Received April 14, 2021, accepted May 7, 2021, date of publication May 19, 2021, date of current version May 28, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3081922

MGFN: A Multi-Granularity Fusion Convolutional Neural Network for Remote Sensing Scene Classification

ZHIGUO ZENG[1,2], XIHONG CHEN[1], AND ZHIHUA SONG[3]
[1]Air and Missile Defense College, Air Force Engineering University, Xi'an 710051, China
[2]Graduate School, Hunan University of Science and Technology, Xiangtan 411201, China
[3]College of Equipment Management and UAV Engineering, Air Force Engineering University, Xi'an 710051, China

Corresponding author: Zhihua Song (szhele@163.com)

This work was supported in part by the National Natural Science Foundation of China under Grant 72071209, in part by the Natural Science Foundation of Shaanxi Province under Grant 2021JQ-367, and in part by the grants from the Air Force Engineering University.

ABSTRACT Convolutional neural networks (CNNs) have been successfully used in remote sensing scene classification and identification due to their ability to capture deep spatial feature representations. However, the performance of deep models inevitably encounters a bottleneck when multimodality-dominated scene classification rather than single-modality-dominated scene classification is performed, due to the high similarity among different categories. In this study, we propose a novel multi-granularity fusion convolutional neural network (MGFN) to automatically capture the latent ontological features of remote sensing images. We firstly design a multigranularity module that can progressively crop input images to learn multigrained features, which can describe images to different degrees. Based on a comparison of different granularities, we then design a maxout-based module to learn the corresponding Gaussian covariance matrices of different granularities, which can extract second-order features to express the latent ontological essence of inputs and select the most distinguished inputs. We thirdly provide an adaptive fusion module to fuse all features via normalization to combine features of different degrees using the adaptive fused module. Finally, an SVM classifier is used to classify the fused matrix of every input image. Extensive experimentation and evaluations, particularly for multimodality-dominated scenes, demonstrate that the proposed network can achieve promising results for public remote sensing datasets.

INDEX TERMS Convolutional neural network, multi-granularity fusion, Gaussian covariance matrix, remote sensing scene classification.

I. INTRODUCTION

Remote sensing scene classification is one of the most researched areas and challenging topics in the geoscience and remote sensing community, since it is a process of classifying remotely sensed images into discrete sets of land use and land cover categories with semantic meanings [1]–[4]. Characterized by their rich and detailed spatial information, remote sensing scene images allow effective discrimination of objects by capturing subtle discrepancies from the contiguous shape of signatures associated with their pixels [5]. To extract features of different objects, a variety of handcrafted and learning-based classification algorithms have been successfully designed in recent [6]–[9]. Of the various classification algorithms, deep convolutional neural networks (CNNs) have been used extensively and typically yield high classification performances.

A CNN [10]–[12], attempts to express high level features, and has been acknowledged as the most successful deep learning model for remote sensing scene classification [5], [13]–[16] [17], [18]. Penatti assessed the generalization power of CNNs using pretrained CNNs as feature extractors to classify remote sensing images [14]. Jia used an off-the-shelf CNN model to extract high-dimensional features and fine-tuned the CNN model with the target dataset [5]. Hu surveyed how to apply pretrained CNNs to remote sensing scene classification [19]. Zhu provided a tutorial about deep learning-based remote sensing data analysis [20]. Nogueira analysed the performances of CNNs under three strategies: full training, fine tuning and using CNNs as feature

The associate editor coordinating the review of this manuscript and approving it for publication was Weipeng Jing.

