All Grains, One Scheme (AGOS): Learning Multigrain Instance Representation for Aerial Scene Classification

Qi Bi, *Student Member, IEEE*, Beichen Zhou, Kun Qin, Qinghao Ye, and Gui-Song Xia, *Senior Member, IEEE*

***Abstract*— Aerial scene classification remains challenging as: 1) the size of key objects in determining the scene scheme varies greatly and 2) many objects irrelevant to the scene scheme are often flooded in the image. Hence, how to effectively perceive the region of interests (RoIs) from a variety of sizes and build more discriminative representation from such complicated object distribution is vital to understand an aerial scene. In this article, we propose a novel all grains, one scheme (AGOS) framework to tackle these challenges. *To the best of our knowledge*, it is the first work to extend the classic multiple instance learning (MIL) into multigrain formulation. Specifically, it consists of a multigrain perception (MGP) module, a multibranch multi-instance representation (MBMIR) module, and a self-aligned semantic fusion (SSF) module. First, our MGP module preserves the differential dilated convolutional features from the backbone, which magnifies the discriminative information from multigrains. Then, our MBMIR module highlights the key instances in the multigrain representation under the MIL formulation. Finally, our SSF module allows our framework to learn the same scene scheme from multigrain instance representations and fuses them, so that the entire framework is optimized as a whole. Notably, our AGOS is flexible and can be easily adapted to existing convolutional neural networks (CNNs) in a plug-and-play manner. Extensive experiments on UCM, aerial image dataset (AID), and Northwestern Polytechnical University (NWPU) benchmarks demonstrate that our AGOS achieves a comparable performance against the state-of-the-art methods.**

***Index Terms*— Aerial scene classification, differential dilated convolution (DDC), multigrain instance representation, multiple instance learning (MIL), self-alignment strategy.**

Manuscript received 27 April 2022; revised 23 July 2022; accepted 23 August 2022. Date of publication 25 August 2022; date of current version 12 September 2022. This work was supported in part by the National Natural Science Foundation of China under Contract U2033216, Contract U1833201, Contract 61922065, Contract 61771350, and Contract 41820104006; in part by the Foundation from Key Laboratory of National Geographic Census and Monitoring, Ministry of Nature Resources under Contract 2020NGCMZD03; in part by the Science and Technology Major Project of Hubei Province (Next-Generation AI Technologies) under Grant 2019AEA170; and in part by the Shanghai Aerospace Science and Technology Innovation Project under Grant SAST2019-094. *(Corresponding authors: Kun Qin; Gui-Song Xia.)*

Qi Bi, Beichen Zhou, and Kun Qin are with the School of Remote Sensing and Information Engineering, Wuhan University, Wuhan 430072, China (e-mail: qink@whu.edu.cn).

Qinghao Ye is with the Department of Computer Science and Engineering, University of California, San Diego, CA 92093 USA.

Gui-Song Xia is with the State Key Laboratory of LIESMARS, National Engineering Research Center for Multimedia Software, School of Computer Science, Institute of Artificial Intelligence, Wuhan University, Wuhan 430072, China (e-mail: guisong.xia@whu.edu.cn).

Digital Object Identifier 10.1109/TGRS.2022.3201755

I. Introduction

AERIAL scene classification stands at the crossroad of image processing and remote sensing, and has drawn increasing attention in the computer vision community in the past few years [1], [2], [3], [4], [5]. Moreover, aerial scene classification is a fundamental task toward the understanding of aerial images, as it plays a significant role on many aerial image applications such as land use classification [6], [7], [8] and urban planning [9].

A. Problem Statement

Despite the great performance gain led by deep learning for image recognition [10], [11], [12], [13], [14], [15], aerial scene classification remains challenging due to some unique characteristics.

1) More Varied Object Sizes in Aerial Images: As both the spatial resolution and viewpoint of the sensor vary greatly in aerial imaging [1], [17], [18], the object size from bird view is usually more varied compared with the ground images. Specifically, the objects in ground images are usually middle-sized. In contrast, there are much more small-sized objects in aerial images, but some of the objects such as airport and roundabout are extremely large sized. As a result, the average object size from aerial images is much higher than the ground images [shown in Fig. 1(a) and (c)].

Thus, it is difficult for existing convolutional neural networks (CNNs) with a fixed receptive field to fully perceive the scene scheme of an aerial image due to the more varied sizes of key objects [1], [5], [19], [20], [21], which pulls down the understanding capability of a model for aerial scenes.

2) More Crowded Object Distribution in Aerial Images: Due to the bird view from imaging platforms such as unmanned aerial vehicles and satellites, the aerial images are usually large scale and thus contain much more objects than ground images [1], [2], [22] (see Fig. 1(b) and (d) for an example).

Unfortunately, existing CNNs are capable of preserving the global semantics [11], [12], [13] but are unqualified to highlight the key local regions [23], [24], i.e., region of interests (RoIs), of a scene with complicated object distributions. Therefore, CNNs are likely to be affected by the local semantic information irrelevant to the scene label and fail to predict the

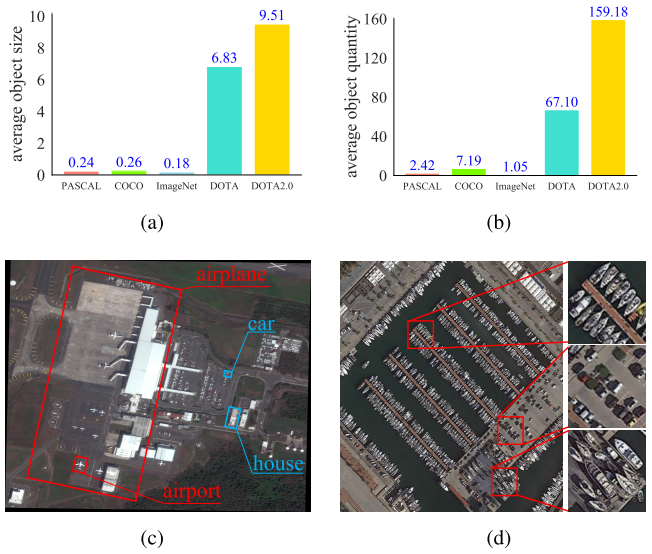

Fig. 1. Different statistics between aerial image samples (from DOTA and DOTA2.0) and ground image samples (from PASCAL, COCO, and ImageNet) on (a) average object sizes and (b) average object quantity. All the original statistics are quoted from [16]. It can be clearly seen that objects from aerial images are much more varied in sizes, and each aerial image usually has much more objects. (c) and (d) Example on the dramatically varied object size and huge object amount in aerial images. (a) Average object size. (b) Average object quantity. (c) Example on object size. (d) Example on object quantity.

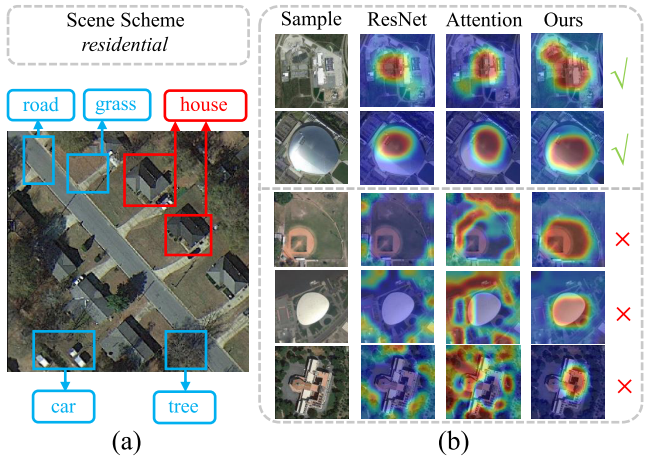

Fig. 2. Intuitive illustration on how aerial scenes contain more objects irrelevant to the scene scheme (a) and existing CNNs can fail to activate the RoIs in aerial scenes (b). In (a), local regions relevant and irrelevant to the scene scheme are labeled in red and blue, respectively. In (b), attention refers to the feature maps from the framework in [29].

correct scene scheme [2], [25], [26], [27], [28] (see Fig. 2 for an intuitive illustration).

B. Motivation and Objectives

We are motivated to tackle the aforementioned challenges in aerial scene classification, hoping to build a more discriminative aerial scene representation. Specific objectives include the following.

1) Highlighting the Key Local Regions in Aerial Scenes: Great effort is needed to highlight the key local regions of an aerial scene for existing deep learning models, so as to correctly perceive the scene scheme rather than activate the background or other local regions in an aerial scene.

Therefore, the formulation of classic multiple instance learning (MIL) [30], [31] is adapted in our work to describe the relation between the aerial scene (bag) and the local image patches (instances). This formulation helps highlight the feature responses of key local regions, and thus enhances the understanding capability for the aerial scene.

2) Aligning the Same Scene Scheme for Multigrain Representation: Allowing for the varied object sizes in an aerial scene, it is natural to use existing multiscale convolutional features [18], [19], [20], [21] for more discriminative aerial scene representation. However, given the aforementioned complicated object distribution in the aerial scene, whether the representation of each scale learned from existing multiscale solutions can focus on the scene scheme remains to be an open question but is crucial to depict the aerial scenes.

Hence, different from existing multiscale solutions [32], we extend the classic MIL formulation to a multigrain manner under the existing deep learning pipeline, in which a set of instance representations is built from multigrain convolutional features. More importantly, in the semantic fusion stage, we develop a simple yet effective strategy to align the instance representation from each grain to the same scene scheme.

C. Contribution

To realize the aforementioned objectives, our contribution in this article can be summarized as follows.

1) We propose an all grains, one scheme (AGOS) framework for aerial scene classification. *To the best of our knowledge*, we are the first to formulate the classic MIL into deep multigrain form. Notably, our framework can be adapted into the existing CNNs in a plug-and-play manner.
2) We propose a bag scheme self-alignment strategy, which allows the instance representation from each grain to highlight the key instances corresponding to the bag scheme without additional supervision. Technically, it is realized by our self-aligned semantic fusion (SSF) module and semantic-aligning loss function.
3) We propose a multigrain perception (MGP) module for multigrain convolutional feature extraction. Technically, the absolute difference from each two adjacent grains generates more discriminative aerial scene representation.
4) Extensive experiments not only validate the state-of-the-art performance of our AGOS on three aerial scene classification benchmarks but also demonstrate the generalization capability of our AGOS on a variety of CNN backbones and two other classification domains.

This article is an extension of our conference paper accepted by the International Conference on Acoustics, Speech, and Signal Processing (ICASSP) 2021 [33]. Compared with [33], the specific improvement of this article includes: 1) the newly designed bag scheme self-alignment strategy, realized by our SSF module and the corresponding loss function, is capable of aligning the bag scheme to the instance representation from each grain; 2) we design an MGP module, which

additionally learns the *base instance representation*, to align the bag scheme and to highlight the key local regions in aerial scenes; and 3) empirically, our AGOS demonstrates superior performance against our initial version [33]. Also, more experiments, discussion, and visualization are provided to analyze the insight of our AGOS.

The rest of this article is organized as follows. In Section II, related work is provided. In Section III, the proposed method is demonstrated. In Section IV, we report and discuss the experiments on three aerial image scene classification benchmarks. Finally, in Section V, the conclusion is drawn.

II. Related Work

A. Aerial Scene Classification

Aerial scene classification remains a heated research topic for both the computer vision and the remote sensing community. In terms of the utilized features, these solutions are usually divided into the low-level (e.g., color histogram [34], wavelet transformation [35], local binary pattern [36], [37], etc.), middle-level (e.g., bag of visual words [38], potential latent semantic analysis [39], [40], latent Dirichlet allocation [41], etc.), and high-level feature-based methods.

High-level feature methods, also known as deep learning methods, have become the dominant paradigm for aerial scene classification in the past decade. Major reasons accounting for its popularity include their stronger feature representation capability and end-to-end learning manner [42], [43].

Among these deep learning-based methods, CNNs are the most commonly utilized [2], [18], [19], [20], [21], [44], as the convolutional filters are effective to extract multilevel features from the image. In the past two years, CNN-based methods (e.g., DSENet [45], MS2AP [46], MSDFF [47], CADNet [48], LSENet [5], GBNet [49], MBLANet [50], MG-CAP [51], Contourlet CNN [52], STHP [53], SAGM [54], DARTS [55], LML [56], and GCSANet [57]) still remain heated for aerial scene classification. On the other hand, recurrent neural network (RNN)-based [25], autoencoder-based [58], [59], and generative adversarial network (GAN)-based [60], [61] approaches have also been reported effective for aerial scene classification.

Meanwhile, although recently vision transformer (ViT) [62], [63], [64] has also been reported to achieve high classification performance for remote sensing scenes, they focus more on the global semantic information with the self-attention mechanism while our motivation focuses more on the local semantic representation and activation of RoIs. Also, the combination of MIL and deep learning is currently based on the CNN pipelines [2], [23], [65], [66], [67]. Hence, the discussion and comparison of ViT-based methods are beyond the scope of this work.

To sum up, as the global semantic representation of CNNs is still not capable enough to depict the complexity of aerial scenes due to the complicated object distribution [2], [25], how to properly highlight the RoIs from the complicated background of aerial images to enhance the scene representation capability still remains rarely explored.

