CCENet: Cascade Class-Aware Enhanced Network for High-Resolution Aerial Imagery Semantic Segmentation

Qiongxiong Wang, Student Member, IEEE, Xiaoyan Luo, Member, IEEE, Jiaqi Feng, Sen Li, and Jihao Yin, Senior Member, IEEE

Abstract—Semantic segmentation of high-resolution aerial images is a challenging task on account of interclass homogeneity and intraclass heterogeneity of land cover. Recent works have sought to mitigate this issue by exploiting pixelwise global contextual information using self-attention mechanism. However, the existing attention-based methods usually generate inaccurate object boundary segmentation results, as the self-attention model is embedded in high-level features with low resolution due to prohibitively computational complexity. Moreover, existing attention-based models ignore class-aware contextual information from intermediate results, which leads to undesirable feature separability. To obtain discriminative feature as well as generate accurate segmentation boundaries, we present a novel segmentation framework, named cascade class-aware enhanced network (CCENet) for high-resolution aerial imagery. The proposed CCENet predicts segmentation results on multiple stages, and the result of the previous stage is used to refine object boundary details for the latter stage. To exploit the class-aware prior information in previous stage, we propose a lightweight class-aware enhanced module (CaEM) to grab the class-aware contextual dependencies. Specifically, CaEM first extracts a set of class representation of the land covers by global class pooling block and then reconstructs enhanced features using class relation measurement, which alleviates the interclass homogeneity and intraclass heterogeneity of ground objects in feature space. Quantitative and qualitative experimental results on three publicly available datasets demonstrate the superiority of our CCENet over other state-of-the-art methods in the items of high labeling accuracy and computation efficiency.

Index Terms—Aerial imagery, class context information, deep convolution neural networks, self-attention mechanism, semantic segmentation.

I. INTRODUCTION

WITH the recent development of remote sensing (RS) technologies, the spatial resolution of RS images can reach less than 10 cm. Based on the fine description of high-resolution aerial images, automatic interpretation plays an important role in a widespread of applications, such as road extraction [1], traffic monitoring [2], urban planning [3], intelligent agriculture [4], and disaster management [5]. To aid scene understanding, semantic segmentation for aerial images (i.e., semantic labeling) is a very crucial step by inferring pixelwise semantic class labels.

In semantic segmentation of high-resolution aerial images, interclass homogeneity and intraclass heterogeneity are two issues [6], [7]. On one hand, the aerial images are acquired from a bird’s view with less common structural information, which could make the interclass homogeneity of land covers worse. Fig. 1 illustrates an examples of this issue. For example, the roof of buildings presents very similar visual characteristics to the road surface. On the other hand, high spatial resolution aerial images contain only a few spectral channels, which is insufficient to narrow the intraclass heterogeneity caused by various appearances, such as the different roofs in Fig. 1. In recent years, semantic segmentation of high-resolution aerial images has achieved remarkable progress, owing to the emergence of deep convolutional neural network (CNN) and massive training data [8], [9], [10]. To adaptively discover patterns in the high-resolution aerial images instead of traditional hand-crafted feature [11], fully convolutional network (FCN) [10] is used to automatically learn the features by sliding the filters (convolutional kernels) at all locations. However, due to the lack of contextual information caused by limited receptive fields of convolutional kernels [12], [13], [14], the features learned by CNN are usually not discriminative to solve the problems on interclass homogeneity and intraclass heterogeneity.

To exploit richer contextual information, some research works in computer vision focus on enlarging the network receptive field. The early study is mainly about increasing the spatial scale of contexts, including large kernel convolution [12], atrous spatial pyramid pooling (ASPP) [13], and pyramid pooling module (PPM) [14], to exploit multiscale contexts. These multiscale context improvements have been proved effective for high-resolution aerial images [15], [16], [17], [18]. Nevertheless, these strategies ignore the long-range relational dependencies between the objects and scenes. In order to aggregate long-range relational context, self-attention mechanism [19], [20] is widely adopted in semantic segmentation. Specifically, OCNet [21],
DANet [22] and relation-augmented FCN (RAFCN) [23] calculate the weighted point-by-point attention map and reconstruct the feature of each pixel by fusing information from all positions. However, when processing high-resolution aerial images, the point-by-point attention-based methods face heavy calculation. For instance, given an input feature map with size $C \times H \times W$, generating the similarities map of all locations between each other requires a matrix multiplication of computational complexity $\mathcal{O}(CH^2W^2)$, which is prohibitively tremendous for high-resolution aerial images.

Generally, most existing works simply embed the self-attention model in high-level features with low resolution due to the huge computational complexity, which leads to inaccurate object boundary segmentation results. It can be seen in Fig. 1(c) that the feature from RAFCN [23] has defective boundary details. Moreover, the above methods hold less discriminative ability to learn features. For example, Fig. 1(c) shows the RAFCN feature cosine similarity visualization of all pixels to the pixel marked by yellow cross sign. It can be seen that RAFCN fails to distinguish the confused building and impervious surface. The main reason is that the feature learned by self-attention module collect all the pixel representation, which could bring background interference during feature learning. Moreover, the intermediate feature of network lacks global class-aware prior knowledge, i.e., no direct supervision from groundtruth label, thus leading to undesirable feature separability.

To promote the discriminative ability and generate accurate segmentation boundaries, we present a novel segmentation framework, named cascade class-aware enhanced network (CCENet) for high-resolution aerial imagery. The key insight of CCENet is to refine object boundary details recursively under the guidance of previous output. Specifically, CCENet predicts segmentation results on multiple stages, and the result of the previous stage is used to refine object boundary details for the latter stage. To exploit the class-aware prior information in previous stage, a lightweight class-aware enhanced module (CaEM) is proposed to grab the class-aware contextual dependencies. Specifically, previous segmentation result and feature map are first used to extract class representation. Then the class representation is utilized to measure the similarity with feature map to get the refined result, which is supervised by our proposed similarity loss. Last, the enhanced feature is reconstructed by the class representation and refined result. The proposed CaEM has similar structure with OCR [24], which reduces the attention computational complexity from $\mathcal{O}(CH^2W^2)$ to $\mathcal{O}(CHWK)$ with $K$ classes in dataset, so that CaEM can be applied into low-level feature with high-resolution details.

To summarize, our main contributions are as follows.

1) We design a coarse-to-fine CCENet to deliver the global class-aware prior information from the deep layer to the following portion, which refines object boundary details in a recursive paradigm for high-resolution aerial imagery labeling. It can be seen in Fig. 1(c) and (d) that our CCENet can learn more distinguishable object boundaries compared with RAFCN [23].

2) We propose a novel lightweight attention module CaEM to boost the representational ability of the network by introducing metric learning. The proposed CaEM similar to OCR to some degree but with less computation load by directly measuring the pixel-class relation with feature cosine similarity instead of matrix multiplication, which simultaneously excludes the interference of other class representation. As shown in Fig. 1(d), our method can exclude the interference of irrelevant land covers for learning more discriminative features.

3) We design a series of experiments to demonstrate the effectiveness of the proposed CCENet on the challenging International Society for Photogrammetry and Remote Sensing (ISPRS) Potsdam and Vaihingen datasets. Quantitative and qualitative results show that our model outperforms other state-of-the-art methods.

The remainder of this article is structured as follows. Related work is briefly reviewed in Section II. Section III introduces the proposed approach for high-resolution aerial imagery segmentation in detail. The effectiveness of the proposed method and ablation studies is demonstrated in Section IV by results on three real high-resolution aerial imagery datasets. Finally, Section V concludes this article and suggests future research directions.