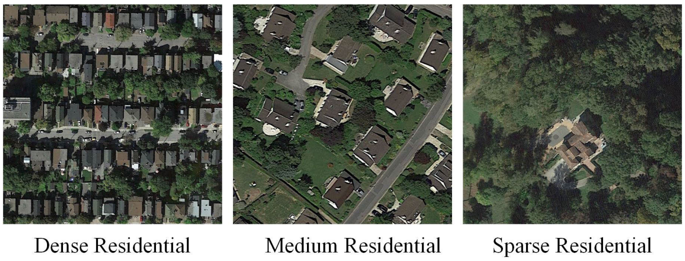

FIGURE 1. Residential images with different densities selected from different categories in the AID dataset, are composed of the same two modalities: houses and trees.

extractors [1]. Cheng proposed a method named the bag of convolutional features to encode convolutional feature descriptors [21]. Yuan attempted to encode deep features via the locality-constrained affine subspace coding method [22]. The primary strategy of these methods is to apply a pretrained CNN to target remote sensing scene images or fine-tune the pretrained CNN model with the target dataset. However, most of these CNN models, either transferring or not, are designed and used for single-modality-dominated scenes. The ability to identify materials in multimodality-dominated scenes that have similar objects and cannot be accurately classified by only a single modality remains limited.

Considering the latent ontology of remote sensing images, two primary challenges must be resolved [4]. The first problem is the visual-semantic discrepancy that is caused by a lack of alignment between the hand-crafted or learned features of an image and its corresponding semantic labels. The classification of remote sensing images that cover a large geographic area with significant unstructured information requires different levels of annotation to express the latent ontological essence. For example, in the NWPU-RESISC45 dataset [21], the category Airport may be composed of airplanes and runways, and similarly, railways and railway stations may belong to the category Railway, while bridges may be categorized as freeways. Specifically, airports, railways and freeways may come from the transportation category. Most relationships in these images can be disassembled into three levels: the superordinate level (e.g., transportation), the basic level (e.g., airport and railway) and the subordinate level (e.g., airplane and runway) [23], [24]. Therefore, classifying the subordinate or basic level is relatively easy, while more discriminative features are required to recognize superordinate-level objects. Currently, most deep learning-based classification methods can learn high-level features but cannot incorporate them with high-level semantic meanings in category labels because most remote sensing datasets lack well-constructed ontological structures.

The second challenge arises from the variances that naturally appear in different categories of the same dataset. Specifically, there are two major variances that must be considered: intraclass diversity and interclass similarity. Remote-sensing scene images are more easily classified with higher intraclass diversity and interclass similarity, and vice versa. However, real data typically exhibits the opposite characteristics. For example, in the AID dataset [25], tthree categories (Dense Residential, Medium Residential and Spars Residential) are all composed of the same two modalities (houses and trees), and the only difference between them is the number of each modality in each category, as shown in Figure1. Such categories with high intraclass similarities are difficult to classify. Another scenario is also possible; for example, certain Beach images seem more like Desert than Beach, and vice versa, as shown in Figure2. Such categories with high interclass diversity are difficult to classify. Because most existing deep learning-based methods are designed for scene classification and ignore the special considerations of higher intraclass similarity and interclass diversity, they are successfully used to classify single-modality-dominated scenes but achieve limited performance when classifying multimodality-dominated scenes, which have similar objects and cannot be accurately classified by only single modalities. For multimodality-dominated scenes, problems are more difficult to solve because many categories have hierarchical ontologies:

To address these problems, we propose a novel multigranularity fused convolutional neural network (MGFN) to capture the latent ontological features of remote sensing images automatically. To address the first problem, dividing the images into different levels should be a viable solution to decrease large visual-semantic discrepancies because images can be disassembled into the different hierarchical levels. To address the second problem, focusing on the superordinate-level or basic-level intraclass diversity and weighting them more heavily when fusing multiple granularities should be effective.

The primary contributions of this study include the following four aspects:

(1) We design a multigranularity module that can progressively crop input images to learn multigrained features that can describe images to different degrees.

(2) We design a maxout-based module to learn the corresponding Gaussian covariance matrices of different granularities, extract second-order features that can express the latent ontological essence of the input image, and select the most distinguished features.

(3) We provide an adaptive fusion module to fuse all obtained features using normalization to combine features of different degrees.