B. Multiscale Feature Representation

Multiscale convolutional feature representation has long been investigated in the computer vision community [68], [69]. As the object sizes are usually more varied in aerial scenes, multiscale convolutional feature representation has also been widely utilized in the remote sensing community for a better understanding of aerial images.

Till now, multiscale feature representation for aerial images can be classified into two categories, that is, using multilevel CNN features in a nontrainable manner and directly extracting multiscale CNN features in the deep learning pipeline.

For the first category, the basic idea is to derive multilayer convolutional features from a pretrained CNN model, and then to feed these features into a nontrainable encoder such as BoW or LDA. Typical works include [19], [21], and [44]. Although the motivation of such approaches is to learn more discriminative scene representation in the latent space, they are not end to end, and the performance gain is usually marginal.

For the second category, the basic idea is to design spatial pyramid pooling [20], [46] or image pyramid [18] to extend the convolutional features into multiscale representation. Generally, such multiscale solutions can be further divided into four categories [32], namely, encoder–decoder pyramid, spatial pyramid pooling, image pyramid, and parallel pyramid.

Although nowadays multiscale representation methods become mature, whether the representation from each scale can effectively activate the RoIs in the scene has not been explored.

C. Multiple Instance Learning

MIL was initially designed for drug prediction [30] and then became an important machine learning tool [31]. In MIL, an object is regarded as a bag, and a bag consists of a set of instances [70]. Generally speaking, there is no specific instance label, and each instance can only be judged as either belonging or not belonging to the bag category. This formulation makes MIL especially qualified to learn from the weakly annotated data [67], [71], [72].

The effectiveness of MIL has also been validated on a series of computer vision tasks such as image recognition [73], saliency detection [74], [75], spectral–spatial fusion [76], and object localization/detection [77], [78], [79], [80], [81].

On the other hand, the classic MIL theory has also been enriched. Specifically, Sivan and Naftali [82] relaxed the Boolean OR assumption in MIL formulation, so that the relation between bag and instances becomes more general. More recently, Alessandro *et al.* [83] investigated a three-level MIL. The three hierarchical levels are in a vertical manner, and they are top-bag, sub-bag, and instance, where the sub-bag is an embedding between the top-bag and instances. Note that our deep MIL under multigrain form is quite distinctive from [83], as our formulation still has two hierarchical levels, i.e., bag and instances, and the instance representation is generated from multigrain features.

In the past few years, deep MIL draws some attention, in which MIL has the trend to be combined with deep learning in a trainable manner. To be specific, Wang *et al.* [67] utilized

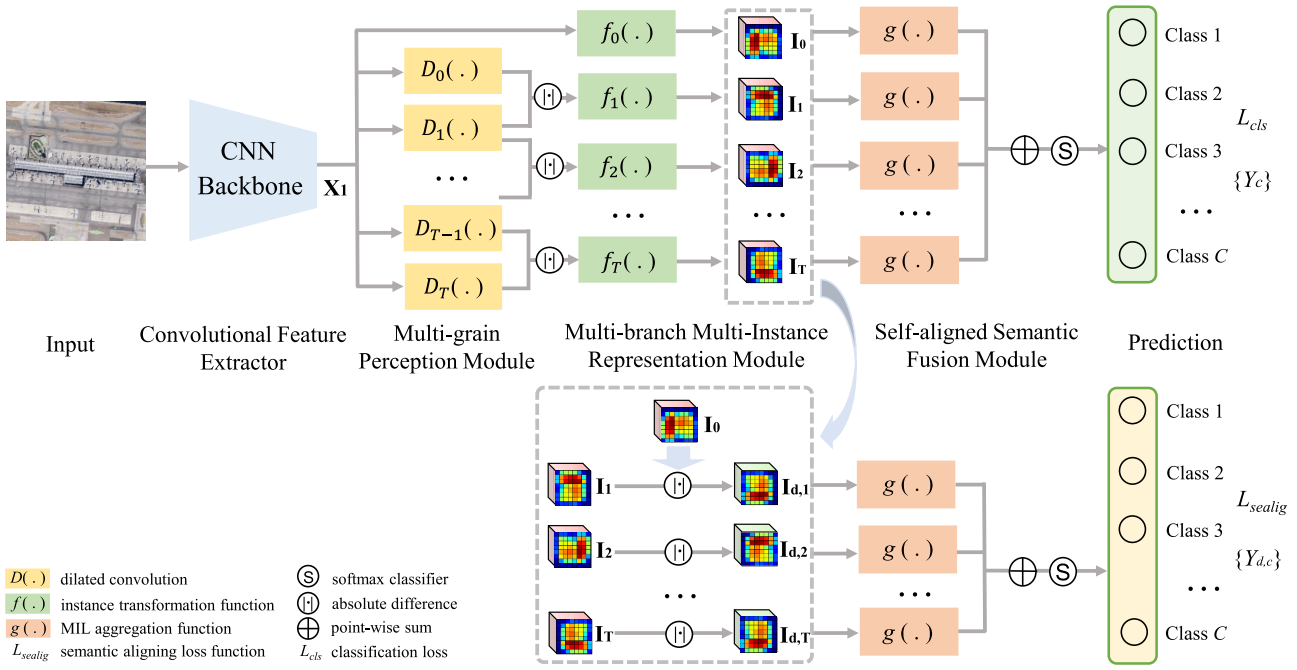

Fig. 3. Our proposed AGOS framework.

TABLE I

BRIEF SUMMARY ON THE ATTRIBUTES OF OUR AGOS AND RECENT DEEP MIL ALGORITHMS. FOR SPACE ATTRIBUTE, *I*: INSTANCE-SPACE PARADIGM, *E*: EMBEDDING-SPACE PARADIGM; AGGREGATE ATTRIBUTE REFERS TO THE MIL AGGREGATION FUNCTION; GRAIN, *S*: SINGLE-GRAIN, *M*: MULTIGRAIN

Method	Space	Aggregation	Hierarchy	Grain
[67]	I, E	max/mean	bag-instance	S
[23]	E	attention	bag-instance	S
[2]	I	attention	bag-instance	S
[65], [66]	I	attention	bag-instance	S
[83]	E	max	top bag-bag-instance	S
AGOS (Ours)	I	mean	bag-instance	M

either max pooling or mean pooling to aggregate the instance representation in the neural networks. Later, Ilse *et al.* [23] used a gated attention module to generate the weights, which are utilized to aggregate the instance scores. Bi *et al.* [2] utilized both spatial attention module and channel-spatial attention module to derive the weights and directly aggregate the instance scores into bag-level probability distribution. More recently, Shi *et al.* [65], [66] embedded the attention weights into the loss function so as to guide the learning process for deep MIL.

On the other hand, it should be noted that the MIL formulation discussed in this section is distinctive from some recent works [84] that formulate each geo-spatial object as an instance, as our MIL formulation: 1) treat each equally partitioned image patch as an instance and the entire image as a bag and 2) the dimension of instance representation is equal to the bag representation so that the feature response is represented from the semantic-level and is supposed to be independent of each other. Thus, the rigid MIL formulation better benefits the semantic-level representation for local regions, and thus better highlights the key local regions for remote sensing scenes.

III. PROPOSED METHOD

A. Preliminary

1) Classic and Deep MIL Formulation: For our aerial scene classification task, according to the classic MIL formulation [30], [31], a scene X is regarded as a bag, and the bag label Y is identical as the scene category. As each bag X consists of a set of instances $\{x_1, x_2, \ldots, x_l\}$, each image patch of the scene is regarded as an instance.

All the instances indeed have labels $y_1, y_2, \ldots, y_l$, but all these instance labels are weakly annotated, i.e., we only know each instance either belongs to (denoted as 1) or does not belong to (denoted as 0) the bag category. Then, whether or not a bag belongs to a specific category c is determined via

$$Y = \begin{cases} 0 & \text{if } \sum_{t=1}^{l} y_t = 0 \\ 1 & \text{else.} \end{cases} \tag{1}$$

In deep MIL, as the feature response from the gradient propagation is continuous, the bag probability prediction Y is assumed to be continuous in [0, 1] [2], [23]. It is determined to be a specific category c via

$$Y = \begin{cases} 1 & \text{if } p_c = \max\{p_1, \ldots, p_C\} \\ 0 & \text{else} \end{cases} \tag{2}$$

where $p_1, p_2, \ldots, p_c, \ldots, p_C$ denotes the bag probability prediction of all the total C bag categories.

2) MIL Decomposition: In both classic MIL and deep MIL, the transition between instances $\{x_s\}$ (where $s = 1, 2, \ldots, l$) to the bag label Y can be presented as

$$Y = h(g(f(\{x_s\}))) \tag{3}$$

where f denotes a transformation which converts the instance set into an instance representation, g denotes the MIL aggregation function, and h denotes a transformation to get the bag probability distribution.

3) Instance Space Paradigm: The combination of MIL and deep learning is usually conducted in either instance space [2], [66], [67] or embedding space [23]. Embedding space-based solutions offer a latent space between the instance representation and bag representation, but this latent space in the embedding space can sometimes be less precise in depicting the relation between instance and bag representation [2], [23]. In contrast, instance space paradigm has the advantage to generate the bag probability distribution directly from the instance representation [2], [67]. Thus, the h transformation in 3 becomes an identity mapping, and it is rewritten as

$$Y = g(f(\{x_s\})). \tag{4}$$

4) Problem Formulation: As we extend MIL into multigrain form, the transformation function f in 4 is extended to a set of transformations $\{f_t\}$ (where $t = 1, 2, \ldots, T$). Then, Y is generated from all these grains and thus 4 can be presented as

$$Y = g(f_1(\{x_s\}), f_2(\{x_s\}), \ldots, f_T(\{x_s\})). \tag{5}$$

Hence, how to design a proper and effective transformation set $\{f_t\}$ and the corresponding MIL aggregation function g under the existing deep learning pipeline is our major task.

5) Objective: Our objective is to classify the input scene X in the deep learning pipeline under the formulation of multigrain multi-instance learning. To summarize, the overall objective function can be presented as

$$\arg\min_{W,b} \mathcal{L}(Y, g(f_1(\{x_s\}), \ldots, f_T(\{x_s\}); W, b)) + \Psi(W) \tag{6}$$

where W and b are the weight matrix and the bias matrix to train the entire framework, $\mathcal{L}$ is the loss function, and Ψ is the regularization term.

Moreover, how the instance representation of each grain $f_t(\{x_s\})$ is aligned to the same bag scheme is also taken into account in the stage of instance aggregation g and optimization $\mathcal{L}$, which can be generally presented as

$$\begin{aligned} s.t. \quad & g(f_1(\{x_s\})) = g(f_2(\{x_s\})) = \cdots \\ & = g(f_t(\{x_s\})) = \cdots = g(f_T(\{x_s\})) = Y_c \end{aligned} \tag{7}$$

where Y_c denotes the ground truth of the bag scheme, i.e., the category that the bag belongs to.

B. Network Overview

As is shown in Fig. 3, our proposed AGOS framework consists of three components after the CNN backbone. To be specific, the MGP module (in Section III-C) implements our proposed differential dilated convolution (DDC) on the convolutional features so as to get a discriminative multigrain representation. Then, the multigrain feature presentation is fed into our multibranch multi-instance representation (MBMIR) module (in Section III-D), which converts the aforementioned features into instance representation, and then directly generates the bag-level probability distribution. As aligning the instance representation from each grain to the same bag scheme is another important objective, we propose a bag scheme self-alignment strategy, which is technically fulfilled by our self-aligned semantic module (in Section III-E) and the corresponding loss function (in Section III-F). In this way, the entire framework is trained in an end-to-end manner.

C. MGP Module

1) Motivation: Our MGP module intends to convert the convolutional feature from the backbone to multigrain representations. Different from existing multiscale strategies [18], [19], [20], [21], our module builds same-sized feature maps by perceiving multigrain representations from the same convolutional feature. Then, the absolute difference of the representations from each two adjacent grains is calculated to highlight the differences from a variety of grains for more discriminative representation (shown in Fig. 4).

2) Dilated Convolution: Dilated convolution is capable of perceiving the feature responses from different receptive fields while keeping the same image size [85]. Thus, it has been widely utilized in many visual tasks in the past few years.

Generally, dilation rate r is the parameter to control the window size of a dilated convolution filter. For a 3×3 convolution filter, a dilation rate r means that $r - 1$ zero-valued elements will be padded into two adjacent elements of the convolution filter. For example, for a 3×3 convolution filter, a dilation rate will expand the original convolutional filter to the size of $(2r + 1) \times (2r + 1)$. Specifically, when $r = 0$, there is no zero padding and the dilated convolutional filter degrades into the traditional convolution filter.

3) Multigrain Dilated Convolution: Let the convolutional feature from the backbone be denoted as $\mathbf{X}_1$. Assume that there are T grains in our MGP, then T dilated convolution filters are implemented on the input $\mathbf{X}_1$, which we denote as $D_1, D_2, \ldots, D_T$, respectively. Apparently, the set of multigrain dilated convolution feature representation $\mathbb{X}'_1$ from the input $\mathbf{X}_1$ can be presented as

$$\mathbb{X}'_1 = \left\{\mathbf{X}_1^1, \mathbf{X}_1^2, \ldots, \mathbf{X}_1^T\right\} \tag{8}$$

where we have

$$\mathbf{X}_1^t = D_t(\mathbf{X}_1) \tag{9}$$

and $t = 1, 2, \ldots, T$.