II. RELATED WORKS

A. Architecture of Semantic Segmentation

Over the past few years, the breakthrough of deep CNN has led to remarkable progress in semantic segmentation. FCN [10]
is the first end-to-end segmentation network, which converts the fully connected layers in classification network [8] into the convolutional layers for the dense pixelwise labeling. Subsequently, plenty of variants were proposed to improve the performance of FCN. In order to prevent the loss of spatial detailed information caused by downsampling operations, SegNet [25] saves pooling index information and conducts nonlinear upsampling to recover the spatial details. U-Net [26] concatenates the downsampled feature in encoder with the upsampled ones in decoder via designing a U-shape structure. HRNet [27] generates and maintains high-resolution representations during the whole process, which becomes a widely used backbone in semantic segmentation. To refine segmentation results, coarse-to-fine concept has been adopted in segmentation framework. Early researchers tried to address this issue with graphical models such as CRF [28]. However, these methods rely on low-level color boundaries without leveraging high-level semantic information and cannot fix large error regions. In order to recover a precise boundary, low-level texture features are utilized by skip-connection into the deeper layers. For example, RefineNet [29] merges all the information available along the downsampling process to enable high-resolution prediction. CascadePSP [30] refines boundaries in different resolution using PPM. However, they ignore global class-aware prior information from the deep layer when refining the object boundary. In contrast, our cascade architecture can output multiple intermediate results, which is supervised by groundtruth during end-to-end training. This concept is similar to deep supervision [31], [32], and the difference lies in that our CCENet can transfer the global class prior information from previous stage to the following portion, which can further improve the feature separability and refine object boundary.

B. Context Aggregation Model

Other than detailed refinement achieved by structural innovation, many researchers try to improve segmentation result by designing plug-and-play model for aggregating more contextual information. For the purpose of exploiting multiscale context, GCN [12] adopts global convolutional module and global pooling to capture global context information. DeepLabv3+ [13] employs ASPP consisting of parallel convolutions with different dilated rates to increase the receptive field. PSPNet [14] introduces a PPM in which different kernel pooling layers are applied to aggregate multiscale contextual information. DMNet [33] utilizes multiple dynamic convolutional modules arranged in parallel, each of which explores context-aware filters for a specific scale. Different from multiscale context aggregation model, self-attention mechanism [19], [20] is more efficient to aggregate long-range relational context information. Typically, OCNet [21], DANet [22], and RAFCN [23] calculate the weighted point-by-point attention map, i.e., the relation between pixels, and augment the feature of each pixel by fusing information from all positions. However, generating the weighted attention map consumes tremendous computing and memory resources, which hinders its usage in real-time operations of an aerial platform.

Several works are proposed to reduce the computation load and memory usage of attention-based methods. Specifically, CCNet [45] collects relational context of all the positions by stacking two serial cross-cross attention module. EMA-Net [34] and asymmetric nonlocal neural network [35] explore a group of global descriptors to reconstruct the feature maps instead of treating all pixels themselves as the reconstruction descriptors. ACFNet [31] and OCR [24] are related to our method, which further improves the global descriptors with class information. Our proposed CaEM is inspired by the above relational context approach [24], [35]. The main difference between CaEM and other relational context model lies in similarity calculation and explicit similarity loss supervising the intermediate results. Specifically, our CaEM measures the pixel-class relation by feature cosine similarity without interference of other class. Besides, inspired by metrics learning [36] and deep supervision [32], we propose similarity loss to explicitly supervise the intermediate result, which alleviates the interclass homogeneity and intra-class heterogeneity problem.

C. Semantic Segmentation of Aerial Imagery

Compared with natural images, semantic segmentation for aerial images is more challenging, as the aerial images are captured from a bird’s view with less common structural information and few spectral channels. This aggravates the interclass homogeneity and intra-class heterogeneity [6], [7] problems. Several works have been proposed to learn more discriminative feature for semantic segmentation of high-resolution aerial images. For example, Tree-UNet [37] adaptively constructs the tree-shape convolutional blocks though the tree-cutting algorithm to fuse the multiscale features and learn the best weights. CSE-HRNet [7] adopts nested dilated residual block to enhance the representational power of multiscale contexts. ScasNet [17] proposes a self-cascaded encoder–decoder network to improve the segmentation by sequential global-to-local context aggregation and object refinement subnetworks. DDCM-Net [38] combines the dilated convolutions merged with varying dilation rates to enlarge the network’s receptive fields. CAM-DFCN [39] is introduced to automatically weight the channels of feature maps to perform feature selection.

In order to grep global spatial contextual information, attention mechanism has been introduced into semantic segmentation of aerial imagery. Typically, RAFCN [23] proposes spatial relation module and channel relation module to learn relationships between any two positions. SSAtNet [40] proposes pyramid attention pooling module to introduce the attention mechanism into the multiscale module for adaptive features refinement. In order to reduce the large time and space demands of self-attention operation, MSCA [41] adopts a multibranch spatial-channel attention model to efficiently extract global dependency and combine it with multiscale and channel-attention methods. HMANet [42] adaptively captures global correlations from the perspective of space, channel, and category in an efficient manner. MANet [19] proposes a novel attention mechanism of kernel attention with linear complexity and adopts multiple attention modules to extract contextual dependencies.

However, the above models aim to learn the distribution across the entire dataset, so that they balance all the classes
on different aerial images. Therefore, some categories could be depressed for one specific aerial image. Inspired by the dynamic prototype extraction of few-shot semantic segmentation [43], our work introduces class-aware prior information to ensure the discrimination of each category for each specific image in a recurrent manner, which is helpful for the pixel classification in this image.

III. APPROACH

A. Overall Framework

In high-resolution aerial images, both high-level contextual information and low-level texture features are vital for pixelwise segmentation task. Thus, we propose CaEM module to enhance high-level contextual information integrated class label prior information. To further keep low-level texture feature, we construct a novel cascade segmentation framework CCENet based on multiple-level recursive CaEM. The overall structure of our proposed CCENet is shown in Fig. 2, which follows the similar coarse-to-fine framework [17], [29], [30].

To generate feature maps with different resolutions and initial segmentation, we first use the dilated ResNet-101 [13], [22] as encoder. Specifically, the dilation convolutions are used to improve the convolutional layers, and the last downsampling operation is removed, which can preserve the details of land covers as well as to enlarge the receptive fields of the feature map. Based on this modified encoder, an initialization segmentation map is 1/8 size of the input image instead of 1/16 in original ResNet-101. Subsequently, we propose a cascade class-aware enhanced decoder to transfer the global class-aware prior information from the bottom-up. The decoder reutilizes the low-level features from shallow layers by long-range connections, so that the low-level details can be recovered. For each stage, the enhanced process can be formulated as follows:

\[
E^t = \begin{cases} 
F^t + \delta \left[ f_{\text{CaEM}} \left( E^{t-1}, P^{t-2} \right) \right], & t = 2, 3, 4 \\
F^4, & t = 1
\end{cases}
\]  

For stage-\( t \), \( E^t \) and \( F^t \) denote the enhanced feature map and raw feature map, respectively, and \( P^t \) is the segmentation result. Notably, \( E^1 \) is initialized as \( F^1 \) at the stage-1, and \( F^4 = 0 \) as there is no corresponding feature map for the last stage. \( f_{\text{CaEM}}(\cdot) \) represents the mapping function of our proposed CaEM, which will be described in Section III-B. \( \delta(\cdot) \) is a transformation function for multilevel feature fusion, which implements bilinear interpolation, \( 3 \times 3 \) convolution, ReLU [44], and batch normalization step-by-step. After three refinement stages, the last stage feature is refined through bilinear interpolation, \( 3 \times 3 \) convolution layer, and softmax layer to get the final segmentation result.
TABLE I
DETAILED CONFIGURATION OF CCENet

| Layer Name           | Input (size)                        | Output (size)                        | Details                                      |
|----------------------|--------------------------------------|--------------------------------------|----------------------------------------------|
| Resnet stem          | Image I (3, H, W)                    | F1 (64, H/2, W/2)                    | 7×7 convolution (stride 2), bn, relu          |
| Resnet layer1        | F1 (3, H/2, W/2)                     | F2 (256, H/4, W/4)                   |                                               |
| Resnet layer2->4     | F2 (256, H/4, W/4)                   | F3 (2048, H/8, W/8)                  | Removing last down-sampling                  |
| initial segmentation | F1 (2048, H/8, W/8)                  | P0 (K, H/8, W/8)                     |                                               |
| Stage1 refinement    | F1 (2048, H/8, W/8)                  | P0 (K, H/8, W/8)                     | CaEM_1, Bilinear interpolation               |
|                      |                                      | E1 (2048, H/4, W/4)                  | 3×3 convolution, softmax                      |
| Stage2 refinement    | F1 (2048, H/4, W/4)                  | P1 (K, H/4, W/4)                     | CaEM_2, Bilinear interpolation               |
|                      |                                      | E2 (256, H/2, W/2)                   | 3×3 convolution, relu, bn                     |
| Stage3 refinement    | F1 (256, H/2, W/2)                   | P2 (K, H/2, W/2)                     | CaEM_3, Bilinear interpolation               |
|                      |                                      | E3 (64, H, W)                        | 3×3 convolution, relu, bn                     |
| Final segmentation   | E3 (64, H, W)                        | P4 (K, H, W)                         |                                               |

The size of input image I and output map P is 3×H×W and K×H×W.