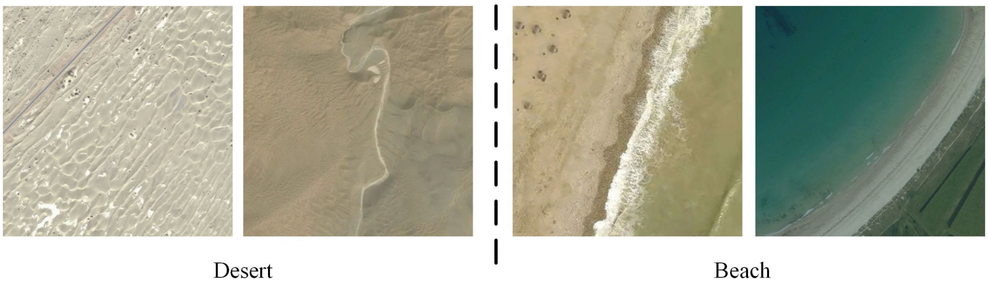

FIGURE 2. Images selected from different categories in the AID dataset. Some images may be more similar with images different category.

(4) We use several experiments on different remote sensing scene datasets to evaluate and verify the performance of the proposed network.

The remainder of this paper is organized as follows. The related work is described in Section II. We provide a novel Multi-Granularity Fused convolutional neural Network in Section III. Experiments are described and the results are discussed in Section IV, and conclusions are reviewed in Section V.

II. RELATED WORK

Propelled by the high-level feature learning capabilities of CNNs, remote sensing scene classification driven by deep neural networks has drawn remarkable attention and achieved significant breakthroughs. In this section, we review recent achievements with regard to deep learning-based remote sensing scene classification methods without referring to handcrafted feature-based and mid-level feature learning-based methods.

Because AlexNet, the first deep CNN designed by Krizhevskey *et al.* [26] in 2012, achieved the best results in a large-scale visual recognition challenge [27],many advanced CNNs have emerged in remote sensing image scene classification. There are three primary CNN-based strategies: full training, using CNNs as feature extractors and fine-tuning. We briefly introduce and analyse these three strategies in this section.

A. FULL TRAINING

Full training fully classifies remote sensing scene images with CNN models. Because these models for remote sensing scene classification have the same function and type as general CNNs, the reader is referred to the general papers [5], [28] and [29] for more information. Specifically, for remote sensing, Wang reduced low- and middle-level features via principal component analysis to obtain hierarchical global features and then aggregated these rich hierarchical features to manage images of different sizes [30]. Cheng combined CNNs with metric learning to obtain more discriminative features for remote sensing scene classification [31]. Yao proposed a weakly supervised learning method to semantically annotate high-resolution satellite images [31].

B. USING CNNs AS FEATURE EXTRACTORS

Proposed by Penatti [14], using CNNs as feature extractors is a strategy that removes the last classification layer of a pretrained CNN model and regards the remaining layers as a feature extractor. Features extracted by the remaining layers can be transferred to remote sensing scene classification because the features extracted from earlier layers are generic. Based on Penatti's work, Castelluccio preferred to replace the last layer of the pretrained network with a fully connected layer [28]. Xie transferred a CNN model that was trained on daytime remote sensing images to night-time images [32]. Hu preferred to extract CNN features from different layers by encoding multiple scale features that had been extracted from different layers into global image features [29].

Using CNNs as feature extractors can create samples for training because they require no other operations or adjustments. This strategy also performs well due to the generalization power of the features learned from the source dataset. As mentioned in [33]–[35],features extracted from earlier layers achieve better generalization than those learned in higher layers. Therefore, this strategy performs better when the target dataset is similar to the source dataset.

C. FINE-TUNING

Proposed by Jia [5], the fine-tuning strategy fine-tunes the parameters of higher layers in a pretrained CNN with a target dataset. Typically, the earlier layers of the pretrained CNN are preserved because they encode generic features, and higher layers are fine-tuned to exhibit specific features of the target dataset. Therefore, the earlier layers continue to contain low features at the pixel level, while the later layers progressively learn more specific midlevel characteristics and high-level characteristics. Based on Jia's work, Liu proposed a novel fine-tuning mode with triplet networks, in which triplet networks were pretrained on the source images and fine-tuned on the target images [36]. Additionally, Cheng verified the power of fine-tuning transfer learning on a new remote sensing scene dataset [21].