The determination of the dilation rate r for the multigrain dilated convolution set $\{D_t\}$ follows the existing rules [85] that r is set as an odd value, i.e., $r = 1, 3, 5, \ldots$. Hence, for D_t, the dilation rate r is $2t - 1$.

4) Differential Dilated Convolution: To reduce the feature redundancy from different grains while stressing the discriminative features that each grain contains, absolute difference of each pair of two adjacent representations in $\mathbb{X}'_1$ is calculated via

$$\mathbf{X}_{d,t} = \|D_t(\mathbf{X}_1) - D_{t-1}(\mathbf{X}_1)\| \tag{10}$$

where $\|\cdot\|$ denotes the absolute difference, and $\mathbf{X}_{d,t}$ $(t = 1, 2, \ldots, T)$ denotes the calculated differential

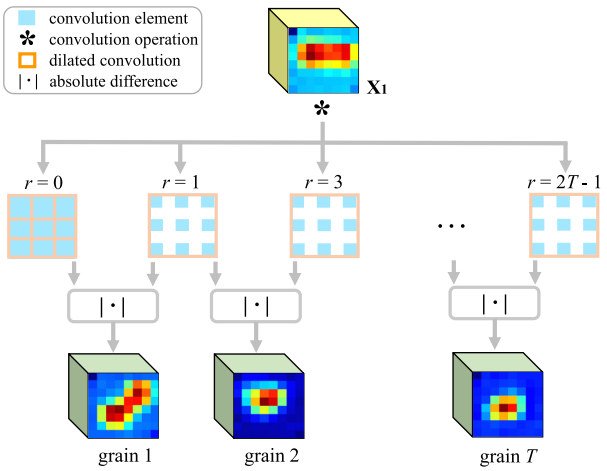

Fig. 4. Demonstration on our MGP module. The inputted convolutional features are processed by a set of dilated convolution with different dilated rates. Then, absolute difference of each adjacent representation pair is calculated to get the multigrain representation output.

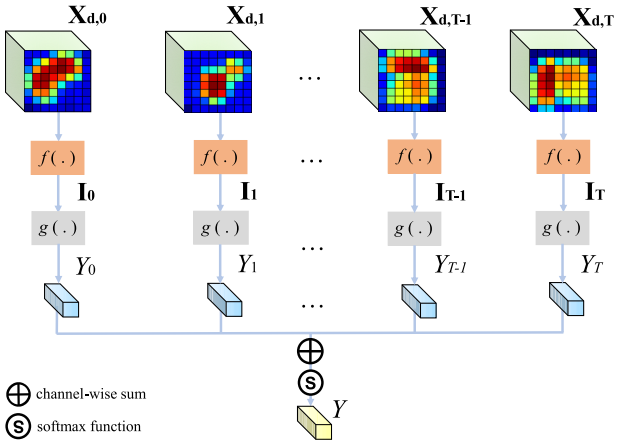

Fig. 5. Illustration on the instance representation and the generation of bag probability distribution.

dilated convolutional feature representation. It is worth noting that when $t = 1$, $D_0(\mathbf{X}_1)$ means that the dilated convolution degrades to the conventional convolution.

Finally, the output of this MGP module is a set of convolutional feature representation $\mathbb{X}_1''$, presented as

$$\mathbb{X}_1'' = \left\{\mathbf{X}_{d,0}, \mathbf{X}_{d,1}, \mathbf{X}_{d,2}, \ldots, \mathbf{X}_{d,T}\right\} \tag{11}$$

where $\mathbf{X}_{d,0}$ denotes the base representation in our bag scheme self-alignment strategy, the function of which will be discussed in detail in Sections III-D and III-E.

Generally, $\mathbf{X}_{d,0}$ is a direct refinement of the input $\mathbf{X}_1$ in the hope of highlighting the key local regions. The realization of this objective is straightforward, as the 1×1 convolutional layer has recently been reported to be effective in refining the feature map and highlighting the key local regions [2], [10]. This process is presented as

$$\mathbf{X}_{d,0}^{W\times H\times C_1} = W_{d,0}^{W\times H\times C_1}\mathbf{X}_1^{W\times H\times C_1} + b_{d,0}^{W\times H\times C_1} \tag{12}$$

where $W_{d,0}$ and $b_{d,0}$ denote the weight matrix and the bias matrix of this 1×1 convolutional layer, and W and H denote the width and height of the feature representation $\mathbf{X}_1$. Moreover, as the channel number C_1 of $\mathbf{X}_{d,0}$ keeps the same with $\mathbf{X}_1$, so the number of convolutional filters in this convolutional layer also equals to the aforementioned channel number C_1.

5) Summary: As shown in Fig. 4 and depicted from (8)–(12), in our MGP, the inputted convolutional features are processed by a series of dilated convolution with different dilated rates. Then, the absolute difference of each representation pair from the adjacent two grains (i.e., $r = 1$ and $r = 3$, $r = 3$ and $r = 5$) is calculated as output, so as to generate the multigrain differential convolutional features for more discriminative representation.

D. MBMIR Module

1) Motivation: The convolutional feature representations $\mathbb{X}_1''$ from different grains contain different discriminative information in depicting the scene scheme. Hence, for the representation $\mathbf{X}_{d,t}$ from each grain ($t = 1, 2, \ldots, T$), a deep MIL module is utilized to highlight the key local regions. Specifically, each module converts the convolutional representation into an instance representation, and then utilizes an aggregation function to get the bag probability distribution. All these parallel modules are organized as a whole for our MBMIR module.

2) Instance Representation Transformation: Each convolutional representation $\mathbf{X}_{d,t}$ (where $t = 0, 1, \ldots, T$) in the set $\mathbb{X}_1''$ needs to be converted into an instance representation by a transformation at first, which is exactly the f function in 3 and 4. Specifically, for $\mathbf{X}_{d,t}$, this transformation can be presented as

$$\mathbf{I}_t^{W\times H\times C} = W_{d,t}^{1\times 1\times C}\mathbf{X}_{d,t}^{W\times H\times C_1} + b_{d,t}^{1\times 1\times C} \tag{13}$$

where $\mathbf{I}_t$ is the corresponding instance representation, $W_{d,t}$ is the weight matrix of this 1×1 convolutional layer, $b_{d,t}$ is the bias matrix of this convolutional layer, and $t = 0, 1, 2, \ldots, T$.

Regarding the channel number, assume that there are overall C bag categories, then the instance representation $\mathbf{I}_t$ also has C channels so that the feature map of each channel corresponds to the response on a specific bag category, as it has been suggested in 2. Thus, the number of 1×1 convolution filters in this layer is also C.

Apparently, each 1×1 image patch on the $W \times H$-sized feature map corresponds to an instance. As there are C bag categories and the instance representation also has C channels, each instance corresponds to a C-dimensional feature vector and thus each dimension corresponds to the feature response on the specific bag category (demonstrated in Fig. 5).

3) Multigrain Instance Representation: After processed by 13, each differential dilated convolutional feature representation $\mathbf{I}_t$ generates an instance representation at the corresponding grain. Generally, the set of multigrain instance representation $\{\mathbf{I}_t\}$ can be presented as $\{\mathbf{I}_0, \mathbf{I}_1, \ldots, \mathbf{I}_T\}$.

4) MIL Aggregation Function: As is presented in 4, under the instance space paradigm, the MIL aggregation function g converts the instance representation directly into the bag probability distribution. On the other hand, the MIL aggregation

function is required to be permutation-invariant [30], [31] so that the bag scheme prediction is invariant to the change of instance positions. Therefore, we utilize the mean-based MIL pooling for aggregation.

Specifically, for the instance representation $\mathbf{I}_t$ from each scale, assume that each instance can be presented as $\mathbf{I}_t^{w,h}$, where $1 \leq w \leq W$ and $1 \leq h \leq H$. Then, the bag probability from a certain grain t, denoted as Y_t, is generated via

$$Y_t = \frac{\sum_{w=1}^{W} \sum_{h=1}^{H} \mathbf{I}_t^{w,h}}{W \times H}. \tag{14}$$

Apparently, after aggregation, Y_t can be regarded as a C-dimensional feature vector. This process can be technically solved by a global average pooling (GAP) function in existing deep learning frameworks.

5) Bag Probability Generation: The final bag probability distribution Y is the sum of the predictions from each grain, which is calculated as

$$Y = \text{softmax}\left(\sum_{t=0}^{T} Y_i\right) \tag{15}$$

where softmax is the softmax function for normalization.

To sum up, the pseudocode of all the aforementioned steps on learning MBMIR is summarized in Algorithm 1, in which conv1d refers to the 1×1 convolution layer in 12.

Algorithm 1 Learning MBMIR

Input: convolutional feature $\mathbf{X}_1$, grain number T
Output: bag probability distribution Y, instance representation set $\{\mathbf{I}_t\}$

zero initialization Y
for $t = 0 \rightarrow T$ **do**
  $\mathbf{X}_t^t \leftarrow D_t(\mathbf{X}_1)$
end for
for $t = 0 \rightarrow T$ **do**
  if t $\geq$ 1 **then**
    $\mathbf{X}_{d,t} \leftarrow \|\mathbf{X}_1^t - \mathbf{X}_1^{t-1}\|$
  else
    % $conv1d$: the convolutional layer in Eq. 12
    $\mathbf{X}_{d,t} \leftarrow conv1d(\mathbf{X}_1)$
  end if
end for
for $t = 0 \rightarrow T$ **do**
  $\mathbf{I}_t \leftarrow f_t(\mathbf{X}_{d,t})$
end for
for $t = 0 \rightarrow T$ **do**
  $Y_t \leftarrow g(\mathbf{I}_t)$
  $Y += Y_t$;
end for
$Y \leftarrow softmax(Y)$

E. SSF Module

1) Motivation: To make the instance representation from different grains focus on the same bag scheme, we propose a bag scheme self-alignment strategy. Specifically, it first finds the difference between a base instance representation and the instance representations from other grains, and then minimizes this difference by our semantic aligning loss function. Fig. 6 offers an intuitive illustration of this module.

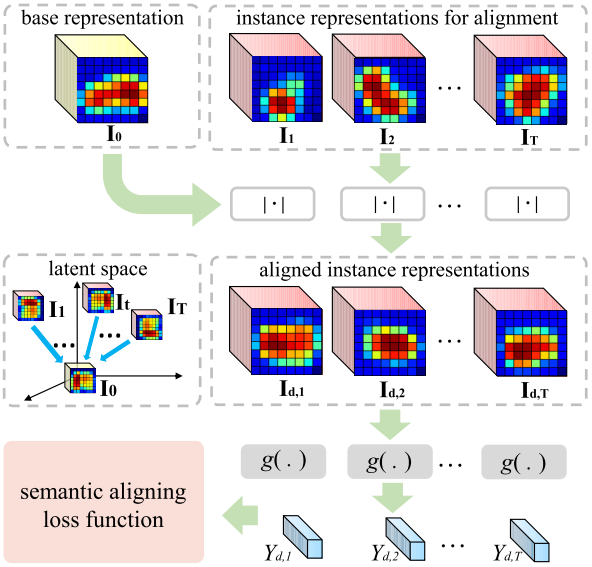

Fig. 6. Demonstration of our SSF module.

2) Base Representation: The instance representation $\mathbf{I}_0$, only processed by a 1×1 convolutional layer rather than any dilated convolution, is selected as our base representation. One of the major reasons for using $\mathbf{I}_0$ as the base representation is that the processing of the 1×1 convolutional layer can highlight the key local regions of an aerial scene.

3) Difference From Base Representation: The absolute difference between other instance representation $\mathbf{I}_t$ (here $t = 1, 2, \ldots, T$) and the base representation $\mathbf{I}_0$ is calculated to depict the differences between the base representation and the other instance representation from different grains t. This process can be presented as

$$\mathbf{I}_{d,t} = \|\mathbf{I}_t - \mathbf{I}_0\| \tag{16}$$

where $\|\cdot\|$ denotes the absolute difference, $\mathbf{I}_{d,t}$ denotes the difference of each pair of instance representations at the corresponding grains, and $t = 1, 2, \ldots, T$.

4) Bag Scheme Alignment: By implementing the MIL aggregation function g on $\mathbf{I}_{d,t}$, the bag probability that depicts the difference of instance representations from adjacent grains, denoted as $Y_{d,t}$, is generated via

$$Y_{d,t} = \frac{\sum_{w=1}^{W} \sum_{h=1}^{H} \mathbf{I}_{d,t}^{w,h}}{W \times H} \tag{17}$$

where all the notations follow the paradigm in 14, that is, $1 \leq w \leq W$ and $1 \leq h \leq H$, and W and H denote the width and the height, respectively.

The overall bag scheme probability distribution differences Y_d between the base instance representation $\mathbf{I}_{d,0}$ and other instance representations $\mathbf{I}_{d,t}$ (where $t = 1, 2, \ldots, T$) can be

calculated as

$$Y_d = \text{softmax}\left(\sum_{t=1}^{T} Y_{d,t}\right) = \text{softmax}\left(\frac{\sum_{t=1}^{T}\sum_{w=1}^{W}\sum_{h=1}^{H}\mathbf{I}_{d,t}^{w,h}}{W \times H}\right) \quad (18)$$

where softmax denotes the softmax function.