![Flowchart of class-aware enhanced module (CaEM)](image)

Fig. 3. Flowchart of class-aware enhanced module (CaEM), which consists of global class pooling (GCP) block and class relation enhanced (CRM) block. Technically, CaEM contains three steps. First, the GCP extracts class representation of feature map using segmentation result from last stage, then CRM measures the cosine similarity between each pixel and class representation to get the refined result, and reconstructs the enhanced feature map using the class representation and refined result. (a) Global class pooling (GCP). (b) Class relation measurement (CRM).

result. The detailed configuration of CCENet is captured in Table I.

It is worth noting that the embedded CaEM generates intermediate segmentation result for each stage, which can deliver the global class-aware prior information to refine the object boundary in a recursive manner from the deep layer to its following portion.

B. Class-Aware Enhanced Module (CaEM)

Inspired by self-attention mechanism [19], [20] and its variants [31], [34], we propose CaEM to introduce classwise information and boost the representational ability of the network. The detailed structure of CCENet is shown in Fig. 3. It consists of two subsequent blocks, global class pooling (GCP) block and class relation measurement (CRM) block. Given an input feature map \( E^t \in \mathbb{R}^{C\times H\times W} \) at stage \( t \) with channel number \( C \), height \( H \) and width \( W \), and the segmentation result \( P^t \in \mathbb{R}^{K\times H\times W} \) including \( K \) classes, GCP block can extract a set of class representation \( c^t \in \mathbb{R}^{K\times C} \). The similarity between feature map \( E^t \in \mathbb{R}^{C\times H\times W} \) and class representation \( c^t \in \mathbb{R}^{K\times C} \) is measured via CRM block, which not only gets the refined segmentation result \( P^{t+1} \in \mathbb{R}^{K\times H\times W} \) but also reconstructs the enhanced feature map \( E^{t+1} \in \mathbb{R}^{C\times H\times W} \).

1) Global Class Pooling (GCP): It has been shown in [12], [13], and [45] that global context features turn out to be advantageous in segmentation tasks, which can be easily achieved by global average pooling (GAP). However, GAP aggregates the feature of all pixels without considering categorical information, which could result in interclass indistinction problem. To improve GAP from categorical view, we propose a GCP operator which can generate global descriptors with class information. Therefore, GCP can provide a strong global class prior information for a specific image. As shown in Fig. 3(a), the inputs at stage \( t \) are the feature map \( E^t \in \mathbb{R}^{C\times H\times W} \) and class representation \( c^t \in \mathbb{R}^{K\times C} \) is computed by GCP operator as

\[
\begin{align*}
\text{Stage-1 feature:} & \quad E^i (C\times H\times W) \\
\text{Stage-2 result:} & \quad P^i (K\times H\times W) \\
\text{Softmax with temperature} & \quad \text{Stage-1 result:} \\
\text{Reshape & Transpose (N×C)} & \quad \text{Stage-2 result:} \\
\text{Stage-3 result:} & \quad P^i (K\times H\times W) \\
\text{Softmax with temperature} & \quad \text{Stage-3 result:} \\
\text{Reshape & Transpose} & \quad \text{Stage-3 result:} \\
\text{Stage-3 result:} & \quad P^i (K\times H\times W) \\
\text{Class representation} & \quad \text{Stage-3 result:} \\
\text{(a) Global Class Pooling (GCP)} & \quad \text{Stage-3 result:} \\
\text{Query} & \quad \text{Stage-3 result:} \\
\text{1×1 Conv} & \quad \text{Stage-3 result:} \\
\text{Similirarity matrix (N×K)} & \quad \text{Stage-3 result:} \\
\text{1×1 Conv} & \quad \text{Stage-3 result:} \\
\text{Value} & \quad \text{Stage-3 result:} \\n\text{Transpse & Reshape} & \quad \text{Stage-3 result:} \\
\text{Stage-3 result:} & \quad \text{Stage-3 result:} \\
\text{Transpse & Reshape} & \quad \text{Stage-3 result:} \\
\text{Stage-3 result:} & \quad \text{Stage-3 result:} \\
\end{align*}
\]
follows:

\[ c_k = \sum_{i=1}^{N} \frac{\exp(P_{ki}/T)}{\sum_{j=1}^{N} \exp(P_{kj}/T)} E_i, \quad k = 1, 2, \ldots, K \tag{2} \]

where \( E_i \in \mathbb{R}^{C \times 1} \) denotes the feature depiction of pixel \( i \), and \( P_{ki} \in [0, 1] \) denotes the confidence of pixel \( i \) belonging to class \( k \). To control the influence of other classes on specific category, we use a temperature \( T \) of softmax referring to knowledge distillation [46]. We note that GCP can be deducted into two special cases by adjusting the temperature \( T \). When \( T \to +\infty \), GCP degenerates as follows:

\[ c_k = \frac{1}{N} \sum_{i=1}^{N} E_i, \quad k = 1, 2, \ldots, K. \tag{3} \]

This has the same formulation to GAP. Equally, we can view that GAP is the extreme case of GCP with maximum of \( T \). Therefore, with smaller \( T \), GCP takes the class information into consideration compared with GAP, so that the irrelevant land cover can be excluded to learn more discriminative features. Another extreme situation is \( T \to 0^+ \); the class representation computed by GCP can be formulated as follows:

\[ c_k = \arg\max_i P_{ki}, \quad k = 1, 2, \ldots, K. \tag{4} \]

In this case, the class representation becomes the feature of most representative pixel [47]. However, only preserving one representative feature could lead to exceptional situation when some ground object categories do not exist in image. Instead, GCP with larger \( T \) aggregates global context features in the whole feature map in a soft form, and thereby GCP is easier to optimize.

2) Class Relation Measurement (CRM): To lead the input feature close to the class representation extracted by GCP block, we design CRM block to measure their similarity, which can enhance the representation capacity of network. As shown in Fig. 3(b), CRM inherits the structure of self-attention mechanism [20], [24]. Different from pixelwise matrix multiplication of self-attention mechanism, CRM introduces metric learning to measure the similarity between pixel feature and class representation. In particular, the reshaped feature map \( E_{\text{reshape}} \in \mathbb{R}^{N \times C} \) serves as the input of query branch, and class representation \( c \in \mathbb{R}^{K \times C} \) serves as the input of key branch and value branch, which can be formulated as follows:

\[ X^{\text{Query}} = W_Q E_{\text{reshape}} \tag{5} \]
\[ X^{\text{Key}} = W_K c^T \tag{6} \]
\[ X^{\text{Value}} = W_V c^T \tag{7} \]

where \( W_Q, W_K, W_V \in \mathbb{R}^{C \times C} \) are three linear transformations implemented by \( 1 \times 1 \) convolution, and \( X^{\text{Query}} \in \mathbb{R}^{N \times C}, X^{\text{Key}} \in \mathbb{R}^{C \times K}, X^{\text{Value}} \in \mathbb{R}^{C \times K} \) are the output of three branches. Next, \( X^{\text{Query}} \) and \( X^{\text{Key}} \) are used to conduct metric learning, which calculates the cosine similarity matrix \( S \in \mathbb{R}^{N \times K} \) as follows:

\[ S_{ij} = \frac{X_i^{\text{Query}} X_j^{\text{Key}}}{\|X_i^{\text{Query}}\|_2 \|X_j^{\text{Key}}\|_2} \tag{8} \]

where \( S_{ij} \in [0, 1] \) represents the cosine similarity between the \( i \)th pixel and the \( j \)th class representations. Different from pixelwise matrix multiplication of self-attention mechanism, e.g., DANet [22] and OCR [24], CRM treats pixel and class representation separately without interclass comparison. This improvement filters background interference during feature learning.