Because fine-tuning adjusts the parameters of the higher layers in a pretrained CNN model by retraining the model on the target dataset, this strategy is more complex than full training and using CNN as a feature extractor. Fine-tuned parameters can also describe features more precisely.

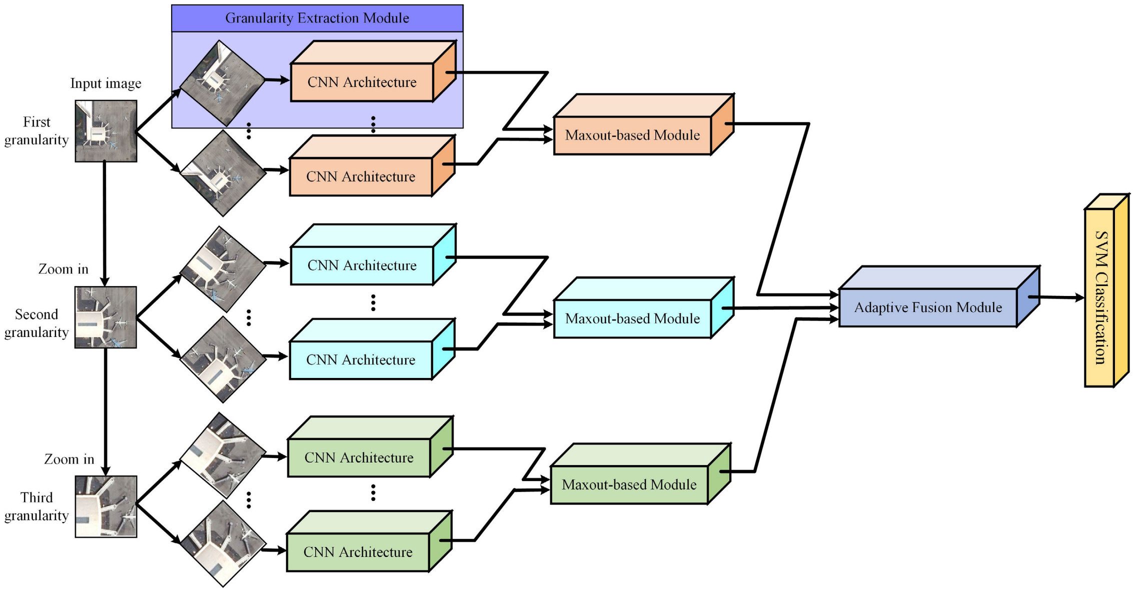

FIGURE 3. Overview of proposed model. The main strategy involves extracting granularities under different transformation, learning corresponding Gaussian covariance matrices and selecting the most distinguished ones, and finally fusing these most distinguish features for further classifying.

III. PROPOSED MGFN ARCHITECTURE

The core idea of the proposed MGFN is granularly learning hierarchical features to reduce visual-semantic discrepancies and then fusing multiple granularities with emphasis on the most distinguishing diversity. Specifically, we seek a solution for the two challenges described above. An illustration of the proposed MGFN is shown in Figure3. In this section, we first design a multigranularity extraction module that can progressively crop input images to learn multigrained features with the help of a CNN architecture. Then, we design a maxout-based module to learn the corresponding Gaussian covariance matrices of different granularities, extract the second-order features that can express the latent ontological essence of the input image, and select the most distinguished features. We then adaptively fuse all obtained features via normalization using an adaptive fused module. Finally, an SVM classifier is used to classify the fused matrix of every input image. The structure of the proposed MGFN model is shown in Figure3.

A. GRANULARITY EXTRACTION MODULE

Given an input remote sensing scene image $\boldsymbol{X} \in \mathbb{R}^{H\times W\times C}$, H,W and $C$are the height, width and channel of the image, respectively. The entire input image is defined to be the first granularity. With the central point of the previous granularity $\tilde{\boldsymbol{g}}_i$ as the core, the next granularity $\tilde{\boldsymbol{g}}_{i+1}$is obtained by cropping the middle half elements. All granularities are amplified to the same size and are obtained via scale transformation.