By minimizing the overall bag scheme probability differences Y_d, the bag prediction from each grain tends to be aligned to the same category. Technically, this minimization process is realized by our loss function in Section III-F.

Algorithm 2 Bag Scheme Self-Alignment Strategy

Input: instance representation set $\{\mathbf{I}_t\}$, bag probability distribution Y, ground truth of bag scheme Y_c
Output: loss function L for optimization

```
zero initialization Y_d
for t = 1 → T do
    I_{d,t} ← ‖I_t − I_0‖
end for
for t = 1 → T do
    Y_{d,t} ← g(I_{d,t})
    Y_d += Y_{d,t}
end for
L_crs: the cross-entropy loss function
L_cls ← L_crs(Y, Y_c)
L_sealig ← L_crs(Y_d, Y_c)
L ← L_cls + α L_sealig
```

F. Loss Function

1) Cross-Entropy Loss Function: Following the aforementioned notations, still assume that Y is the predicted bag probability distribution (in 15), Y_c is the exact bag category, and there are overall C categories. Then, the classic cross-entropy loss function serves as the classification loss L_{cls}, presented as

$$L_{\text{cls}} = -\frac{1}{C}\sum_{i=1}^{C}(Y_c \log Y_i + (1 - Y_c)\log(1 - Y_i)). \quad (19)$$

2) Semantic-Aligning Loss Function: The formulation of the classic cross-entropy loss is also adapted to minimize the overall bag probability differences Y_d in 18. Thus, this semantic-aligning loss term L_{sealig} is presented as

$$L_{\text{sealig}} = -\frac{1}{C}\sum_{i=1}^{C}\big(Y_c \log Y_{d,i} + (1 - Y_c)\log\big(1 - Y_{d,i}\big)\big). \quad (20)$$

3) Overall Loss: The overall loss function L to optimize the entire framework is the weighted average of the aforementioned two terms L_{cls} and L_{sealig}, calculated as

$$L_{=}L_{\text{cls}} + \alpha L_{\text{sealig}} \quad (21)$$

where α is the hyperparameter to balance the impact of the aforementioned two terms. Empirically, we set $\alpha = 5 \times 10{-}4$.

The pseudocode of our proposed overall bag scheme self-alignment strategy is provided in Algorithm 2, which covers the content in Sections III-E and III-F.

TABLE II
DATA PARTITION AND EVALUATION PROTOCOLS OF THE THREE AERIAL SCENE CLASSIFICATION BENCHMARKS FOLLOWING THE EVALUATION PROTOCOLS [17], [87], WHERE RUNS DENOTES THE REQUIRED INDEPENDENT REPETITIONS TO REPORT THE CLASSIFICATION ACCURACY

	#classes	# samples	training / testing ratio	runs
UCM	21	210	50% / 50%, 80% / 20% [17]	10
AID	30	220-420	20% / 80%, 50% / 50% [17]	10
NWPU	45	700	10% / 90%, 20% / 80% [87]	10

IV. EXPERIMENT AND ANALYSIS

A. Datasets

1) UC Merced Land Use Dataset (UCM): Till now, it is the most commonly used aerial scene classification dataset. It has 2100 samples in total, and there are 100 samples for each of the 21 scene categories [86]. All these samples have the size of 256 × 256 with a 0.3-m spatial resolution. Moreover, all these samples are taken from the aerial craft, and both the illumination condition and the viewpoint of all these aerial scenes are quite close.

2) Aerial Image Dataset (AID): It is a typical large-scale aerial scene classification benchmark with an image size of 600 × 600 [17]. It has 30 scene categories with a total amount of 10 000 samples. The sample number per class varies from 220 to 420. As the imaging sensors in photographing the aerial scenes are more varied in aerial image dataset (AID) benchmark, the illumination conditions and viewpoint are also more varied. Moreover, the spatial resolution of these samples varies from 0.5 to 8 m.

3) Northwestern Polytechnical University (NWPU) Dataset: This benchmark is more challenging than the UCM and AID benchmarks, as the spatial resolution of samples varies from 0.2 to 30 m [87]. It has 45 scene categories and 700 samples per class. All the samples have a fixed image size of 256 × 256. Moreover, the imaging sensors and imaging conditions are more varied and complicated than AID.

B. Evaluation Protocols

Following the existing experiment protocols [17], [87], we report the overall accuracy (OA) in the format of "average±deviation" from ten independent runs on all these three benchmarks.

Experiments on UCM, AID, and Northwestern Polytechnical University (NWPU) datasets are all in accordance with the corresponding training ratio settings. To be specific, for UCM, the training set proportions are 50% and 80%, respectively; for AID, the training set proportions are 20% and 50%, respectively; and for NWPU, the training set proportions are 10% and 20%, respectively.

C. Experimental Setup

1) Parameter Settings: In our AGOS, C_1 is set 256, indicating that there are 256 channels for each dilated convolutional filter. Moreover, T is set 3, which means that there are four branches in our AGOS module. Finally, C is set 21, 30, and

45, respectively, when trained on UCM, AID, and NWPU benchmarks, respectively, which equals to the total scene category number of these three benchmarks.

2) Model Initialization: A set of backbones, including ResNet-50, ResNet-101, and DenseNet-121, all utilize pre-trained parameters on ImageNet as the initial parameters. For the rest of our AGOS framework, we use random initialization for weight parameters with a standard deviation of 0.001. All bias parameters are set to zero for initialization.

3) Training Procedure: The model is optimized by the Adam optimizer with $\beta_1 = 0.9$ and $\beta_2 = 0.999$. Moreover, the batch size is set 32. The initial learning rate is set to be 0.0001 and is divided by 0.5 every 30 epochs until finishing 120 epochs. To avoid the potential over-fitting problem, L_2 normalization with a parameter setting of 5×10^{-4} is utilized and a dropout rate of 0.2 is set in all the experiments.

4) Other Implementation Details: Our experiments were conducted under the TensorFlow deep learning framework by using the Python program language. All the experiments were implemented on a work station with 64-GB RAM and an i7-10700 central processing unit (CPU). Moreover, two RTX 2080 SUPER graphics processing units (GPUs) are utilized for acceleration. Our source code is available at https://github.com/BiQiWHU/AGOS.

D. Comparison With State-of-the-Art Approaches

We compare the performance of our AGOS with three hand-crafted features (PLSA, BOW, and LDA) [17], [87], three typical CNN models (AlexNet, VGG, and GoogLeNet) [17], [87], 22 latest CNN-based state-of-the-art approaches (MIDCNet [2], RANet [29], APNet [88], SPPNet [20], DCNN [28], TEXNet [89], MSCP [18], VGG + FV [21], DSENet [45], MS2AP [46], MSDFF [47], CADNet [48], LSENet [5], GBNet [49], MBLANet [50], MG-CAP [51], Contourlet CNN [52], STHP [53], SAGM [54], DARTS [55], LML [56], and GCSANet [57]), one RNN-based approach (ARCNet [25]), two autoencoder-based approaches (SGUFL [59] and PARTLETS [58]), and two GAN-based approaches (MARTA [60] and AGAN [61]), respectively. The performance under the backbone of ResNet-50, ResNet-101, and DenseNet-121 is all reported for fair evaluation, as some latest methods [47], [48] use much deeper networks as backbone.

1) Results and Comparison on UCM: In Table III, the classification accuracy of our AGOS and other state-of-the-art approaches is listed. It can be seen in the table.

1) Both our current AGOS framework and its initial version [33] outperform the existing state-of-the-art approaches on both cases when the training ratios are 50% and 80%, respectively, including CNN-based, RNN-based, autoencoder-based, and GAN-based approaches. Plus, no matter using lighter backbone ResNet-50 or stronger backbone ResNet-101 and DenseNet-121, our AGOS shows superior performance against all the compared methods.
2) Our AGOS framework significantly outperforms the existing approaches that exploit the multiscale representation for aerial scenes [18], [20], [21], [46].

TABLE III

CLASSIFICATION ACCURACY OF OUR AGOS AND OTHER APPROACHES ON UCM DATASET. RESULTS PRESENTED IN THE FORM OF "AVERAGE±DEVIATION" [17]; METRICS PRESENTED IN %; *H*, *C*, *R*, *A*, AND *G* DENOTE HAND-CRAFTED, CNN-, RNN-, AUTOENCODER-, AND GAN-BASED APPROACHES, RESPECTIVELY. VALUES IN BOLD AND IN BLUE DENOTE THE BEST AND SECOND BEST RESULTS

Method	Type & Year	Training ratio	
		50%	80%
PLSA(SIFT) [17]	*H*, 2017	67.55±1.11	71.38±1.77
BoVW(SIFT) [17]	*H*, 2017	73.48±1.39	75.52±2.13
LDA(SIFT) [17]	*H*, 2017	59.24±1.66	75.98±1.60
AlexNet [17]	*C*, 2017	93.98±0.67	95.02±0.81
VGGNet-16 [17]	*C*, 2017	94.14±0.69	95.21±1.20
GoogLeNet [17]	*C*, 2017	92.70±0.60	94.31±0.89
ARCNet [25]	*R*, 2018	96.81±0.14	99.12±0.40
SGUFL [59]	*A*, 2014	—-	82.72±1.18
PARTLETS [58]	*A*, 2015	88.76±0.79	—-
MARTA [60]	*G*, 2017	85.50±0.69	94.86±0.80
AGAN [61]	*G*, 2019	89.06±0.50	97.69±0.69
VGG+FV [21]	*C*, 2017	—-	98.57±0.34
SPPNet [20]	*C*, 2017	94.77±0.46*	96.67±0.94*
TEXNet [89]	*C*, 2017	94.22±0.50	95.31±0.69
DCNN [28]	*C*, 2018	—-	98.93±0.10
MSCP [18]	*C*, 2018	—-	98.36±0.58
APNet [88]	*C*, 2019	95.01±0.43	97.05±0.43
MSDFF [47]	*C*, 2020	98.85±—-	99.76±—-
CADNet [48]	*C*, 2020	98.57±0.33	99.67±0.27
MIDCNet [2]	*C*, 2020	94.93±0.51	97.00±0.49
RANet [29]	*C*, 2020	94.79±0.42	97.05±0.48
GBNet [49]	*C*, 2020	97.05±0.19	98.57±0.48
MG-CAP [51]	*C*, 2020	—-	99.00±0.10
DSENet [45]	*C*, 2021	96.19±0.13	99.14 ±0.22
MS2AP [46]	*C*, 2021	98.38±0.35	99.01 ±0.42
Contourlet CNN [52]	*C*, 2021	—-	98.97±0.21
LSENet [5]	*C*, 2021	97.94±0.35	98.69±0.53
MBLANet [50]	*C*, 2021	—-	99.64±0.12
DMSMIL [33]	*C*, 2021	99.09±0.36	99.45±0.32
STHP [53]	*C*, 2022	98.57±0.33	99.16±0.27
GCSANet [57]	*C*, 2022	98.32±0.71	99.31±0.56
AGOS (ResNet-50)	*C*	99.24±0.22	99.71±0.25
AGOS (ResNet-101)	*C*	99.29±0.23	99.86±0.17
AGOS (DenseNet-121)	*C*	**99.34±0.20**	**99.88±0.13**

'—-': not reported, '*': not reported & conducted by us

3) Generally speaking, other approaches that achieve the most competitive performance usually highlight the key local regions of an aerial scene [2], [18], [25], [29]. For the autoencoder- and GAN-based approaches, as this aspect remains unexplored, their performance is relatively weak.

Per-category classification accuracy (with ResNet-50 backbone) when the training ratios are 50% and 80% is displayed in Fig. 7(a) and (b), respectively. It is observed that almost all the samples in the UCM are correctly classified. Still, it is notable that the hard-to-distinguish scene categories such as dense residential, medium residential, and sparse residential are all identified correctly.

The potential explanations are summarized as follows.