For stage \( t \), the similarity matrix can be obtained as \( S^t \) from the above processes, and the enhanced feature \( E'(t + 1) \) can be reconstructed by

\[ E^{t+1} = S^t (X^{\text{Value}})^t. \tag{9} \]

Finally, similarity matrix \( S^t \in \mathbb{R}^{N \times K} \) is transposed and reshaped into \( \mathbb{R}^{K \times H \times W} \) as the intermediate segmentation result \( P^t \) of stage \( t \), which is supervised by our proposed similarity loss. The enhanced feature \( E^{t+1} \in \mathbb{R}^{N \times C} \) is transposed and reshaped into \( \mathbb{R}^{C \times H \times W} \) as the output feature of CaEM.

C. Loss Function

Following the idea of deep supervision [34], [51], we combine all the losses on multiple stage results. Specifically, both initial (stage 0) and final prediction map (stage 4) are supervised with cross-entropy (CE) loss since they are obtained by softmax layer, which can be formulated as follows:

\[ L_{\text{init}} = \sum_{i=1}^{N} \sum_{k=1}^{K} \mathbb{I}(y_i = k) \log(P_{ki}^0) \tag{10} \]
\[ L_{\text{final}} = \sum_{i=1}^{N} \sum_{k=1}^{K} \mathbb{I}(y_i = k) \log(P_{ki}^4) \tag{11} \]

where \( P_{ki} \) denotes the output probability of pixel \( i \) belonging to class \( k \), and \( y_i \) is the groundtruth of pixel \( i \), and \( \mathbb{I}(\cdot) \) refers to the indicator function as illustrated in (11)

\[ \mathbb{I}(y_i = k) = \begin{cases} 1, & y_i = k \\ 0, & \text{otherwise} \end{cases}. \tag{12} \]

Different from segmentation result predicted by softmax, CaEM measures the cosine similarity between pixel and class representation. Therefore, it is inappropriate to supervise the intermediate results produced using CE loss, which would result in that all elements in similarity map close to 1. To avoid this situation, we propose to adopt the mean squared error loss in metrics learning [36] to constrain intermediate result during training, which can be formulated as follows:

\[ L_{\text{sim}}^{t} = \sum_{i=1}^{N} \sum_{k=1}^{K} (P_{i,k}^t - \mathbb{I}(y_i = k))^2 \tag{13} \]

Intuitively, if a pixel feature \( F_i \) has the same label as the class representation \( c_k \), the similarity \( S_{ik} \) should be close to 1. On the contrary, if \( F_i \) and \( c_k \) do not belong to the same category, the similarity \( S_{ik} \) should be close to 0. This ensures that pixels with
the different category labels obtain lower similarity. To sum up, the overall loss function can be formulated as follows:

$$
L = \lambda_{init} L_{init} + \sum_{t=1}^{3} \lambda_{sim} L_{sim}^{t} + \lambda_{final} L_{final}
$$

(14)

where hyperparameters $\lambda_{init}, \lambda_{sim},$ and $\lambda_{final}$ are used to balance the initial prediction loss $L_{init}$, similarity loss $L_{sim}$, and final prediction loss $L_{final}$.

D. Computation Analysis

It has been reported that the computational complexity of self-attention module is prohibitively tremendous [31], [34], which conflicts with the requirements for limited computational capabilities and real-time operations of an aerial platform. In our proposed CCENet, the similarity is used to depict the relationship between feature and class information; so it has much lower computation complexity even though we apply CaEM into low-level feature with high-resolution details. Specifically, previous self-attention based methods [20], [22], [23] have quadratic computation complexity $O(CN^2)$ to input image size due to computation of self-attention globally. In contrast, the proposed CaEM has linear computational complexity with respect to image size; since we only use class representation as the input of key pass and value pass instead of feature from all positions, the computational complexity and memory occupation of generating a similarity matrix $S$ is reduced from $O(CN^2)$ to $O(CNK)$ compared with original nonlocal block [20], where $K$ is the number of class, and $N = H \times W$ denotes the number of pixels in feature map; therefore, $K \ll N$ holds. For example, if the input image size is 256 $\times$ 256, the feature map size used in three-stage CaEM is 128 $\times$ 128 $\times$ 64, 64 $\times$ 64 $\times$ 256 and 32 $\times$ 32 $\times$ 2048, respectively. In the situation with class number 6, the CCENet spends merely 0.024 times of computation and memory for matrix multiplication compared with self-attention operation.

IV. EXPERIMENTS

To verify the effectiveness of our proposed CCENet, extensive experiments including ablation analysis and comparison experiments have been conducted using three publicly available high resolution aerial imagery datasets, i.e., ISPRS Vaihingen and Potsdam datasets, and iSAID dataset [48].

A. Datasets

1) Vaihingen: The ISPRS Vaihingen dataset contains 33 TOP tiles and DSMs with an average size of 2494 $\times$ 2064 and spatial resolution of 9 cm, including three spectral bands: near-infrared, red (R), and green (G) channels. Notably, 16 out of 33 tiles were fully annotated and used for the experiments, in which 11 images are used for training and the other 5 images (with image IDs: 11, 15, 28, 30, and 34) are for validation following [17], [23], [41], and [49]. Referring to [15], [17], and [38], we only selected the raw TOP images in this work to compare fairly and keep network general. The groundtruth labels contain six fully annotation classes: impervious surfaces, building, low vegetation, high vegetation, car, and clutter. The clutter category includes ground objects like water bodies, swimming pools, and containers. Following [23], [41], [49], and [50], we only predict five classes in the case of Vaihingen, ignoring the clutter class, due to the lack of training data (less than 1%) for that class.

2) Potsdam: The ISPRS Vaihingen dataset consists of 38 TOP tiles and DSMs with size of 6000 $\times$ 6000 and spatial resolution of 5 cm. Different from Vaihingen, these TOP files contain four bands: infrared (IR), red (R), green (G), and blue (B). Following [23], [41], and [50], we only focus on the IR-R-G channel data in our work, and 24 of 38 images are carefully annotated; so we use these 24 images in our experiments. Seven TOP tiles (with image IDs: 2_11, 2_12, 4_10, 5_11, 6_7, 7_8, 7_10) are selected as validation data while the other 17 annotated images comprise the training data. Different from Vaihingen dataset, we predict all six classes from Potsdam dataset.

3) iSAID: The iSAID [48] dataset is the largest instance segmentation dataset which consist of 2806 high spatial resolution RS images. Concretely, the training, validation, and test set contains 1411, 458, and 937 images respectively. Besides instance-level annotation, iSAID also provides semantic mask annotation for segmentation, which is used for our experiment. The images of iSAID are captured from multiple sensors and platforms with multiple resolutions. The original image scale ranges from 800 $\times$ 800 to 4000 $\times$ 13 000 pixels. The iSAID dataset contains 15 categories, i.e., ship (Ship), storage tank (ST), baseball diamond (BD), tennis court (TC), basketball court (BC), ground field track (GTF), bridge (Bridge), large vehicle (LV), small vehicle (SV), helicopter (HC), swimming pool (SP), roundabout (RA), soccerball field (SBF), plane (Plane), and harbor (Harbor). In our experiment, we only use the training and validation set for training and evaluating, as the groundtruth of test set is unavailable.

