeGAN [38] and the Benjdira’s [47] as shown in Fig. 4(b) and (c) are also unsatisfactory for the limited generated images by global consistency constraint. CsDA [50] as shown in Fig. 4(d), achieves a significant improvement by introducing extra feature generation and output-space adversarial learning that considers the spatial distribution information. However, it still suffers from unclear object boundaries such as the “Build.” category and confuses misclassification such as “Low veg.” and “Tree” categories. The Baseline, AdaptSegNet [33], and Advent [46] pursue the global margin alignment, which results in much misclassification such as “Tree” and “Low veg.” Furthermore, the “Build.” boundaries are still rough and involving many holes, as shown in Fig. 4(e)–(g). CLAN [34] is able to generate more precise and accurate predictions by considering the category-level data distribution and local semantic consistency, as shown in Fig. 4(h). However, due to the large appearance discrepancy, such as the left and the right “Build.” comparing to the middle “Build.” as shown in the second row, it suffers the challenge of intradomain discrepancy for aerial image semantic segmentation. MRNet [59] presents refined segmentation results by considering the intradomain adaptation. However, it fails to isolated individuals from each other, such as “Build.”
Fig. 4. Visual Results of our proposed method compared with other state-of-the-art methods on Potsdam-to-Vaihingen. The blue, green, yellow, cyan, white, and red separately represent the categories of “Build.,” “Tree,” “Car,” “Low veg.,” “Imp. surf.,” and “Clu./Back.” (a) NoAdapt, (b) CycleGAN, (c) Benjdira’s, (d) CsDA, (e) Baseline, (f) AdaptSegNet, (g) Advent, (h) CLAN, (i) MRNet, (j) Ours.

TABLE V
ABLAATION STUDIES ON THE POTSDAM-TO-VAIHINGEN DOMAIN ADAPTATION

<table>
<thead>
<tr>
<th>Method</th>
<th>PA(%)</th>
<th>mIoU (%)</th>
<th>Imp. surf.</th>
<th>Build.</th>
<th>Low veg.</th>
<th>Tree</th>
<th>Car</th>
<th>Clu./Back.</th>
<th>mIoU(%)</th>
<th>PAD</th>
<th>Cost( /iteration)</th>
</tr>
</thead>
<tbody>
<tr>
<td>B1</td>
<td>61.76</td>
<td>46.94</td>
<td>54.05</td>
<td>60.82</td>
<td>21.78</td>
<td>56.67</td>
<td>15.56</td>
<td>0.40</td>
<td>34.87</td>
<td>0.90</td>
<td>0.41</td>
</tr>
<tr>
<td>B2</td>
<td>65.21</td>
<td>52.72</td>
<td>53.81</td>
<td>66.65</td>
<td>28.32</td>
<td>58.84</td>
<td>23.37</td>
<td>5.42</td>
<td>39.40</td>
<td>0.54</td>
<td>0.54</td>
</tr>
<tr>
<td>B3</td>
<td>64.32</td>
<td>50.92</td>
<td>55.40</td>
<td>66.14</td>
<td>24.98</td>
<td>57.04</td>
<td>24.45</td>
<td>1.81</td>
<td>38.14</td>
<td>0.72</td>
<td>0.42</td>
</tr>
<tr>
<td>B4</td>
<td>67.65</td>
<td>53.56</td>
<td>57.11</td>
<td>70.24</td>
<td>28.73</td>
<td>56.55</td>
<td>23.65</td>
<td>5.84</td>
<td>40.35</td>
<td>-</td>
<td>0.55</td>
</tr>
</tbody>
</table>

in the second and fourth rows in Fig. 4(i). Furthermore, it presents obvious misclassification, such as the “Low veg.” misclassified as “Tree” as shown in the sixth row. Our proposed method with the LSR model explores the long-range dependencies of the input image for structure information and self-adaptive adversarial learning for local feature alignment. Thus it produces more precise predictions and preserves better object boundary information, as shown in Fig. 4(j). Especially, we achieve much accurate prediction on small objects such as “Car” than other methods by the local feature alignment, as shown in the fourth and eighth rows. By comparing the third, fourth, and fifth rows of Fig. 4(j) with other state-of-the-art methods, our method can predict a more precise “Build.” boundary. Even though some “Build.” boundary is still rough in our method, as shown in the second row of in Fig. 4(j), our method can effectively separate them from adjacent objects. Furthermore, due to the sensor variation between domains, the “Low veg.” and “Tree” result in high similarity in appearance, as shown in the last two rows. This brings a big challenge for the state-of-the-art methods while can be successfully segmented by our method benefit from the proposed EGA learning. The key reason is we consider the discrepancies of local regions between domains and the weight of information entropy to guide the local feature alignment.