1) Compared with ground images, aerial images are usually large scale. Thus, the highlight of key local regions related to the scene scheme is vital. The strongest performed approaches, both CNN-based [2], [18], [25], [28], [29] and our AGOS, take the advantage of these strategies.
2) Another important aspect for aerial scene classification is to consider the case that the sizes of key objects

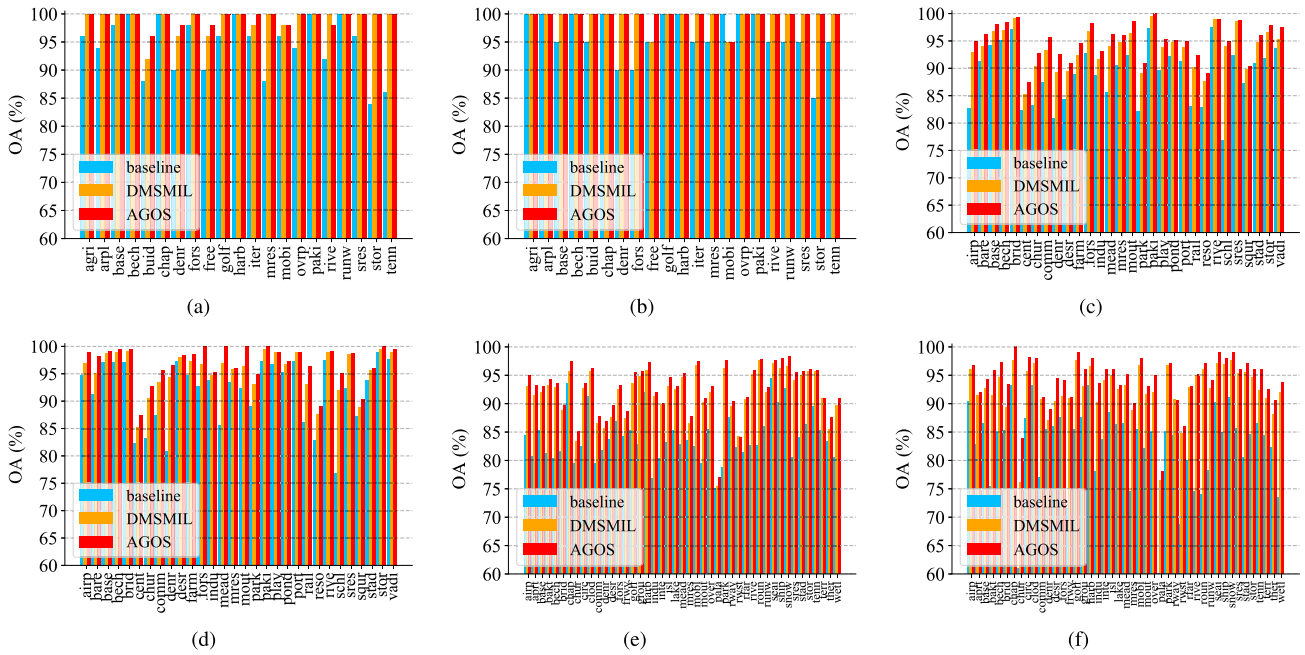

Fig. 7. Classification accuracy of the ResNet-50 baseline and our proposed AGOS with ResNet-50 backbone on three datasets. DMSMIL with orange bar denotes the performance of our initial version [33]; AGOS with red bar denotes the performance of our current version. (a) UCM 50%. (b) UCM 80%. (c) AID 20%. (d) AID 50%. (e) NWPU 10%. (f) NWPU 20%.

in aerial scenes vary a lot. Hence, it is observed that many competitive approaches are utilizing the multiscale feature representation [18], [20], [21], [46]. Our AGOS also takes advantage of this and contains an MGP module. More importantly, our AGOS further allows the instance representation from each grain to focus on the same scene scheme, and thus the performance improves.

3) Generally speaking, the performance of autoencoder-based [58], [59] and GAN [60], [61]-based solutions is not satisfactory, which may also be explained from the lack of the aforementioned capabilities such as the highlight of key local regions and multigrain representation.

2) Results and Comparison on AID: In Table IV, the results of our AGOS and other state-of-the-art approaches on AID are listed. Several observations can be made.

1) Our proposed AGOS with DenseNet-121 outperforms all the state-of-the-art methods under both the training ratio of 20% and 50%. Its ResNet-101 version achieves the second best results under training ratio 50%. Moreover, AGOS with ResNet-50 and our former version [33] also achieve a satisfactory performance on both experiments.
2) Other state-of-the-art approaches that either highlight the key local regions [2], [25], [28], [29] or build a multiscale representation [18], [20], [21], [46] also perform well on both experiments.
3) Similar to the situations in UCM, the strongest performance mainly comes from CNN-based methods [2], [25], [28], [29], while the performance of GAN-based methods is far from satisfactory [60], [61].

Per-category classification accuracy under the training ratio of 20% and 50% is shown in Fig. 7(c) and (d), respectively. It can be seen that most scene categories are well distinguished, and some categories are difficult to classify, i.e., dense residential, medium residential, and sparse residential are also classified well by our solution. Possible explanations include the following.

1) The sample size in AID is generally larger than UCM, and the key objects to determine the scene category are more varied in terms of sizes. As our AGOS can highlight the key local regions via MIL and can build a more discriminative multigrain representation than existing multiscale aerial scene classification methods [18], [20], [21], [46], it achieves the strongest performance.
2) Highlighting the key local regions is also quite important to enhance the aerial scene representation capability for the deep learning frameworks [2], [25], [28], [29], and this can also be one of the major reasons to account for the weak performance of GAN-based methods [60], [61].
3) As there are much more training samples in AID benchmark than in UCM, the gap of representation capability between traditional hand-crafted features and deep learning-based approaches becomes more obvious. In fact, it is a good example to illustrate that the traditional hand-crafted feature-based methods are far enough to depict the complexity of the aerial scenes.

3) Results and Comparison on NWPU: Table V lists the per-category classification results of our AGOS and other state-of-the-art approaches on NWPU benchmark. Several observations similar to the AID can be made.

1) Our AGOS outperforms all the compared state-of-the-art performance when the training ratios are both 10% and 20%. Its DenseNet-121 and ResNet-101 versions achieve the best and second best results on both settings,

TABLE IV

CLASSIFICATION ACCURACY OF OUR PROPOSED AGOS AND OTHER APPROACHES ON AID DATASET. RESULTS PRESENTED IN THE FORM OF "AVERAGE±DEVIATION" [17]; METRICS PRESENTED IN %; H, C, R, AND G DENOTE HAND-CRAFTED, CNN-, RNN-, AND GAN-BASED APPROACHES. VALUES IN BOLD AND IN BLUE DENOTE THE BEST AND SECOND BEST RESULTS

Method	Type	Training ratio	
		20%	50%
PLSA(SIFT) [17]	H, 2017	56.24±0.58	63.07±1.77
BoVW(SIFT) [17]	H, 2017	62.49±0.53	68.37±0.40
LDA(SIFT) [17]	H, 2017	51.73±0.73	68.96±0.58
AlexNet [17]	C, 2017	86.86±0.47	89.53±0.31
VGGNet-16 [17]	C, 2017	86.59±0.29	89.64±0.36
GoogLeNet [17]	C, 2017	83.44±0.40	86.39±0.55
ARCNet [25]	R, 2018	88.75±0.40	93.10±0.55
MARTA [60]	G, 2017	75.39±0.49	81.57±0.33
AGAN [61]	G, 2019	78.95±0.23	84.52±0.18
SPPNet [20]	C, 2017	87.44±0.45*	91.45±0.38*
TEXNet [89]	C, 2017	87.32±0.37	90.00±0.33
MSCP [18]	C, 2018	91.52±0.21	94.42±0.17
APNet [88]	C, 2019	88.56±0.29	92.15±0.29
MSDFF [47]	C, 2020	93.47±—-	96.74±—-
CADNet [48]	C, 2020	95.73±0.22	97.16±0.26
MIDCNet [2]	C, 2020	88.26±0.43	92.53±0.18
RANet [29]	C, 2020	88.12±0.43	92.35±0.19
GBNet [49]	C, 2020	92.20±0.23	95.48±0.12
MG-CAP [51]	C, 2020	93.34±0.18	96.12±0.12
MF^2Net [90]	C, 2020	93.82±0.26	95.93±0.23
DSENet [45]	C, 2021	94.02±0.21	94.50±0.30
MS2AP [46]	C, 2021	92.19±0.22	94.82±0.20
Contourlet CNN [52]	C, 2021	—-	96.65±0.24
LSENet [5]	C, 2021	94.07±0.19	95.82±0.19
MBLANet [50]	C, 2021	95.60±0.17	97.14±0.03
LiGNet [91]	C, 2021	94.17±0.25	96.19±0.28
DMSMIL [33]	C, 2021	93.98±0.17	95.65±0.22
STHP [53]	C, 2022	92.29±0.13	—-
SAGM [54]	C, 2022	95.44±0.25	97.08±0.17
DARTS [55]	C, 2022	—-	95.65±—-
LML [56]	C, 2022	94.71±0.17	96.86±0.11
GCSANet [57]	C, 2022	95.52±0.51	96.68±0.42
AGOS (ResNet-50)	C	94.99±0.24	97.01±0.18
AGOS (ResNet-101)	C	95.54±0.23	97.22±0.19
AGOS (DenseNet-121)	C	**95.81±0.25**	**97.43±0.21**

'—-': not reported, '*': not reported & conducted by us

TABLE V

OA OF THE PROPOSED AGOS AND OTHER APPROACHES ON NWPU DATASET. RESULTS PRESENTED IN THE FORM OF "AVERAGE±DEVIATION" [87]; METRICS PRESENTED IN %; H, C, R, AND G DENOTE HAND-CRAFTED, CNN-, RNN-, AND GAN-BASED APPROACHES, RESPECTIVELY. VALUES IN BOLD AND IN BLUE DENOTE THE BEST AND SECOND BEST RESULTS

Method	Type	Training ratio	
		10%	20%
BoVW(SIFT) [87]	H, 2017	41.72±0.21	44.97±0.28
AlexNet [87]	C, 2017	76.69±0.21	79.85±0.13
VGGNet-16 [87]	C, 2017	76.47±0.18	79.79±0.15
GoogLeNet [87]	C, 2017	76.19±0.38	78.48±0.26
MARTA [60]	G, 2017	68.63±0.22	75.03±0.28
AGAN [61]	G, 2019	72.21±0.21	77.99±0.19
SPPNet [20]	C, 2017	82.13±0.30*	84.64±0.23*
DCNN [28]	C, 2018	89.22±0.50	91.89±0.22
MSCP [18]	C, 2018	85.33±0.17	88.93±0.14
MSDFF [47]	C, 2020	91.56±—-	93.55±—-
CADNet [48]	C, 2020	92.70±0.32	94.58±0.26
MIDCNet [2]	C, 2020	85.59±0.26	87.32±0.17
RANet [29]	C, 2020	85.72±0.25	87.63±0.28
MG-CAP [51]	C, 2020	90.83±0.12	92.95±0.11
MF^2Net [90]	C, 2020	90.17±0.25	92.73 ±0.21
MS2AP [46]	C, 2021	87.91±0.19	90.98±0.21
Contourlet CNN [52]	C, 2021	85.93±0.51	89.57±0.45
LSENet [5]	C, 2021	91.93±0.19	93.14±0.15
MBLANet [50]	C, 2021	92.32±0.15	94.66 ±0.11
LiGNet [91]	C, 2021	90.23±0.13	93.25±0.12
DMSMIL [33]	C, 2021	91.93±0.16	93.05±0.14
SAGM [54]	C, 2022	92.35±0.17	94.84±0.09
DARTS [55]	C, 2022	—-	93.73±—-
LML [56]	C, 2022	92.67±0.15	94.73±0.11
GCSANet [57]	C, 2022	92.75±0.47	94.49±0.38
AGOS (ResNet-50)	C	92.35±0.17	94.28±0.16
AGOS (ResNet-101)	C	92.91±0.17	94.69±0.18
AGOS (DenseNet-121)	C	**93.04±0.35**	**94.91±0.17**

'—-': not reported, '*': not reported & conducted by us

while the performance of ResNet-50 version is competitive.

2) Generally speaking, those approaches highlighting the key local regions of an aerial scene [2], [25], [28], [29], [88] or building a multiscale convolutional feature representation tend to achieve a better performance [18], [20], [46].
3) The performance of GAN-based approaches [60], [61] degrades significantly when compared with other CNN-based methods on NWPU. Specifically, they are weaker than some CNN baselines such as VGGNet and GoogLeNet.

Moreover, the per-category classification accuracy under the training ratio of 10% and 20% is shown in Fig. 7(e) and (f). Most categories of the NWPU dataset are classified well. Similar to the discussion on AID, potential explanations of these outcomes include the following.

1) The difference of spatial resolution and object size is more varied in NWPU than in AID and UCM. Thus, the importance of both highlighting the key local regions and building more discriminative multigrain representation is critical for an approach to distinguish the aerial scenes of different categories. The weak performance of GAN-based methods can also be accounted that no effort has been investigated on either of the aforementioned two strategies, which is an interesting direction to explore in the future.
2) As our AGOS builds multigrain representations and highlights the key local regions, it is capable of distinguishing some scene categories that are varied a lot in terms of object sizes and spatial density. Thus, the experiments on all three benchmarks reflect that our AGOS is qualified to distinguish such scene categories.

E. Ablation Studies

Apart from the ResNet-50 baseline, our AGOS framework consists of an MGP module, an MBMIR module, and an SSF module. To evaluate the influence of each component on the classification performance, we conduct an ablation study on AID benchmark and the results are reported in Table VI. It can be seen in the table.

1) The performance gain led by MGP is about 1.26% and 1.36% if directly fused and then fed into the classification layer. Thus, more powerful representation learning strategies are needed for aerial scenes.