To avoid feature variance, we perform a rotation transformation for each granularity and formulate the rotation transformation as:

$$\boldsymbol{g}_i^t = \varphi\left(\tilde{\boldsymbol{g}}_i\right) \tag{1}$$

where $\boldsymbol{g}_i^t$ is tth transformed state of the ith granularity and $\varphi\left(\cdot\right)$ denotes the function of rotation.

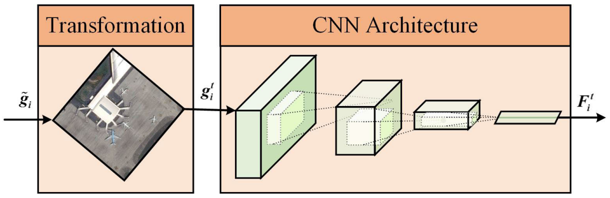

FIGURE 4. Granularity extraction module. This module is composed of two parts: rotation transformation and CNN architecture. The input of this module is the granularity cropped and zoomed from the input image and the output is CNN feature learned by CNN architecture.

Because we must learn the features of the transformed images that have the highest response to classification, we design a CNN model to extract features by:

$$\boldsymbol{F}_i^t = f_e\left(\boldsymbol{g}_i^t\right) \tag{2}$$

where, $\boldsymbol{F}_i^t$ is the feature of $\boldsymbol{g}_i^t$ extracted by function $f_e\left(\cdot\right)$. Obviously, $f_e\left(\cdot\right)$ denotes the feature extraction process with CNN architecture. The architecture of granularity extraction module is shown in Figure4.

B. MAXOUT-BASED MODULE

The maxout-based module is used to learn the corresponding Gaussian covariance matrices of different granularities to extract the optimal second-order features that can express the latent ontological essence of the input image and to select the most distinguished features. The maxout-based module can help address the problem of large intraclass variations of the multi-transformations.

The inputs of the maxout-based module are the CNN features that have been extracted by the granularity extraction module, and the outputs are the maximum optimal second-order feature $S_i^{t_0}$ and its corresponding Gaussian covariance matrix $\boldsymbol{G}_i^{t_0}$. Because CNN features of the same granularity are extracted differently from rotation transformation,

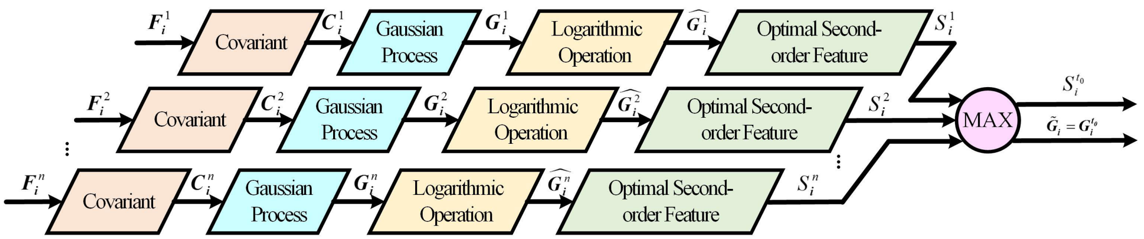

FIGURE 5. The maxout-based module. The primary purpose of this module is to extract the latent ontological features for further classification. We investigate covariant, Gaussian process, logarithmic operation and optimal second-order feature extraction for maxout operation.

these instances are invariant in different directions. Therefore, we use maxout [37] to choose the most representative features; the reader is referred to the original papers [29] for more information about maxout.