B. Evaluation Metrics

To assess the quantitative performance, two metrics are employed in our experiments according to the ISPRS dataset guidelines, including the overall pixel accuracy (OA) and the mean F1 (mF1) following [17], [23], [41], and [49]

$$
OA = \frac{TP + TN}{TP + FN + FP + TN + FN}
$$

(15)

$$
mF1 = \frac{1}{N} \sum_{k=1}^{N} \frac{2TP_k}{2TP_k + FP_k + FN_k}
$$

(16)

where TP, FP, TN, and FN are the number of true positive, false positive, true negative, and false negative, respectively. For iSAID dataset, we use the mean intersection over union (mIoU) as metric following [51] and [52]

$$
mIoU = \frac{1}{N} \sum_{k=1}^{N} \frac{TP_k}{TP_k + FP_k + FN_k}.
$$

(17)

C. Implementation Details

The implementation is based on Pytorch1.7, and all the experiments are conducted on a server with eight NVIDIA GeForce
RTX 3090 (24 GB). The size of input image is cropped to (512, 512) for Vaihingen and Potsdam during training. For iSAID dataset, we crop the image into (896, 896) following [51] and [52]. To avoid overfitting during training, two data augmentation methods are considered in this article. On the one hand, a random sliding cropping strategy is employed instead of using fixed sliding window to crop the labeled tiles, which can generate new training images in a random position to expand the number of training set. On the other hand, we randomly transform training images by flipping (horizontal and vertical or both) with a probability of 0.5.

For training details, we adopt stochastic gradient descent algorithm as the optimizer with an initial learning rate of 0.01, a momentum of 0.9, and a weight decay of 0.0005. Each training process contains 50 epochs in total, and we use multistep learning schedule with learning rate decaying by the factor of 0.1 in 20th, 30th, and 40th epochs. In addition, due to the GPU memory capacity, the batch size is set to 24 for the Vaihingen and Potsdam datasets and set to 16 for iSAID according to the different sizes of input patches for the two datasets. As for inference, sliding windows with 50% overlap is utilized to traverse the whole images to achieve seamless prediction results, and then multiple predictions are averaged to get the final prediction.

D. Ablation Study

In this subsection, we conduct a series of experiments to reveal the effect of each component in our proposed module. Notably, we first use the dilated ResNet-101 as the baseline, and final segmentation result is achieved by directly upsampling the output. To avoid the influence of the training deviation of each experiment, we conduct five experiments for each group of the settings and calculate the average value and standard deviation as the final results.

1) Impact of Temperature in GCP: As proved in Section III-B1, the temperature \( T \) in GCP block controls the influence of other categories on a category. To investigate the impact of temperature \( T \) in GCP block, we change it from 0.1 to 10, and the corresponding accuracy results are recorded in Table II. Specially, when \( T \) is set to 0.5, our model achieves the best performance with 90.09% in overall accuracy. As \( T \) increases to 10, the OA drops to 88.93%. This could be because more irrelevant land cover is introduced to class representation, which leads to less discriminative features for segmentation. When \( T \) decreases to 0.1, the overall accuracy descends to 88.61%. The reason could be that GCP only preserving few representative features and filter too much global information, which causes some exceptional situation and brings difficulty to the optimizer when some ground object categories do not exist in image. Therefore, the temperature \( T \) should be set neither too large nor too small, and then we set \( T \) to 0.5 in the later experiments.

2) Impact of Different Loss and Hyperparameters: As mentioned in Section III-C, the proposed CCENet consists of three parts of loss at training stage, i.e., initial prediction loss \( \mathcal{L}_{\text{init}} \), similarity loss \( \mathcal{L}_{\text{sim}} \), and final prediction loss \( \mathcal{L}_{\text{final}} \), which are controlled by hyperparameters \( \lambda_{\text{init}}, \lambda_{\text{sim}}, \) and \( \lambda_{\text{final}} \). Here, we fix the final loss \( \lambda_{\text{final}} \) to 1 and conduct extension experiments to analyze the influence of weight parameters \( \lambda_{\text{init}} \) and \( \lambda_{\text{sim}} \) on the segmentation results separately, i.e., setting the other parameter to 0. As shown in Tables III and IV, it can be observed that the model is not sensitive to the change of hyperparameter. Moreover, the best results are achieved under the setting of \( \lambda_{\text{init}} = 0.4 \) and \( \lambda_{\text{sim}} = 0.6 \). Based on the above observation, we choose the optimal settings in the following experiments.

To further evaluate the impact of the auxiliary supervision on initial and intermediate results, we perform experiments using different loss setting. As shown in Table V, it can be observed that both auxiliary loss functions \( \mathcal{L}_{\text{sim}} \) and \( \mathcal{L}_{\text{init}} \) have certain improvement effects on model training optimization. Notably, if neither \( \mathcal{L}_{\text{sim}} \) nor \( \mathcal{L}_{\text{init}} \) is applied, our CCENet does not consider class information and degenerated to EMANet [29] where the number of basis is equal to the number of class. It can only achieve 89.04% in mean \( F1 \) score and 89.25% in overall accuracy. When we only add supervision \( \mathcal{L}_{\text{sim}} \), the labeling result merely increases to 89.27% in mean \( F1 \) score and 89.44% in overall accuracy, due to the lack of the guidance of initial segmentation. While the supervision \( \mathcal{L}_{\text{init}} \) leads to notable improvement to 89.63% in mean \( F1 \) score and 89.78% in overall accuracy, which indicates that class-aware prior information is beneficial for segmentation. When both \( \mathcal{L}_{\text{init}} \) and \( \mathcal{L}_{\text{sim}} \) are applied, our model achieves the best performance with 89.91% in mean \( F1 \) score and 90.09% in OA. This proves that the proposed similarity

| TABLE II |
|-----------|
| **ABALATION STUDY OF TEMPERATURE IN GCP** |
| \( T \) | 0.1 | 0.5 | 1 | 5 | 10 |
| mF1 (%) | 88.24 | 89.91 | 89.46 | 88.53 | 88.45 |
| OA (%) | 88.61 | 90.09 | 89.83 | 89.07 | 88.93 |

| TABLE III |
|-----------|
| **ABALATION STUDY OF DIFFERENT HYPERPARAMETER \( \lambda_{\text{init}} \)** |
| \( \lambda_{\text{init}} \) | 0.2 | 0.3 | 0.4 | 0.5 | 0.6 |
| mF1 (%) | 89.35 | 89.43 | 89.63 | 89.56 | 89.42 |
| OA (%) | 89.43 | 89.58 | 89.78 | 89.67 | 89.55 |

| TABLE IV |
|-----------|
| **ABALATION STUDY OF DIFFERENT HYPERPARAMETER \( \lambda_{\text{sim}} \)** |
| \( \lambda_{\text{init}} \) | 0.4 | 0.5 | 0.6 | 0.7 | 0.8 |
| mF1 (%) | 89.13 | 89.23 | 89.27 | 89.19 | 89.08 |
| OA (%) | 89.27 | 89.36 | 89.44 | 89.30 | 89.23 |

| TABLE V |
|-----------|
| **ABALATION STUDY OF LOSS FUNCTIONS ON THE VAIHINGEN DATASETS** |
| \( \mathcal{L}_{\text{init}} \), \( \mathcal{L}_{\text{sim}} \), | mF1(%) | OA(%) |
| \( \times \times \) | 89.04 ± 0.15 | 89.25 ± 0.19 |
| \( \times \checkmark \) | 89.27 ± 0.26 | 89.44 ± 0.29 |
| \( \checkmark \times \) | 89.63 ± 0.12 | 89.78 ± 0.14 |
| \( \checkmark \checkmark \) | 89.91 ± 0.14 | 90.09 ± 0.15 |
supervision \( L_{\text{sim}} \) improves the separability of ground objects in feature space when the class prior information is provided, thus boosting the labeling results. Based on the above comparative analysis, both \( L_{\text{sim}} \) and \( L_{\text{un}} \) are adopted in our method.