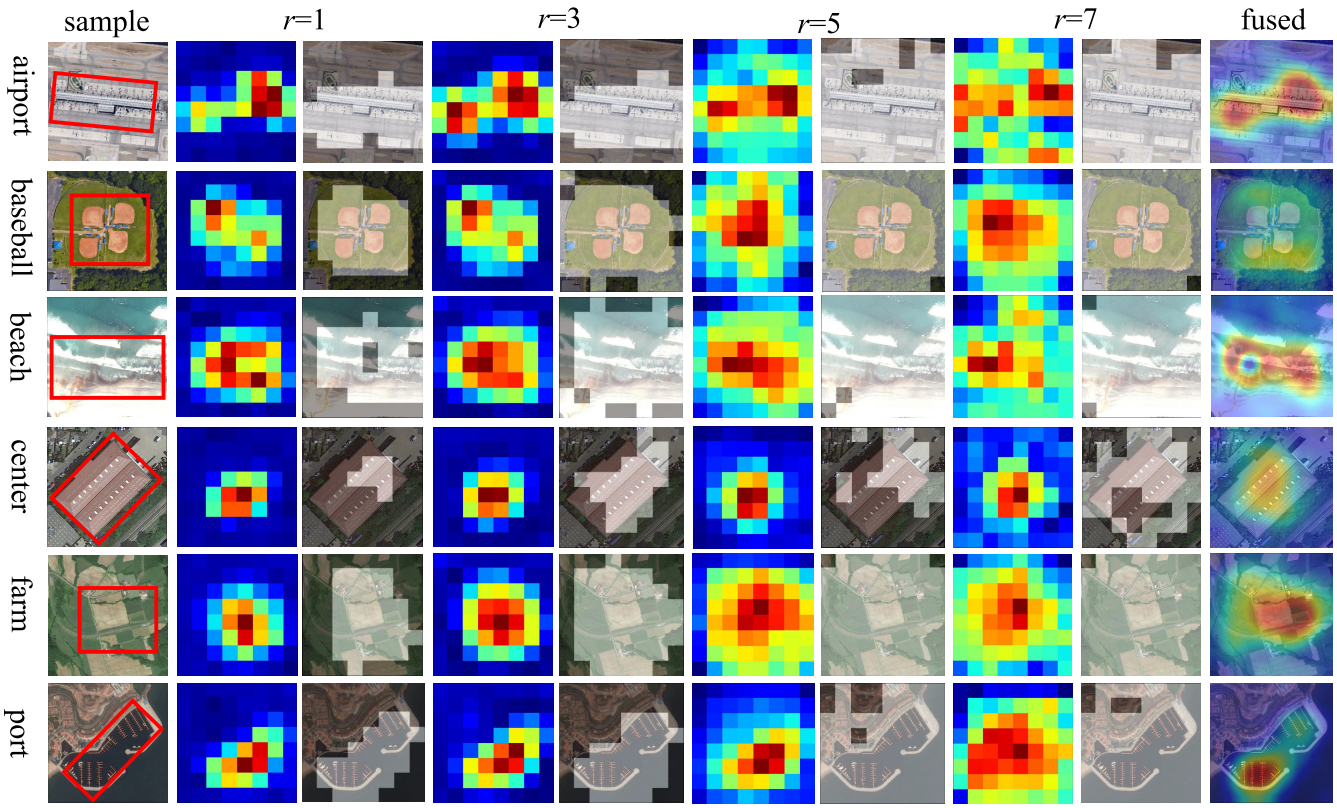

Fig. 8. Visualized feature responses by our AGOS (with ResNet-50 backbone) (denoted as fused) and the intermediate instance presentations are from four different grains where the dilated rate is 1, 3, 5, and 7 (denoted as $r = 1, 3, 5, 7$), respectively. For each image adjacent to the instance representation map, the activated instances are highlighted in white. Key local regions of the scene are labeled in red bounding boxes for reference. Heatmaps (fused) are generated from the average of instance representations from all grains and resized to the image size, and then normalized to [0, 255].

TABLE VI

ABLATION STUDY OF OUR AGOS (WITH RESNET-50 BACKBONE) ON AID DATASET; METRICS PRESENTED IN %; RESNET: BACKBONE RESNET-50; MGP: MGP MODULE; MBMIR MODULE; SSF MODULE; L_{CLS}: ONLY USE THE CLASSIFICATION TERM IN THE LOSS FUNCTION

Module				AID	
ResNet	MGP	MBMIR	SSF	20%	50%
✓				88.63±0.26	91.72±0.17
✓	✓			89.99±0.21	92.98±0.19
✓	✓	✓		94.16±0.24	96.20±0.16
✓	✓		✓	92.26±0.25	95.13±0.15
✓	✓	✓	L_{cls}	94.27±0.19	96.47±0.23
✓	✓	✓	✓	**94.99±0.24**	**97.01±0.18**

2) Our MBMIR module leads a performance gain of 4.17% and 3.22%, respectively. Its effectiveness can be explained from: 1) highlighting the key local regions in aerial scenes by using classic MIL formulation and 2) building more discriminative multigrain representation by extending MIL to the multigrain form.
3) Our SSF module improves the performance by about 1% in both the cases. This indicates that our bag scheme self-alignment strategy is effective to further refine the multigrain representation so that the representation from each grain focuses on the same bag scheme.

To sum up, an MGP serves as a basis in our AGOS to perceive the multigrain feature representation, and an MBMIR is the key component in our MBMIR which allows the entire feature representation learning under the MIL formulation, and the performance gain is the most. Finally, our SSF helps further refine the instance representations from different grains and allows the aerial scene representation more discriminative.

F. Generalization Ability

1) On Different Backbones: Table VII lists the classification performance, parameter number, and inference time of our AGOS framework when embedded into three commonly used backbones, that is, VGGNet [12], ResNet [11], and Inception [13], respectively. It can be seen that, on all three backbones, our AGOS framework leads to a significant performance gain while only increasing the parameter number and decreasing down the inference time slightly. The marginal increase of parameter number is quite interesting, as our AGOS removes the traditional fully connected layers in CNNs, which usually occupy a large number of parameters.

2) On Classification Task From Other Domains: Table VIII reports the performance of our AGOS framework on a medical image classification [92] and a texture classification [93] benchmark, respectively. The dramatic performance gain compared with the baseline on both benchmarks indicates that

TABLE VII

PERFORMANCE OF OUR AGOS ON DIFFERENT BACKBONES ON AID DATASET UNDER THE 50% TRAINING RATIO [17]; METRIC PRESENTED IN %; PARA. NUM.: PARAMETER NUMBERS; PRESENTED IN MILLION; FPS: FRAME PER SECOND

	OA	Para. num.	FPS
VGG-16	90.64±0.14	15.43	245.82
AGOS with VGG-16	**96.26±0.15**	19.94	227.48
ResNet-50	91.72±0.17	23.46	422.30
AGOS with ResNet-50	**97.01±0.18**	29.69	367.40
Inception-v2	91.40±0.19	6.64	704.22
AGOS with Inception-v2	**96.64±0.16**	8.79	652.35

TABLE VIII

PERFORMANCE OF OUR AGOS FRAMEWORK (WITH RESNET-50 BACKBONE) ON LARGE-SCALE ATTENTION BASED GLAUCOMA (LAG) [92] AND KYLBERG TEXTURE DATASET (KTD) DATASET [93]; BOTH BENCHMARKS REQUIRE THE FIVEFOLD CLASSIFICATION ACCURACY; METRIC PRESENTED IN %

	LAG	KTD
ResNet	91.75	91.74
AGOS with ResNet	**98.05**	**99.95**

TABLE IX

GENERALIZATION CAPABILITY OF OUR AGOS ON ZERO-SHOT REMOTE SENSING SCENE CLASSIFICATION BENCHMARK ZSRSSC [96]; THREE EXPERIMENTS ARE ALL UNDER GENERAL KNOWLEDGE WITH A SEEN/UNSEEN RATIO OF 40/30, 50/20, AND 60/10, RESPECTIVELY; OA IS PRESENTED IN %

	40/30	50/20	60/10
SAE [94]	0.052±0.014	0.095±0.016	0.167±0.041
AGOS-SAE	**0.061±0.012**	**0.107±0.019**	**0.184±0.020**
DMaP [95]	0.104±0.009	0.167±0.022	0.260±0.036
AGOS-DMaP	**0.116±0.013**	**0.184±0.018**	**0.281±0.027**

our AGOS has great generalization capability on other image recognition domains.

3) On Zero-Shot Scene Classification: Zero-shot remote sensing scene classification is a challenging topic and has drawn increasing attention in the past few years. Despite the essence difference that the pipeline of zero-shot classification is based on the alignment of high-dimensional feature vectors (e.g., 512-d) while the pipeline of our AGOS is based on the low-dimensional instance representation (i.e., equals to the semantic categories), we remove the MBMIR and SSF components and adapt rest parts that can be embedded into the zero-shot learning pipelines into the baseline of SAE [94] and DMaP [95]. Results are shown in Table IX.

It can be seen that under all three experimental settings, on both baselines, our MGP module from AGOS can boost the performance of zero-shot remote sensing scene classification.

4) On Potential Down-Stream Applications: The feature representation from scene recognition pipelines is capable of transferring into the down-stream applications such as geo-spatial object detection and land cover classification (pixel-level). However, most of these existing pipelines take the high-dimensional convolutional feature maps as input, but our AGOS generates low-dimensional instance representations, in which the dimension is equal to the category number and is highly compressed. Hence, our full AGOS can only be adapted to those down-stream application frameworks that take category-level feature maps as input, or only the components before instance aggregation can be embedded into these existing down-stream application frameworks.

Meanwhile, it remains an interesting topic to investigate how these low-dimensional representations can be adapted into the task-specific heads such as detection and segmentation that demand high-dimensional representations as input.

TABLE X

COMPARISON OF OUR AGOS (WITH RESNET-50 BACKBONE) WITH SOME CLASSIC SOLUTIONS ON ALIGNING THE AERIAL SCENE SCHEME ON AID BENCHMARK; MEAN: MEAN OPERATION; MAX: MAX SELECTION; MV; LSS METHOD; METRIC PRESENTED IN %

Method	AID 50%
Mean	96.20±0.16
Max	95.94±0.21
MV	96.43±0.17
LS	96.38±0.19
AGOS (ours)	**97.01±0.18**

G. Discussion on Bag Scheme Alignment

Generally speaking, the motivation of our SSF module is to learn a discriminative aerial scene representation from multigrain instance-level representations. However, in classic machine learning and statistical data processing, there are also some solutions that either select or fit an optimal outcome from multiple representations. Hence, it would be quite interesting to compare the impact of our SSF and these classic solutions.

To this end, four classic implementations on our bag probability distributions from multigrain instance representations, namely, naive mean (Mean) operation, naive max (Max) selection, majority vote (MV), and least squares (LSs) method, are tested and compared based on the AID dataset under the 50% training ratio. Table X lists all these results. Note that, by using naive mean, the entire framework degrades to the third case in our ablation studies (in Table VI).

It can be seen that our SSF achieves the best performance while: 1) max selection shows apparent performance decline and 2) other three solutions, namely mean operation, MV, and LS, do not show much performance difference.

To better understand how these methods influence the scene scheme alignment, Fig. 9 offers the visualized covariance matrix of the bag probability distributions from all the test samples. Generally speaking, a good scene representation will have higher response on the diagonal region while the response from other regions should be as low as possible. It is clearly seen that our SSF has the best discrimination capability, while for the other solutions, some confusion between bag probability distributions of different categories always happens.

The explanation may lie in the following aspects: 1) our SSF aligns the scene scheme from both representation learning and loss optimization, and thus leads to more performance gain; 2) naive average on these multigrain instance representations already achieves an acceptable scene scheme representation, and thus leaves very little space for other solutions such as LS and MV to improve; and 3) max selection itself may lead

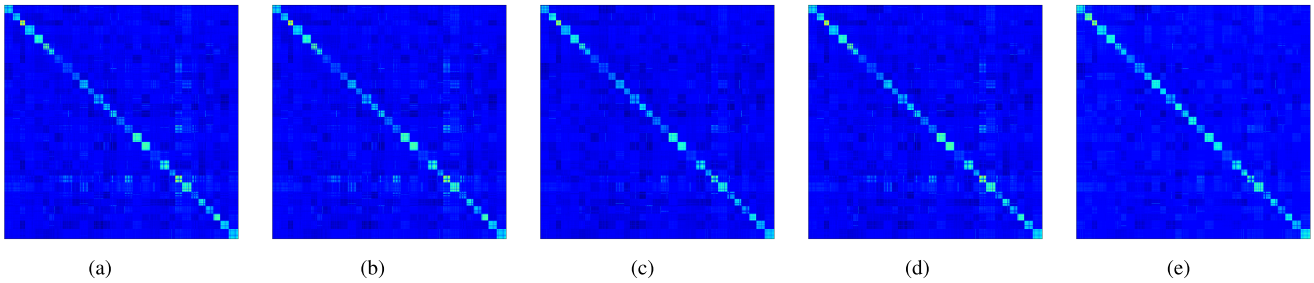

Fig. 9. Visualized covariance matrix of the bag probability distribution after scene scheme alignment, processed by mean selection (a), max selection (b), MV (c), LS method (d), and our AGOS (e). Ideally, the covariance matrix of bag probability distribution should have high responses in the diagonal region and no responses in other regions. (a) Mean selection. (b) Max selection. (c) MV. (d) LS. (e) AGOS (ours).