To distinguish the most representative features of the same granularities, we transform the CNN features into Gaussian covariance matrices, which can express the latent ontological features, before introducing maxout processing. Compared to the first-order feature captured by traditional deep learning networks, the second-order feature expressed by Gaussian covariance matrices preserves more spatial information about the same granularity. Because the computation of the covariance matrix is shown as the compact summarization of the second-order information, we first express second-order CNN features by the covariance matrix as:

$$\boldsymbol{C}_i^t = \boldsymbol{F}_i^t \bar{\boldsymbol{I}} \boldsymbol{F}_i^t \tag{3}$$

where, $\boldsymbol{C}_i^t$ denotes the covariance matrix of $\boldsymbol{F}_i^t$. $\boldsymbol{F}_i^t$ is the feature extracted by CNN architecture of a granularity as mentioned above and it can be written as $\boldsymbol{F}_i^t = \left[\boldsymbol{f}_1, \boldsymbol{f}_2, \ldots, \boldsymbol{f}_N\right]$. $\bar{\boldsymbol{I}}$ is defined as

$$\bar{\boldsymbol{I}} = \frac{1}{N}\left(\boldsymbol{I} - \frac{1}{N}\boldsymbol{1}\boldsymbol{1}^T\right) \tag{4}$$

where, $\boldsymbol{1}$ is the identity column vector of N dimension.

Obviously, since obtained by post-processing the granularity $\boldsymbol{g}_i^t$ on the base of CNN architecture, covariance matrix $\boldsymbol{C}_i^t$ encodes the second-order statistics of input images, preserving more spatial information then the first-order features captured by traditional deep learning networks.

Especially, as suggested by [38], the covariance matrix can be expressed by a single Gaussian model as

$$\boldsymbol{G}_i^t = \begin{bmatrix} \boldsymbol{C}_i^t + \boldsymbol{\mu}\boldsymbol{\mu}^T & \boldsymbol{\mu} \\ \boldsymbol{\mu}^T & 1 \end{bmatrix} \tag{5}$$

where $\boldsymbol{\mu} = \sum_{n=1}^{N} \boldsymbol{f}_n$. The elements of the obtained matrix $\boldsymbol{G}_i^t$ reside on the Riemannian manifold of the SPD matrix. Therefore, we apply the logarithmic operation to flatten its spatial structure and use all of the distance measurements in Euclidean space. We express this process as:

$$\widehat{\boldsymbol{G}_i^t} = \boldsymbol{G}_i^t + trace\left(\boldsymbol{G}_i^t\right)\boldsymbol{I}_{\boldsymbol{G}} \tag{6}$$

where $\boldsymbol{I}_{\boldsymbol{G}}$ denotes the identity matrix with the same dimensions of $\boldsymbol{G}_i^t$.

$\widehat{\boldsymbol{G}_i^t}$denotes feature of the ith granularity of the tth transformation of input image. Therefore we prefer to learn the optimal second-order feature of $\widehat{\boldsymbol{G}_i^t}$, for the further describing of the input image. We formulate it as

$$S_i^t = f_S\left(\widehat{\boldsymbol{G}_i^t}\right) \tag{7}$$

where the scalar S_i^t denotes optimal second-order feature of $\widehat{\boldsymbol{G}_i^t}$ and $f_S(\cdot)$ is the function of learning process. Since S_i^t is the optimal second-order which can describe the deep features of input images, the value of S_i^t indicates the description degree of different transformations with the same granularity. Therefore, we adopt the maximum operator to select the best feature as

$$S_i^{t_0} = \max_{t \in T} S_i^t \tag{8}$$

where t_0 is the maximum value of all the optimal second-order features of different transformations. We then employ the covariance Gaussian matrix of the t_0th transformation to represent the features of the ith granularity as

$$\tilde{\boldsymbol{G}}_i = \boldsymbol{G}_i^{t_0} \tag{9}$$

Obviously, $\tilde{\boldsymbol{G}}_i$ can express the ith granularity of the input image to the largest extent.

C. ADAPTIVE FUSION METHOD

The maxout-based architecture can learn and select the most represented feature of every granularity. Because several granularities exist for the same input image as described above, we now design an adaptive fusion method to fuse the features that have been extracted from all the granularities on the same input image. The inputs of adaptive fusion architecture are the optimal second-order features S_i and their corresponding Gaussian covariance matrices $\tilde{\boldsymbol{G}}_i$. Since the optimal second-order feature S_i can mostly describe the ith granularity of one input image and the value of S_i means the description degree, we prefer to assign the fusion weights according to S_i. All the optimal second-order features of the same image can be written together as a column vector $\boldsymbol{S} = [S_1, S_2, \cdots, S_n]^T$, with a size of $n \times 1$. We first normalize the optimal second-order feature vector $\boldsymbol{S}$ to a range of [0 1] as