3) Effectiveness of CaEM: To demonstrate the advantage of our proposed CaEM, we design comparison experiments on dilated ResNet-101 baseline, its improved model with self-attention SRM [28] (ResNet-101 + SRM), a single CaEM (ResNet-101 + single CaEM), and cascade CaEM (ResNet-101 + cascade CaEM) in terms of performance and efficiency. For fair comparison, we first only apply a single CaEM in decoder and simply output the result by bilinear interpolation, which is the same as self-attention SRM [28]. The efficiency indexes, i.e., GPU Memory, computation complexity (measured by the number of FLOPs), and inference time, are obtained on a single GPU with the high-resolution aerial image size of \( 1 \times 3 \times 1024 \times 1024 \) and the indexes are the smaller the better. Notably, the results contain all the computational overhead of the entire model, which is different from theoretical analysis in Section III-D with only matrix multiplication. From the experiment results in Table VI, it can be observed that ResNet-101 + SRM can bring 1.22% and 0.97% improvement in mean F1 and OA over dilated ResNet-101 baseline, respectively, because of considering the global attention information. Introducing attention is a two-edged sword, which introduces about two times additional memory (+1919.83 MB), one time additional computation complexity (+813.94 G Flops), and 1/5 extra inference time (+1.97 ms) compared to baseline. Similarly, the ResNet-101 + single CaEM improves the baseline performance to 89.56% (1.54%) on mean F1 score and 89.68 (1.34%) on overall accuracy, and it surpasses SRM 0.32% in mean F1 score and 0.37% in OA, respectively. However, it hardly brings additional computation compared with baseline. Furthermore, for our ResNet-101 + cascade CaEM, it brings about three times more resources than ResNet-101 + single CaEM, but it still spends much less resource and running time than self-attention SRM, with only additional 497.34 MB memory, 362.61 G Flops, and 1.42 ms inference time. As for the performance, ResNet-101 + cascade CaEM dramatically exceeds 0.65% in mean F1 score and 0.76% in OA compared with SRM, which demonstrates the superiority of our method in labeling accuracy and computation efficiency.

To better understand the superiority of CaEM, we further conduct feature visualization. As shown in Fig. 1, given a pixel belonging to building [pixel marked by yellow cross sign in Fig. 1(a)], we visualize the feature cosine similarity of RAFCN equipped with self-attention SRM [23] and CCENet equipped with CaEM. It can be seen that the feature of building and impervious surface are indistinguishable in RAFCN, while our method can exclude the interference of irrelevant land covers for learning more discriminative features. Moreover, we visualize the original feature extracted by dilated ResNet-101 and class-aware enhanced feature aggregated by proposed CaEM using TSNE [53]. As shown in Fig. 4, after the process of CaEM, pixels of different classes gather together and come into several groups. Also, we can observe that the intraclass features are more consistent and the interclass features are more distinguishable, which alleviates the interclass homogeneity and intraclass heterogeneity of ground objects in feature space. This is mainly because the class representation introduces class-aware global prior information for a specific image.

4) Intermediate Results of Cascade CaEM: In order to validate the effectiveness of our proposed cascade strategy, we output the quantitative results of intermediate results. As shown in Table VII, our cascade architecture can bring stable improvement compared with initial result, which benefits from object boundary refinement with class prior information from deep layer. Another interesting phenomenon is that the previous stage brings larger improvement, where stage 1 brings largest improvement by 0.72% in OA and 0.84% in mean F1 score, while the final convolution layer only boosts 0.22% in OA and 0.31% in mean F1 score. This might be because the class-aware global information is first introduced into model, while the latter stage with high resolution feature map plays a role of refinement.

### E. Comparisons With Other Methods

To evaluate the CCENet, nine typical networks are selected as comparison methods, including two classical methods (SegNet [25] and MLP [49]), two representative methods (Deeplabv3+[13] and HR-Net [27]), and five similar state-of-the-art methods (RAFCN [23], DANet [22], SSAtNet [40], ACFNet [31], and OCRNet [24]). For comprehensive evaluation, we report computation complexity (measured by the number of FLOPs), quantitative score, and qualitative segmentation visualization.

1) Comparisons on the ISPRS Vaihingen: Numerical results on the Vaihingen dataset are shown in Table VIII, and the per-class F1 scores are also enumerated to assess the performance of...
recognizing different objects. Formally, † in the table indicates that reported scores come from original paper. It can be observed that CCENet outperforms other methods in terms of both mean F1 score and OA. Specifically, though classic MLP [49] have extra DSMs as input, our CCENet outperforms them by a large margin. Moreover, comparisons with widely used Deeplabv3+ [13] validate the effectiveness of the class-level global information, where CCENet contributes to increments of 3.52% and 1.22% in mean F1 score and OA, respectively. Compared to RAFCN [23] and SSAatNet [40], CCENet reaches rise of 1.37% and 1.04% in mean F1 score, 0.86% and 0.73% in OA, which indicates the superiorities of class-level information aggregation over traditional attention-based schemes. In comparison with OCRNet, our model achieves increment of 0.5% and 0.45% in mean F1 score and OA, which demonstrates the effectiveness of cascade object boundary refinement architecture in semantic segmentation of high-resolution aerial images. It is worth noting that our method possesses strong capacity to distinguish confusing categories pair like impervious surface and building, tree, and low vegetation; so it outperforms the others in terms of per-class F1 by a large margin. As for the efficiency, although cascade refinement framework introduces additional computation complexity, CCENet still has much lower computation complexity than self-attention based network, while achieving best results.

To achieve intuitive comparison, samples of segmentation results of SegNet, HRNet, DANet, and the proposed CCENet are depicted in Fig. 5. The first row demonstrates that networks without global class-level information fail to recognize buildings with different facade. Whereas, the proposed CCENet makes relatively accurate predictions. This indicates that our model alleviates the problem of high intraclass heterogeneity in aerial images. Besides, examples in the second row illustrate that CCENet performs better than other methods when land covers have similar visual appearance such as low vegetation and tree. This illustrates that our model has stronger ability to distinguish ground object with interclass homogeneity. To summarize, though the complex scenes contain ambiguous objects, the qualitative comparisons show that our CCENet provides more accurate labeling than other methods due to the introduction of CaEM.

2) Comparisons on the ISPRS Potsdam: In order to further validate the effectiveness of our network, we conduct experiments on the Potsdam dataset, and numerical results are reported in Table IX. It can be seen that our model outperforms the other nine models by a considerable margin on both metrics, achieving...
Fig. 5. Qualitative visualizations of segmentation results on image patches of the ISPRS Vaihingen dataset. Classes: impervious surface (white), building (blue), low vegetation (cyan), and tree (green).

| Method       | Imp surf (%) | building (%) | low vegetation (%) | Tree (%) | car (%) | clutter (%) | mF1 (%) | OA (%) | Whether to use DSM |
|--------------|--------------|--------------|--------------------|----------|---------|-------------|---------|--------|-------------------|
| SegNet [25]  | 89.88        | 92.21        | 85.10              | 83.38    | 95.59   | 72.87       | 86.51   | 86.90  | No                |
| MLP [49]     | 89.31        | 94.37        | 84.83              | 81.10    | 93.56   | 76.54       | 86.62   | 87.02  | Yes               |
| DeepLabv3+ [13] | 91.53      | 94.45        | 86.75              | 83.38    | 95.05   | 75.98       | 87.86   | 88.45  | No                |
| HR-Net [27]  | 90.79        | 93.71        | 86.39              | 83.17    | 95.64   | 76.86       | 87.76   | 88.09  | No                |
| RAFCN [23]   | 91.33        | 94.70        | 86.81              | 83.47    | 94.52   | 77.27       | 88.01   | 88.59  | Yes               |
| DANet [22]   | 91.16        | 94.67        | 86.93              | 85.36    | 94.89   | 77.51       | 88.42   | 88.83  | No                |
| SSA-Net [37] | 91.48        | 93.73        | 85.97              | 83.95    | 95.34   | 76.52       | 87.83   | 88.24  | No                |
| ACFNet [31]  | 91.67        | 94.38        | 87.38              | 84.53    | 94.73   | 78.13       | 88.47   | 88.96  | No                |
| OCRNet [35]  | 91.55        | 94.47        | 87.12              | 84.72    | 95.45   | 78.38       | 88.62   | 88.91  | No                |
| Ours         | 91.72        | 94.73        | 87.67              | 84.54    | 95.09   | 80.17       | 88.98   | 89.27  | No                |