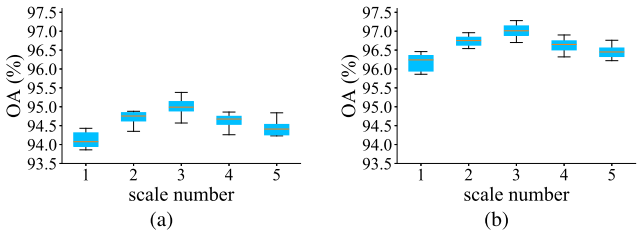

Fig. 10. Performance change influenced by the grain number in our AGOS (with ResNet-50 backbone) on AID 20% (a) and 50% (b). (a) AID 20%. (b) AID 50%.

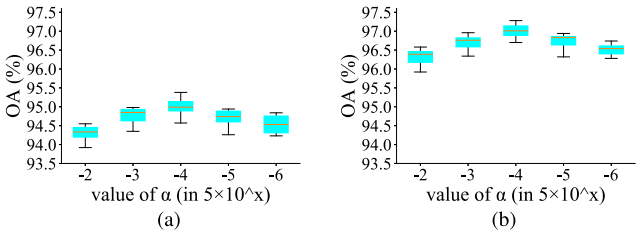

Fig. 11. Performance change influenced by the hyperparameter α in our AGOS (with ResNet-50 backbone) on AID 20% (a) and 50% (b); α presented in 5×10^x, and $x = -2, -3, -4$ and -5. (a) AID 20%. (b) AID 50%.

to more variance on bag probability prediction and thus the performance declines.

H. Sensitivity Analysis

1) Influence of Grain Number: Fig. 10 lists the performance when the grain number T in our AGOS changes. It can be seen that when there are about three or four grains, the classification accuracy reaches its peak. After that, the classification performance slightly declines. This implies that the combined utilization of convolutional features when the dilated rate is 1, 3, and 5 is most discriminative in our AGOS. When there are too many grains, the perception field becomes too large and the scene representation becomes less discriminative. Also, when the grain number is little, the representation is not qualified enough to completely depict the semantic representation where the key objects vary greatly in sizes.

On the other hand, the visualized samples in Fig. 8 also reveal that when the dilation rate in our MGP is too small, the instance representation tends to focus on a small local region of an aerial scene. In contrast, when the dilation rate is too large, the instance representation activates too many local regions irrelevant to the scene scheme. Thus, the importance of our scene scheme self-align strategy reflects, as it helps the representation from different grains to align to the same scene scheme and refines the activated key local regions. Note that, for further investigating the interpretation capability of these patches and the possibility for weakly supervised localization task, details can be found in [66].

2) Influence of Hyperparameter α: Fig. 11 shows the classification accuracy fluctuation when the hyperparameter α in our loss function changes. It can be seen that the performance of our AGOS is stable when α changes. However, when it is too large, the performance shows an obvious decline. When it is too small, the performance degradation is slight.

TABLE XI

COMPARISON OF OUR DDC ON THE CASES WHEN NOT USING DIFFERENTIAL OPERATION (D#DC), NOT USING DILATED CONVOLUTION (DD#C) AND NOT USING EITHER DIFFERENTIAL OPERATION AND DILATED CONVOLUTION (C) ON AID BENCHMARK WITH RESNET-50 BACKBONE; METRIC IN %

	AID 20%	AID 50%
C	90.56±0.25	93.45±0.19
DD#C	91.34±0.21	94.36±0.16
D#DC	93.85±0.22	96.18±0.17
Dense	94.95±0.19	96.98±0.15
DDC	**94.99±0.24**	**97.01±0.18**

3) Influence of Differential Dilated Convolution: Table XI lists the classification performance when every component of DDC in our MGP is used or not used, and the case when all the representation pairs are summed (denoted as Dense). It can be seen that both the differential operation (D#dc) and the dilated convolution (DD#C) lead to an obvious performance gain for our AGOS. Generally, the performance gain led by the dilated convolution is higher than the differential operation, as it enlarges the receptive field of a deep learning model and thus enhances the feature representation more significantly.

Also, the case when using dense prediction (Dense) shows a very close but slightly lower performance when compared with our case only using the neighbor representations (DDC). The explanation may lie in that the sum of all the representation

pairs may contain more redundant information in the final fused representation. Hence, the feature representation itself may become less discriminative.

V. Conclusion

In this article, we propose an AGOS framework for aerial scene classification. *To the best of our knowledge*, it is the first effort to extend the classic MIL into deep multigrain MIL formulation. The effectiveness of our AGOS lies in threefold: 1) the MIL formulation allows the framework to highlight the key local regions in determining the scene category; 2) the multigrain multi-instance representation is more capable of depicting the complicated aerial scenes; and 3) the bag scheme self-alignment strategy allows the instance representation from each grain to focus on the same bag category. Experiments on three aerial scene classification datasets demonstrate the effectiveness of our AGOS and its generalization capability.

As our AGOS is capable of building discriminative scene representation and highlighting the key local regions precisely, our future work includes transferring our AGOS framework to other tasks such as object localization, detection, and segmentation especially under the weakly supervised scenarios.

Acknowledgment

The authors would like to thank the editors and the anonymous reviewers, whose insightful suggestions and comments significantly improved their manuscript.

References

[1] G.-S. Xia *et al.*, "DOTA: A large-scale dataset for object detection in aerial images," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 3974–3983.

[2] Q. Bi, K. Qin, Z. Li, H. Zhang, K. Xu, and G.-S. Xia, "A multiple-instance densely-connected convnet for aerial scene classification," *IEEE Trans. Image Process.*, vol. 29, pp. 4911–4926, 2020.

[3] J. Ding, N. Xue, Y. Long, G.-S. Xia, and Q. Lu, "Learning RoI transformer for oriented object detection in aerial images," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 2849–2858.

[4] L. Mou, Y. Hua, and X. X. Zhu, "A relation-augmented fully convolutional network for semantic segmentation in aerial scenes," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 12416–12425.

[5] Q. Bi, K. Qin, H. Zhang, and G.-S. Xia, "Local semantic enhanced ConvNet for aerial scene recognition," *IEEE Trans. Image Process.*, vol. 30, pp. 6498–6511, 2021.

[6] X.-Y. Tong *et al.*, "Land-cover classification with high-resolution remote sensing images using transferable deep models," *Remote Sens. Environ.*, vol. 237, Feb. 2020, Art. no. 111322.

[7] Q. Bi, K. Qin, H. Zhang, Y. Zhang, and K. Xu, "A multi-scale filtering building index for building extraction in very high-resolution satellite imagery," *Remote Sens.*, vol. 11, no. 5, p. 482, 2019.

[8] D. Hong, N. Yokoya, G.-S. Xia, J. Chanussot, and X. X. Zhu, "X-ModalNet: A semi-supervised deep cross-modal network for classification of remote sensing data," *ISPRS J. Photogramm. Remote Sens.*, vol. 167, pp. 12–23, Sep. 2020.

[9] Y. Zhang, K. Qin, Q. Bi, W. Cui, and G. Li, "Landscape patterns and building functions for urban land-use classification from remote sensing images at the block level: A case study of Wuchang District, Wuhan, China," *Remote Sens.*, vol. 12, no. 11, p. 1831, Jun. 2020.

[10] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 4700–4708.

[11] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

[12] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. Int. Conf. Learn. Represent.*, 2015, pp. 1–14.

[13] C. Szegedy *et al.*, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1–9.

[14] W. Ji *et al.*, "Calibrated RGB-D salient object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 9471–9481.

[15] J. Li *et al.*, "Joint semantic mining for weakly supervised RGB-D salient object detection," in *Proc. NIPS*, vol. 34, 2021, pp. 1–15.

[16] J. Ding *et al.*, "Object detection in aerial images: A large-scale benchmark and challenges," *IEEE Trans. Pattern Anal. Mach. Intell.*, early access, Oct. 6, 2021, doi: 10.1109/TPAMI.2021.3117983.

[17] G.-S. Xia *et al.*, "AID: A benchmark data set for performance evaluation of aerial scene classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 7, pp. 3965–3981, Jul. 2017.

[18] N. He, L. Fang, S. Li, A. Plaza, and J. Plaza, "Remote sensing scene classification using multilayer stacked covariance pooling," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 12, pp. 6899–6910, Dec. 2018.

[19] F. Hu, G.-S. Xia, J. Hu, and L. Zhang, "Transferring deep convolutional neural networks for the scene classification of high-resolution remote sensing imagery," *Remote Sens.*, vol. 7, no. 11, pp. 14680–14707, Nov. 2015.

[20] X. Han, Y. Zhong, L. Cao, and L. Zhang, "Pre-trained alexnet architecture with pyramid pooling and supervision for high spatial resolution remote sensing image scene classification," *Remote Sens.*, vol. 9, no. 8, p. 848, 2017.

[21] E. Li, J. Xia, P. Du, C. Lin, and A. Samat, "Integrating multilayer features of convolutional neural networks for remote sensing scene classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 10, pp. 5653–5665, Oct. 2017.

[22] Z. Zheng, Y. Zhong, J. Wang, and A. Ma, "Foreground-aware relation network for geospatial object segmentation in high spatial resolution remote sensing imagery," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 4096–4105.

[23] M. Ilse, J. Tomczak, and M. Welling, "Attention-based deep multiple instance learning," in *Proc. Int. Conf. Mach. Learn.*, 2018, pp. 2127–2136.

[24] B. Wieland and M. Bethge, "Approximating CNNs with bag-of-local-features models works surprisingly well on ImageNet," in *Proc. Int. Conf. Learn. Represent.*, 2019, pp. 1–15.

[25] Q. Wang, S. Liu, J. Chanussot, and X. Li, "Scene classification with recurrent attention of VHR remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 2, pp. 1155–1167, Feb. 2018.

[26] Q. Zhu, Y. Zhong, L. Zhang, and D. Li, "Adaptive deep sparse semantic modeling framework for high spatial resolution image scene classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 10, pp. 6180–6195, Oct. 2018.

[27] O. A. B. Penatti, K. Nogueira, and J. A. Dos Santos, "Do deep features generalize from everyday objects to remote sensing and aerial scenes domains?" in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2015, pp. 44–51.

[28] G. Cheng, C. Yang, X. Yao, L. Guo, and J. Han, "When deep learning meets metric learning: Remote sensing image scene classification via learning discriminative CNNs," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 5, pp. 2811–2821, May 2018.

[29] Q. Bi, K. Qin, H. Zhang, Z. Li, and K. Xu, "RADC-net: A residual attention based convolution network for aerial scene classification," *Neurocomputing*, vol. 377, pp. 345–359, Feb. 2020.

[30] T. G. Dietterich, R. H. Lathrop, and T. Lozano-Pérez, "Solving the multiple instance problem with axis-parallel rectangles," *Artif. Intell.*, vol. 89, nos. 1–2, pp. 31–71, Jan. 1997.

[31] O. Maron and A. Ratan, "Multiple-instance learning for natural scene classification," in *Proc. Int. Conf. Mach. Learn.*, 1998, pp. 341–349.

[32] Y. Pang, Y. Li, J. Shen, and L. Shao, "Towards bridging semantic gap to improve semantic segmentation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 4230–4239.

[33] B. Zhou, J. Yi, and Q. Bi, "Differential convolution feature guided deep multi-scale multiple instance learning for aerial scene classification," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Jun. 2021, pp. 4595–4599.

[34] J. Y. Choi, Y. M. Ro, and K. N. Plataniotis, "Color local texture features for color face recognition," *IEEE Trans. Image Process.*, vol. 21, no. 3, pp. 1366–1380, Mar. 2012.

[35] B. Luo, J. F. Aujol, Y. Gousseau, and S. Ladjal, "Indexing of satellite images with different resolutions by wavelet features," *IEEE Trans. Image Process.*, vol. 17, no. 8, pp. 1465–1472, Aug. 2008.

[36] A. Satpathy, X. Jiang, and H.-L. Eng, "LBP-based edge-texture features for object recognition," *IEEE Trans. Image Process.*, vol. 23, no. 5, pp. 1953–1964, May 2014.

[37] T. Ojala, M. Pietikainen, and T. Maenpaa, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 7, pp. 971–987, Jul. 2002.

[38] Y. Zhong, Q. Zhu, and L. Zhang, "Scene classification based on the multifeature fusion probabilistic topic model for high spatial resolution remote sensing imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 11, pp. 6207–6222, Nov. 2015.

[39] T. Hofmann, "Unsupervised learning by probabilistic latent semantic analysis," *Mach. Learn.*, vol. 42, no. 1, pp. 177–196, Jan. 2001.

[40] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent Dirichlet allocation," *J. Mach. Learn. Res.*, vol. 3, pp. 993–1022, Mar. 2003.

[41] B. Zhao, Y. Zhong, G.-S. Xia, and L. Zhang, "Dirichlet-derived multiple topic scene classification model for high spatial resolution remote sensing imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 4, pp. 2108–2123, Apr. 2016.

[42] X. X. Zhu *et al.*, "Deep learning in remote sensing: A comprehensive review and list of resources," *IEEE Geosci. Remote Sens. Mag.*, vol. 5, no. 4, pp. 8–36, Dec. 2017.