$$\vec{\boldsymbol{S}} = \frac{\boldsymbol{S}}{\sum \boldsymbol{S}} \tag{10}$$

A granularity with a larger optimal second-order feature will obtain a bigger value in vector $\vec{\boldsymbol{S}}$, which can describe the degree of importance, and vice versa. Therefore, we fuse different granularities of the same input image according to $\vec{S}$ as

$$G^+ = \sum \vec{S}_i \tilde{G}_i \tag{11}$$

where $\vec{S}_i$ is the ith element of $\vec{S}$ as responding to the covariance Gaussian matrix of the ith granularity.

By exploiting the adaptive fusion method, features of different granularities of the same input image can be fused and describe the input image in a most distinguishable way.

D. CLASSIFICATION LAYER

Since SVM is used to find the relationship and resolve the differences between the output space, we classify the obtained fused matrix $G^+$by transmitting it to an SVM layer. For the fused matrix of the pth image G_p^+, we formulate the process as

$$h_{W,b}\left(G_P^+\right) = P\left(y = j | \, G_P^+; W, b\right) = \frac{1}{1 + \exp^{W_j (G_P^+)^T + b_j}} \tag{12}$$

where j is the current class being evaluated, and W and b represent the weights and bias respectively. Because many papers have described SVM classification in detail, the reader is referred to the paper [28] for more information.

IV. EXPERIMENTS AND ANALYSIS

In this section, the performance of MGFN is assessed using three well-known remote sensing scene classification datasets. We consider the following three components when evaluating MGFN's performance: experimental datasets, implementation parameters and experimental results and analysis.

A. EXPERIMENTAL DATASETS

We evaluated the proposed MGFN with three different remote sensing scene datasets: the UC Merced (UCM) dataset [39], the Aerial Images Dataset (AID) [25] and the NWPU-RESISC45 dataset(NR45) [21].

Extracted from large optical images of the US Geological Survey, the UCM dataset was released in 2010 and is composed of 2100 aerial scene images [39]. The spatial resolution of the UCM images is 0.3 m with 256 × 256 pixels. All images were manually labelled to 21 classes, 100 for each class. Collected from Google Earth imagery, AID was released in 2016 and consists of 10,000 aerial images [25]. All images were classified into 30 classes, and the number of each class varies greatly with scene type, from 220 to 420. The special resolution of AID changes from approximately 0.5 m to 8 m. Released in 2017, NR45 is consisted of 31500 images, covering 45 scene classes with 700images in each class. The spatial resolution of NR45 changes from 0.2m to 30m.

We select these three datasets on the consideration of the overall characteristics. There are 100, 220∼420 and 700 images in the three datasets respectively, with different spatial resolutions. In other words, these three datasets can represent general remote sensing scene datasets and validate the applicability of MGFN to some extent.

TABLE 1. The statistics of three remote sensing scene datasets.

Datasets	No. images	No. class	No. class images	Spatial resolution
UCM	2,100	21	100	256
AID	10,000	30	220∼420	600
RN45	31,500	45	700	256

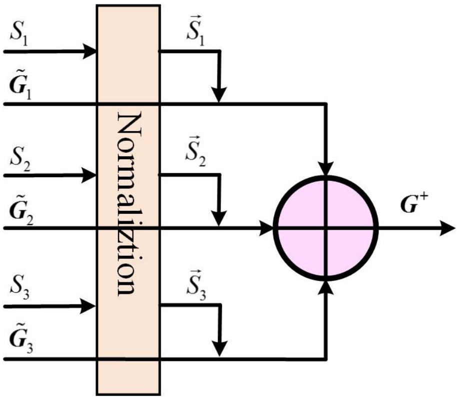

FIGURE 6. The architecture of adaptive fusion method. The main strategy involves normalizing all the optimal second-order features and weighting the corresponding Gaussian covariance matrices with these normalization values.