88.98% and 89.27% mean F1 score and OA, respectively. To be specific, our CCENet achieves improvement of 0.97% and 0.56% in mean F1 score, and 0.68% and 0.44% in OA with respect to competitive RAFCN [23] and DANet [22]. In contrast with the recently proposed SSAtNet [40] and OCRNet [24], our CCENet achieves 1.15% and 0.36% improvement in mF1 score, which verifies the effectiveness of the proposed CaEM and cascade refinement design. As for per-class F1, though our model ranks second in categories of tree and car, the segmentation results of CCENet are still very close to the best values. Notably, CCENet remarkably surpasses other competitors by a large margin in recognizing the category of clutter, which is challenging owing to its various occurrence and complicated textures. This indicates that leveraging the classwise contextual dependencies can learn more discriminative feature, yielding a better classification accuracy.

Furthermore, visual results are presented in Fig. 6. As shown in the first row, albeit cars sheltered by intricate surroundings, the boundaries between areas of trees and car in the segmentation result tend to be more precise and smoother for the case of CCENet. In the second row, the aerial scene is complicated, where the buildings, cars, clutters, and roads show interclass homogeneity. This brings enormous challenge for other competitors, while our model enhances the classwise semantic features to distinguish the similar objects of different semantic categories.

3) Comparisons on the iSAID Dataset: To evaluate the importance of the proposed CaEM and cascade framework, we further conduct experiment on iSAID dataset with more category number. As shown in Table X, we report per-class IOU and mIOU for comprehensive comparison. It can be observed that our model achieved state-of-the-art performance. Specifically, our model surpasses SegNet [25] and HRNet [27] by a large margin. This is mainly because SegNet and HRNet only consist of branches of traditional convolution layer with limited respective field, which fails to extract context information effectively. In contrast, though Deeplabv3+ [13], SSAtNet [40], and DANet [22] increase respective field by additional contextual aggregation module, our model still reaches better result. This phenomenon shows the importance of class-aware prior information for semantic segmentation. Moreover, the proposed CCENet achieves 4.14% and 1.74% improvement compared with ACFNet [31] and OCRNet [24]. This attributes to our...
explicit supervision on the intermediate result and cascade refinement structure. Moreover, we additionally report the state-of-the-art result of FarSeg [51] for competitive comparison. The proposed CCENet outperforms 13 categories in $F_1$ score and achieves 1.01% higher mIOU in contrast with FarSeg [51].

We further visualize the segmentation result in Fig. 7. In the first row, our model can precisely distinguish background (black), ground tracker field (blue), and soccer ball field (light blue), while the comparison models cannot accurately delineate the edges of ground tracker field with the help of cascade refinement. In the second row, the comparison methods mistake large vehicle (dark blue) for small vehicle, while our model tackles the interclass homogeneity problem. This shows the importance

| Method          | Slip | ST  | BD  | TC  | BC  | GTF | Bridge | LV  | SV  | HC  | SP  | RA  | SBP | Plane | Hbar |
|-----------------|------|-----|-----|-----|-----|-----|--------|-----|-----|-----|-----|-----|-----|-------|------|
| SegNet [25]     | 58.82| 64.11| 33.55| 81.78| 52.76| 18.63| 20.36  | 53.93| 41.26| 0   | 32.05| 42.59| 51.1 | 72.53 | 48.80| 44.62|
| Deeplabv3+ [13]| 59.02| 55.15| 75.94| 84.18| 58.52| 59.24| 32.11  | 54.54| 33.79| 31.14| 44.24| 67.51| 73.78| 75.70 | 45.76| 59.33|
| iSR-Net [27]   | 63.91| 69.11| 67.83| 85.91| 49.08| 43.59| 36.02  | 60.68| 36.70| 24.45| 46.77| 33.06| 59.58| 80.41 | 53.95| 54.07|
| DaNet [22]     | 66.71| 74.13| 77.92| 86.82| 52.45| 56.14| 42.80  | 63.71| 48.63| 31.83| 46.85| 59.36| 73.54| 83.74 | 55.77| 61.36|
| SSSNet [40]    | 67.25| 74.87| 73.67| 87.44| 47.46| 49.75| 41.48  | 62.11| 48.48| 18.06| 47.38| 71.89| 72.66| 81.28 | 57.75| 60.10|
| ACFNet [31]    | 68.10| 73.79| 66.01| 88.01| 61.96| 54.68| 39.68  | 63.86| 48.47| 18.88| 47.23| 61.17| 76.16| 82.02 | 58.44| 60.56|
| Fansseg [51]   | 65.38| 61.80| 77.73| 86.35| 62.08| 56.70| 36.70  | 60.59| 46.34| 33.82| 51.21| 71.35| 72.53| 82.03 | 53.91| 63.71|
| OCRNet [35]    | 67.58| 74.97| 78.49| 88.45| 57.40| 59.55| 45.33  | 63.35| 49.17| 33.67| 46.36| 66.65| 72.27| 84.09 | 55.02| 62.96|
| Ours           | 71.96| 74.29| 79.21| 88.85| 64.71| 59.54| 44.70  | 65.54| 51.07| 37.29| 45.54| 69.33| 77.89| 85.37 | 55.23| 64.70|

Fig. 6. Qualitative visualizations of segmentation results on image patches of the ISPRS Potsdam dataset. Classes: Impervious surface (white), building (blue), low vegetation (cyan), tree (green), car (yellow), and clutter (red).

Fig. 7. Qualitative visualizations of segmentation results on image patches of the iSAID dataset.
of exploiting class-aware prior information to moderate the problem of interclass homogeneity.

V. CONCLUSION

In this article, a novel coarse-to-fine CCENet is proposed for the semantic segmentation of high-resolution aerial images. In this framework, classwise global context information is effectively exploited by the proposed CaEM, which alleviates the issue of interclass homogeneity and intraclass heterogeneity. With the help of CaEM, CCENet refines object boundary results using class-aware prior information. The ablation studies and comparative experiment results verify the superiorities of our proposed framework in terms of high labeling accuracy and computation efficiency. On three publicly available datasets, both quantitative and qualitative results show that our CCENet provides more precise labeling results than other contrasting well-known algorithms. In future, we will consider the idea of metrics learning among class representations into domain adaptation to narrow the domain gap between the two ISPRS dataset, thus solving the domain shift problem.

REFERENCES

[1] M. Maboudi, J. Amini, S. Malih, and M. Hahn, “Integrating fuzzy object based image analysis and ant colony optimization for road extraction from remotely sensed images,” ISPRS J. Photogrammetry Remote Sens., vol. 138, pp. 151–163, 2018.

[2] A. Puri, K. Valavanis, and M. Konttis, “Statistical profile generation for traffic monitoring using real-time UAV based video data,” in Proc. Mediterranean Conf. Control Automat., 2007, pp. 1–6.

[3] M. Volpi and D. Tuia, “Dense semantic labeling of subdecimeter resolution images with convolutional neural networks,” IEEE Trans. Geosci. Remote Sens., vol. 55, no. 2, pp. 881–893, Feb. 2017.

[4] K. Nogueira, W. R. Schwartz, and J. A. d. Santos, “Coffee crop recognition using multi-scale convolutional neural networks,” in Proc. Iberoamerican Congr. Pattern Recognit., 2015, pp. 67–74.

[5] A. Ahmed, M. Nagai, C. Tienan, and R. Shibaakis, “UAV based monitoring system and object detection technique development for a disaster area,” Int. Arch. Photogrammetry Remote Sens. Spatial Inf. Sci., vol. 37, pp. 373–377, 2008.