[43] G. Cheng, X. Xie, J. Han, L. Guo, and G.-S. Xia, "Remote sensing image scene classification meets deep learning: Challenges, methods, benchmarks, and opportunities," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 13, no. 99, pp. 3735–3756, Jun. 2020.

[44] K. Nogueira, O. A. B. Penatti, and J. A. Dos Santos, "Towards better exploiting convolutional neural networks for remote sensing scene classification," *Pattern Recognit.*, vol. 61, pp. 539–556, Jan. 2017.

[45] X. Wang, S. Wang, C. Ning, and H. Zhou, "Enhanced feature pyramid network with deep semantic embedding for remote sensing scene classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 9, pp. 7918–7932, Sep. 2021.

[46] Q. Bi, H. Zhang, and K. Qin, "Multi-scale stacking attention pooling for remote sensing scene classification," *Neurocomputing*, vol. 436, pp. 147–161, May 2021.

[47] W. Xue, X. Dai, and L. Liu, "Remote sensing scene classification based on multi-structure deep features fusion," *IEEE Access*, vol. 8, pp. 28746–28755, 2020.

[48] W. Tong, W. Chen, W. Han, X. Li, and L. Wang, "Channel-attention-based densenet network for remote sensing image scene classification," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 13, pp. 4121–4132, 2020.

[49] H. Sun, S. Li, X. Zheng, and X. Lu, "Remote sensing scene classification by gated bidirectional network," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 1, pp. 82–96, Jan. 2020.

[50] S.-B. Chen, Q.-S. Wei, W.-Z. Wang, J. Tang, B. Luo, and Z.-Y. Wang, "Remote sensing scene classification via multi-branch local attention network," *IEEE Trans. Image Process.*, vol. 31, pp. 99–109, 2022.

[51] S. Wang, Y. Guan, and L. Shao, "Multi-granularity canonical appearance pooling for remote sensing scene classification," *IEEE Trans. Image Process.*, vol. 29, pp. 5396–5407, 2020.

[52] M. Liu, L. Jiao, X. Liu, L. Li, F. Liu, and S. Yang, "C-CNN: Contourlet convolutional neural networks," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 6, pp. 2636–2649, Jun. 2021.

[53] X. Gu, C. Zhang, Q. Shen, J. Han, P. P. Angelov, and P. M. Atkinson, "A self-training hierarchical prototype-based ensemble framework for remote sensing scene classification," *Inf. Fusion*, vol. 80, pp. 179–204, Apr. 2022.

[54] Q. Meng, M. Zhao, L. Zhang, W. Shi, C. Su, and L. Bruzzone, "Multilayer feature fusion network with spatial attention and gated mechanism for remote sensing scene classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, pp. 1–5, 2022.

[55] J. Shen, B. Cao, C. Zhang, R. Wang, and Q. Wang, "Remote sensing scene classification based on attention-enabled progressively searching," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–13, 2022.

[56] X. Chen, X. Zheng, Y. Zhang, and X. Lu, "Remote sensing scene classification by local–global mutual learning," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, pp. 1–5, 2022.

[57] W. Chen, S. Ouyang, W. Tong, X. Li, X. Zheng, and L. Wang, "GCSANet: A global context spatial attention deep learning network for remote sensing scene classification," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 15, pp. 1150–1162, 2022.

[58] G. Cheng, J. Han, L. Guo, Z. Liu, S. Bu, and J. Ren, "Effective and efficient midlevel visual elements-oriented land-use classification using VHR remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 8, pp. 4238–4249, Aug. 2015.

[59] F. Zhang, B. Du, and L. Zhang, "Saliency-guided unsupervised feature learning for scene classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 4, pp. 2175–2184, Apr. 2015.

[60] D. Lin, K. Fu, Y. Wang, G. Xu, and X. Sun, "MARTA GANs: Unsupervised representation learning for remote sensing image classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 14, no. 11, pp. 2092–2096, Nov. 2017.

[61] Y. Yu, X. Li, and F. Liu, "Attention GANs: Unsupervised deep feature learning for aerial scene classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 1, pp. 519–531, Jan. 2020.

[62] P. Deng, K. Xu, and H. Huang, "When CNNs meet vision transformer: A joint framework for remote sensing scene classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, pp. 1–5, 2022.

[63] J. Zhang, H. Zhao, and J. Li, "TRS: Transformers for remote sensing scene classification," *Remote Sens.*, vol. 13, no. 20, p. 4143, Oct. 2021.

[64] Y. Bazi, L. Bashmal, M. M. A. Rahhal, R. A. Dayil, and N. A. Ajlan, "Vision transformers for remote sensing image classification," *Remote Sens.*, vol. 13, no. 3, p. 516, 2021.

[65] X. Shi, F. Xing, Y. Xie, Z. Zhang, L. Cui, and L. Yang, "Loss-based attention for deep multiple instance learning," in *Proc. AAAI Conf. Artif. Intell.*, 2020, pp. 5742–5749.

[66] X. Shi *et al.*, "Loss-based attention for interpreting image-level prediction of convolutional neural networks," *IEEE Trans. Image Process.*, vol. 30, pp. 1662–1675, 2021.

[67] X. Wang, Y. Yan, P. Tang, X. Bai, and W. Liu, "Revisiting multiple instance neural networks," *Pattern Recognit.*, vol. 74, pp. 15–24, Feb. 2018.

[68] K. M. He, X. Y. Zhang, S. Q. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 9, pp. 1904–1916, Jun. 2015.

[69] T.-Y. Lin, P. Dollar, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2117–2125.

[70] M.-L. Zhang and Z.-H. Zhou, "Improve multi-instance neural networks through feature selection," *Neural Process. Lett.*, vol. 19, no. 1, pp. 1–10, 2004.

[71] P. O. Pinheiro and R. Collobert, "From image-level to pixel-level labeling with convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1713–1721.

[72] P. Tang, X. Wang, B. Feng, and W. Liu, "Learning multi-instance deep discriminative patterns for image classification," *IEEE Trans. Image Process.*, vol. 26, no. 7, pp. 3385–3396, Jul. 2017.

[73] X. Wang, B. Wang, X. Bai, W. Liu, and Z. Tu, "Max-margin multiple-instance dictionary learning," in *Proc. Int. Conf. Mach. Learn.*, 2013, pp. 846–854.

[74] Q. Wang, Y. Yuan, P. Yan, and X. Li, "Saliency detection by multiple-instance learning," *IEEE Trans. Cybern.*, vol. 43, no. 2, pp. 660–672, Apr. 2013.

[75] D. Zhang, D. Meng, and J. Han, "Co-saliency detection via a self-paced multiple-instance learning framework," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 5, pp. 865–878, May 2016.

[76] X. Liu *et al.*, "Deep multiple instance learning-based spatial–spectral classification for PAN and MS imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 1, pp. 461–473, Jan. 2018.

[77] C. Wang, K. Huang, W. Ren, J. Zhang, and S. Maybank, "Large-scale weakly supervised object localization via latent category learning," *IEEE Trans. Image Process.*, vol. 24, no. 4, pp. 1371–1385, Apr. 2015.

[78] W. Zhu, Q. Lou, Y. Vang, and X. Xie, "Deep multi-instance networks with sparse label assignment for whole mammogram classification," in *Proc. Int. Conf. Med. Image Comput. Comput. Assist. Interv.*, 2017, pp. 603–611.

[79] L. Mou and X. X. Zhu, "Vehicle instance segmentation from aerial image and video using a multitask learning residual fully convolutional network," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 11, pp. 6699–6711, Nov. 2018.

[80] X. Wang, Z. Zhu, C. Yao, and X. Bai, "Relaxed multiple-instance SVM with application to object discovery," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1224–1232.

[81] P. Tang, X. Wang, X. Bai, and W. Liu, "Multiple instance detection network with online instance classifier refinement," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2843–2851.

[82] S. Sabato and N. Tishby, "Multi-instance learning with any hypothesis class," *J. Mach. Learn. Res.*, vol. 13, no. 1, pp. 2999–3039, 2012.
[83] T. Alessandro, J. Manfred, and F. Paolo, "Learning and interpreting multi-multi-instance learning networks," *J. Mach. Learn. Res.*, vol. 21, no. 193, pp. 1–60, 2020.
[84] Y. Hua, L. Mou, and X. X. Zhu, "Relation network for multilabel aerial image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 7, pp. 4558–4572, Jul. 2020.
[85] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 4, pp. 834–848, Apr. 2017.
[86] Y. Yang and S. Newsam, "Geographic image retrieval using local invariant features," *IEEE Trans. Geosci. Remote Sens.*, vol. 51, no. 2, pp. 818–832, Feb. 2013.
[87] G. Cheng, J. Han, and X. Lu, "Remote sensing image scene classification: Benchmark and state of the art," *Proc. IEEE*, vol. 105, no. 10, pp. 1865–1883, Oct. 2017.
[88] Q. Bi, K. Qin, H. Zhang, J. Xie, Z. Li, and K. Xu, "APDC-Net: Attention pooling-based convolutional network for aerial scene classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 17, no. 9, pp. 1603–1607, Sep. 2020.
[89] R. M. Anwer, F. S. Khan, J. van de Weijer, M. Molinier, and J. Laaksonen, "Binary patterns encoded convolutional neural networks for texture recognition and remote sensing scene classification," *ISPRS J. Photogramm. Remote Sens.*, vol. 138, pp. 74–85, Apr. 2018.
[90] K. Xu, H. Huang, Y. Li, and G. Shi, "Multilayer feature fusion network for scene classification in remote sensing," *IEEE Geosci. Remote Sens. Lett.*, vol. 17, no. 11, pp. 1894–1898, Nov. 2020.
[91] C. Xu, G. Zhu, and J. Shu, "A lightweight intrinsic mean for remote sensing classification with lie group kernel function," *IEEE Geosci. Remote Sens. Lett.*, vol. 18, no. 10, pp. 1741–1745, Oct. 2021.
[92] L. Li, M. Xu, X. Wang, L. Jiang, and H. Liu, "Attention based glaucoma detection: A large-scale database and CNN model," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 10571–10580.
[93] G. Kylberg, "The kylberg texture dataset V. 1.0," Centre Image Anal., Swedish Univ. Agricult. Sci. Uppsala, Sweden, Tech. Rep., 2011, pp. 1–5.
[94] E. Kodirov, T. Xiang, and S. Gong, "Semantic autoencoder for zero-shot learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 3174–3183.
[95] Y. Li, D. Wang, H. Hu, Y. Lin, and Y. Zhuang, "Zero-shot recognition using dual visual-semantic mapping paths," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 3279–3287.
[96] Y. Li, Z. Zhu, J.-G. Yu, and Y. Zhang, "Learning deep cross-modal embedding networks for zero-shot remote sensing image scene classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 12, pp. 10590–10603, Dec. 2021.

Qi Bi (Student Member, IEEE) received the B.S. and M.S. degrees from Wuhan University, Wuhan, China, in 2017 and 2020, respectively. He is currently pursuing the Ph.D. degree with the Computer Vision Research Group, University of Amsterdam, Amsterdam, The Netherlands.

He served as a Research Intern with the Jarvis Laboratory, Tencent, Shenzhen, China, for four months in 2020. His research interests include image understanding, deep learning, and the corresponding applications in remote sensing and medical imaging.

Beichen Zhou received the B.S. degree from Wuhan University, Wuhan, China, in 2022, where he is currently pursuing the M.S. degree.

His research interests include aerial image object detection and deep learning.

Kun Qin received the B.S. degree from Lanzhou University, Lanzhou, China, in 1995, the M.E. degree from Nanjing Normal University, Nanjing, China, in 1998, and the Ph.D. degree from Wuhan University, Wuhan, China, in 2004.

He is currently a Professor with Wuhan University. His research interests include spatial-temporal data mining and intelligence spatial information processing.

Qinghao Ye received the B.S. degree from Hangzhou Dianzi University, Hangzhou, China, in 2020, and the M.S. degree from the University of California, San Diego, CA, USA, in 2021.

In 2021, he served as a Research Intern with the DAMO Academy, Alibaba Group, Hangzhou, for three months. His research interests include video understanding, deep learning, medical image analysis, and the application of multimodalities in video.

Gui-Song Xia (Senior Member, IEEE) received the Ph.D. degree in image processing and computer vision from CNRS LTCI, Telecom ParisTech, Paris, France, in 2011.

From 2011 to 2012, he was a Post-Doctoral Researcher with the Centre de Recherche en Mathematiques de la Decision, CNRS, Paris-Dauphine University, Paris, for one-and-half years. He also worked as a Visiting Scholar with DMA, Ecole Normale Superieure (ENS-Paris), for two months in 2018. He is currently working as a Full Professor of computer vision and photogrammetry with Wuhan University. He is also a Guest Professor with the Future Lab AI4EO, Technical University of Munich (TUM), Munich, Germany. His research interests include mathematical modeling of images and videos, structure from motion, perceptual grouping, and remote sensing image understanding. He serves on the Editorial Boards of the journals the *ISPRS Journal of Photogrammetry and Remote Sensing*, *Pattern Recognition*, *Signal Processing: Image Communication*, *EURASIP Journal on Image and Video Processing*, *Journal of Remote Sensing*, and *Frontiers in Computer Science: Computer Vision*.