D. Component Analysis

1) Ablation Study: To further verify the contribution of the components in our model, we conduct the ablation study on several variants of our method on Potsdam-to-Vaihingen domain adaptation task, as reported in Table V. B1 is our Baseline, which takes a global feature alignment base on the DeepLab-V2 framework. B2 embeds GCN-based LSR on Baseline B1 to explore the region correlation and the structure information of the input. B3 combines the information entropy of the target domain input to guide the adversarial process, called EGA on Baseline B1. B4 is our final model, which combines both LSR and EGA to the Baseline. By the information passing mechanism, the embedded LSR model can capture the structure information in the input image, which purges the prediction map, especially on large-scale objects. Therefore, B2 achieves significant improvement comparing with B1, as shown in Table V. By arguing the entropy map.
representing the interdomain shift, the EGA model can refine the segmentation results by the local aligned method, especially on the local regions and the small objects. Compared B1 with B3 in Table V, it demonstrates the EGA model can effectively improve the segmentation accuracy. By integrating both LSR and EGA, our final model B4 further improves the segmentation results, which verifies the contribution of the two components. In addition, we evaluate the computational cost of each module on the baseline during the training as shown in Table V. Both LSR and EGA modules significantly increase the segmentation accuracy without introducing distinct processing costs.

Fig. 5 further visualizes the segmentation results of the variants. Comparing the results of B2 with B1, the “Build.” presents more refined boundaries. However, the two categories “Tree” and “Low veg.” around “Build.” tend to be one category due to the high interclass similarity. The prediction error map in B3 is significantly better than B1. Since EGA concerns the interdomain discrepancy while simultaneously preserves the well-aligned features, B3 can pay different attention to different regions by information map for the domain-variant categories “Tree” and “Low veg.”, which leads to significantly better segmentation than B1 and B2. And, our EGA can preserve well-aligned features and accelerate the unaligned features as a local feature alignment approach. Therefore, B4 achieves a more accurate segmentation performance by combining both LSR and EGA modules, which reinforce each other during the learning.

2) Evaluation on LSR: To evaluate the contribution of the proposed LSR via capturing the long-range semantic dependencies, we further visualize the feature response maps of the DA network on Potsdam-to-Vaihingen task as shown in Fig. 6. By comparing the feature response maps with/without LSR, the feature maps with LSR have high responses by considering the relations and aggregative context information between different regions. Specifically, the small objects have high responses which will not be discarded during the feature extraction, such as the “Tree” in the second and the sixth columns, and the “Car” in the fourth column. The big objects present more clear boundary responses after LSR, shown as the “Build.” in the fourth and the fifth columns, and the “Imp. surf.” in the third column.

3) Evaluation on EGA: Fig. 7 visualizes the segmentation results and the corresponding feature distribution maps by t-SNE [67] of the source-only method NoAdapt (a1), Baseline (b1) and Baseline with EGA (c1) on Potsdam-to-Vaihingen. And the h-dimensional features of (a1)–(c1) are mapped into a 2-D space with t-SNE [67] in (a2)–(c2). Each color represent a class that has drawn to the upper right of distribution maps. The comparison of feature distribution confirms that our method can enhance the feature alignment.
the car category in the second and third rows. By intro-
ducing the proposed EGA to the Baseline to balance the
local and the global feature alignment, Baseline with EGA
as shown in Fig. 7(c1) can well promote the alignment in the
domain-variant regions. Furthermore, by comparing the feature
distribution in Fig. 7(a2)–(c2), EGA can reduce category
confusion while maintaining the original aligned categories by
considering the local feature alignment approach, such as the
“Build,” in the first and third rows and the “Car” in the second
and third rows.