[6] G. Zhang, T. Lei, Y. Cui, and P. Jiang, “A dual-path and lightweight convolutional neural network for high-resolution aerial image segmentation,” ISPRS Int. J. Geo-Inf., vol. 8, no. 12, 2019, Art. no. 582.

[7] F. Wang, S. Piao, and J. Xie, “CSE-HRNet: A context and semantic enhanced high-resolution network for semantic segmentation of aerial imagery,” IEEE Access, vol. 8, pp. 182475–182489, 2020.

[8] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet classification with deep convolutional neural networks,” in Proc. Adv. Neural Inf. Process. Syst., vol. 25, 2012, pp. 1097–1105.

[9] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2016, pp. 770–778.

[10] J. Long, E. Shelhamer, and T. Darrell, “Fully convolutional networks for semantic segmentation,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2015, pp. 3431–3440.

[11] P. Tokarczyk, J. D. Wegner, S. Walk, and K. Schindler, “Features, color spaces, and boosting: New insights on semantic classification of remote sensing images,” IEEE Trans. Geosci. Remote Sens., vol. 53, no. 1, pp. 280–296, Jan. 2015.

[12] C. Peng, X. Zhang, G. Yu, G. Luo, and J. Sun, “Large kernel matters-improve semantic segmentation by global convolutional network,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2017, pp. 4353–4361.

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

[14] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, “Pyramid scene parsing network,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2017, pp. 2881–2890.

[15] X. Chen et al., “Adaptive effective receptive field convolution for semantic segmentation of VHR remote sensing images,” IEEE Trans. Geosci. Remote Sens., vol. 59, no. 4, pp. 3532–3546, Apr. 2021.

[16] N. Audebert, B. Le Saux, and S. Lefèvre, “Beyond RGB: Very high resolution urban remote sensing with multimodal deep networks,” ISPRS J. Photogrammetry Remote Sens., vol. 140, pp. 20–32, 2018.

[17] Y. Liu, B. Fan, L. Wang, J. Bai, S. Xiang, and C. Pan, “Semantic labeling in very high resolution images via a self-cascaded convolutional neural network,” ISPRS J. Photogrammetry Remote Sens., vol. 145, pp. 78–95, 2018.

[18] K. Nogueira, M. Dalla Mura, J. Chanussot, W. R. Schwartz, and J. A. Dos Santos, “Dynamic multicontext segmentation of remote sensing images based on convolutional networks,” IEEE Trans. Geosci. Remote Sens., vol. 57, no. 10, pp. 7503–7520, Oct. 2019.

[19] R. Li et al., “Multiattention network for semantic segmentation of fine-resolution remote sensing images,” IEEE Trans. Geosci. Remote Sens., vol. 60, Jul. 2022, Art. no. 5607713.

[20] X. Wang, R. Girshick, A. Gupta, and K. He, “Non-local neural networks,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2018, pp. 7794–7803.

[21] Y. Yuan, L. Huang, J. Guo, C. Zhang, X. Chen, and J. Wang, “Ocnet: Object context for semantic segmentation,” Int. J. Comput. Vis., vol. 129, no. 8, pp. 2375–2398, 2021, arXiv:1809.00916.

[22] J. Fu et al., “Dual attention network for scene segmentation,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2019, pp. 3146–3154.

[23] L. Mou, Y. Hua, and X. X. Zhu, “Relation matters: Relational context-aware fully convolutional network for semantic segmentation of high-resolution aerial images,” IEEE Trans. Geosci. Remote Sens., vol. 58, no. 11, pp. 7557–7569, Nov. 2020.

[24] Y. Yuan, X. Chen, and J. Wang, “Object-contextual representations for semantic segmentation,” in Proc. Eur. Conf. Comput. Vis., 2020, pp. 173–190.

[25] V. Badrinarayanan, A. Kendall, and R. Cipolla, “SEGNet: A deep convolutional encoder-decoder architecture for image segmentation,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 39, no. 12, pp. 2481–2495, Dec. 2017.

[26] O. Ronneberger, P. Fischer, and T. Brox, “U-Net: Convolutional networks for biomedical image segmentation,” in Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Interv., 2015, pp. 234–241.

[27] K. Sun et al., “Deep high-resolution representation learning for visual recognition,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 43, no. 10, pp. 3349–3364, 2020, arXiv:1904.04514.

[28] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, “Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 40, no. 4, pp. 834–848, Apr. 2018.

[29] G. Lin, A. Milan, C. Shen, and I. Reid, “RefineNet: Multi-path refinement networks for high-resolution semantic segmentation,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2017, pp. 1925–1934.

[30] H. K. Cheng, J. Chung, Y.-W. Tai, and C.-K. Tang, “CascadePSP: Toward class-agnostic and very high-resolution segmentation via global and local refinement,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2020, pp. 8890–8899.

[31] F. Zhang et al., “ACFNet: Attentional class feature network for semantic segmentation,” in Proc. IEEE/CVF Int. Conf. Comput. Vis., 2019, pp. 6798–6807.

[32] C. Li, M. Zhu, Q.-H. Tran, X. Yu, G. D. Hager, and M. Chandraker, “Deep supervision with intermediate concepts,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 41, no. 8, pp. 1828–1843, Aug. 2019.

[33] J. He, Z. Deng, and Y. Qiao, “Dynamic multi-scale filters for semantic segmentation,” in Proc. IEEE/CVF Int. Conf. Comput. Vis., 2019, pp. 3562–3572.

[34] X. Li, Z. Zhong, J. Wu, Y. Yang, Z. Lin, and H. Liu, “Expectation-maximization attention networks for semantic segmentation,” in Proc. IEEE/CVF Int. Conf. Comput. Vis., 2019, pp. 9167–9176.

[35] Z. Zhu, M. Xu, S. Bai, T. Huang, and X. Bai, “Asymmetric non-local neural networks for semantic segmentation,” in Proc. IEEE/CVF Int. Conf. Comput. Vis., 2019, pp. 593–602.

[36] F. Sung, Y. Yang, L. Zhang, T. Xiang, P. H. Torr, and T. M. Hospedales, “Learning to compare: Relation network for few-shot learning,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2018, pp. 1199–1208.
Qixiong Wang (Student Member, IEEE) received the B.S. degree in control engineering from Beihang University, Beijing, China, in 2019. He is currently working toward the Ph.D. degree in pattern recognition with the School of Astronautics, Beihang University, Beijing, China. His research interests include image semantic segmentation, object detection, deep learning, and hyperspectral image classification.

Xiaoyan Luo (Member, IEEE) received the B.S. degree in communication engineering from the Taiyuan University of Technology, Shanxi, China, in 2004, and the M.S. degree in communication engineering and Ph.D. degree in control engineering from Beihang University, Beijing, China, in 2007 and 2012, respectively. She is currently an Associate Professor with Image Processing Center, Beihang University. Her research interests include hyperspectral image processing, target detection, image fusion, and pattern recognition.

Jiaqi Feng received the B.S. degree in control engineering from Beihang University, Beijing, China, in 2020, where he is currently working toward the Ph.D. degree in pattern recognition with the School of Astronautics. His research interests include remote sensing image processing and deep learning.

Sen Li received the B.S. degree in control engineering from Beihang University, Beijing, China, in 2016, where he is currently working toward the Ph.D. degree in pattern recognition with the School of Astronautics. His research interests include hyperspectral image classification, object detection, and deep learning.

Jihao Yin (Senior Member, IEEE) received the Ph.D. degree in computer science from the Beijing Institute of Technology, Beijing, China, in 2007. He joined Beihang University, Beijing, China, where he is currently a Full Professor with the Image Processing Center, School of Astronautics. His research interests include image processing and remote sensing applications. As a Principal Investigator, his research works have been supported by several key research funds in China including the National Natural Science Fund of China, National High Technology Research and Development Program of China, Research Fund for the Doctoral Program of Higher Education of China, etc.