In addition, we use Proxy $A$-distance (PAD) to measure the
domain distance followed by Ben-David et al. [24]. The
PAD is calculated based on the generalization error $\epsilon$ by a
classifier to discriminate source and target examples as
\[
\hat{d}_A = 2(1 - 2\epsilon).
\]

Herein, we keep the discriminator $D$ as the classifier in our
experiments. As reported in Table V, EGA effectively reduces
PAD, which indicates that considering the discrepancy in local
regions can better promote feature alignment and reduce the
domain distance.

E. Other Analysis

1) Convergence Analysis: In addition, we use the loss curve
of the generator $G$ and the discriminator $D$ to indicate the
convergence performance, as shown in Fig. 8. We can see the
loss of the $D$ drops rapidly at the beginning, which shows
that $D$ can effectively distinguish the shift between domains.
In the initial training, the $G$ has extracted a lot of uncertain
information on the target domain due to the domain shift,
our proposed EGA method will increase the weight of the
uncertainty region in adversarial learning, and the loss of
$D$ will increase. As $G$ learns more domain-invariant knowl-
dge, the uncertainty information on the target domain decreases, and the reduced weight weakens the adversarial
loss of the domain-invariant region to maintain local feature alignment. Therefore, the loss of $D$ declines and gradually
converges to a stable value. As the ability of the discriminator gradually weakens, the loss cure of $G$ gradually smooths and
converges to 0.

2) Parameter Analysis: To evaluate the impact of the hyper-
parameters in our proposed method, we perform sensitiv-
ity study on Potsdam-to-Vaihingen domain adaptation task
about the parameters of the self-adaptive EGA loss ($\epsilon$, $\lambda_w$) in (11), the weight of adversarial loss ($\lambda_{adv}$) in (12),
the reduced dimension parameter ($m$), and the results are
reported in Table VI. We design $\lambda_w$ and $\epsilon$ to balance local
and global features alignment. Specifically, $\lambda_w$ controls the
self-adaptive weight for adversarial loss, if $\lambda_w$ is large then
the effect of the entropy weight dominates and the model is
strongly biased toward a few classes. $\epsilon$ represents the influence of
traditional global adversarial loss during training, which is
limited to the range between (0, 1), while larger $\epsilon$ will bias to
the global feature alignment. Following prior works [33], [46],
[50], we set $\lambda_{adv}$ around 0.001. The parameter $m$ is used to
reduce the number of channels in the feature maps to balance the
performance and computation resources. We vary each of
the four parameters while fixing the other three. From Table VI
we can see, our method is not sensitive to the parameters.
We empirically set \{ $\epsilon$, $\lambda_w$, $\lambda_{adv}$, $m$ \} = \{0.6, 5, 0.001, 64\} for
the best performance.

V. Conclusion

In this article, we propose a GCN-based LSR and EGA
learning network for UDA of aerial image semantic seg-
mentation. LSR encodes the dependencies between regions and
learns the structure information by the information pass-
ning mechanism of the graph structure. EGA uses the informa-
tion entropy to guide the adversarial learning between the
two domains, encouraging the network to retain the
domain-invariant features and promote the alignment of the
domain-variant features. Experiments on the two benchmark
datasets demonstrate the robustness and effectiveness of our
proposed method, which yields competitive performance com-
pared with other state-of-the-art methods. In the future, we will
introduce the self-supervised learning method in our frame-
work to provide additional supervised information for further
performance boost.

References

J. Garcia-Rodriguez, “A review on deep learning techniques applied to
http://arxiv.org/abs/1704.06857


suburban area—Comparison of high-resolution remotely sensed datasets
using classification trees and test field points,” Remote Sens., vol. 3,

L. Mou, Y. Hua, and X. X. Zhu, “Relation matters: Relational context-


J. Yuan, “Learning building extraction in aerial scenes with convolutional


Authorized licensed use limited to: Anhui University. Downloaded on December 15,2022 at 14:59:51 UTC from IEEE Xplore. Restrictions apply.