B. IMPLEMENTATION PARAMETERS

Considering the size of the raw input images, we designed three granularities of each instance. All the granularities of different datasets are fixed to 224 × 224 pixels in the data processing stage, for the further processing in the CNN architecture. Due to insufficient images, data augmentation techniques are adopted to avoid overfitting during training. We employed VGG-D [40] as our core processor in the granularity extraction module. The VGG-D network is pretrained on the large scale imagenet dataset as described in Section II, to achieve faster training speeds. We retrain and fine-tune the parameters of VGG-D during training process.

The learning rate of the classification layer is set to be 0.1 initially, and the entire network fine-tuned with a learning rate of 0.001. The learning rate is annealed by 0.15 every 20 epochs initially and decayed after every 5 epochs during the fine-tuning stage. The rotation transformation is set to 12 for each granularity and the batch size is set to 12 for 3 granularities. The weight decay is set to be 0.0005 and the momentum optimizer is 0.9. These parameter values were selected based on [41]–[43] and [11].

C. EXPERIMENTAL RESULTS AND ANALYSIS

We firstly conduct several experiments to analyze the influence of granularity extraction module. For every granularity, there are several corresponding transformations. So we construct different models with different granularities and transformations. As shown in Figure7, the best results gained by combining three different granularities, proved the influence of granularities. Besides, Figure7 also shows the how the numbers of transformations influence the model

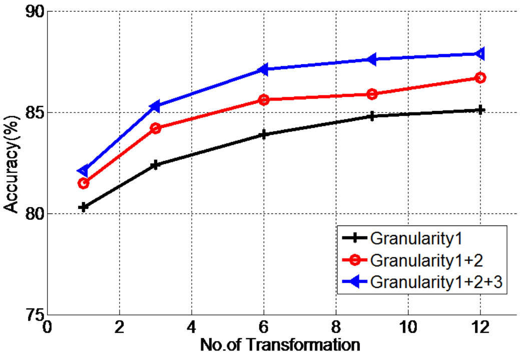

FIGURE 7. Classification accuracy gained by different granularities with different transformations.

performance. The classification accuracies increase proportionally with the number of transformations.

We then conduct several experiments to analyze the effects of maxout-based module and adaptive fusion method. We randomly select an image and compare the visualization of it after the process of these two parts as shown in Figure8. We can see that the maxout-based module can learn the most representative features of every granularity, as shown in the second line of Figure8. As a contrast, the third line in the figure shows the fused state, which is obtained by the adaptive fusion method of the three granularities. It's clearly that the fused state learns more details, compare with the state behind maxout-based module process.

We compare proposed MGFN with handcrafted feature-based methods, unsupervised feature-based methods and deep learning-based methods. Of varied deep learning-based methods, the pre-trained general-purpose networks perform better in remote sensing scene classification than special designed networks because general-purpose networks are well-designed and well-trained. So we choose the best modified deep learning models such as AlexNet, GoogLeNet for comparison.

The classification accuracy (CA) and corresponding standard deviation (SD) of the proposed MGFN are shown in Table2, and compared to several benchmark methods on the most challenging dataset NR45. Among the listed handcrafted feature-based methods, the colour histogram method achieves the best performance, even though the accuracy remains below 30%. Due to their subjectivity, handcrafted feature-based methods achieve the worst performance in classification. Unsupervised feature learning-based methods perform better than all the listed handcrafted feature-based methods, as shown in the table. However, because these methods are inadequate, and it is difficult to completely classify images with a single index, the accuracy of unsupervised feature learning-based methods is still below 50%. Conversely, deep learning-based methods overshadow both handcrafted feature-based and unsupervised feature-based methods, demonstrating their superior performance. The transferred AlexNet, which was pretrained on the large-scale ImageNet dataset, achieves an accuracy of 76.69% with a 10% training split, and 79.85% with a 20% training split. A deeper transferred CNN, the GoogLeNet model, achieves similar classification results to AlexNet because a deeper CNN can learn deeper features of the source dataset, which is not favourable to the target remote sensing scene dataset.

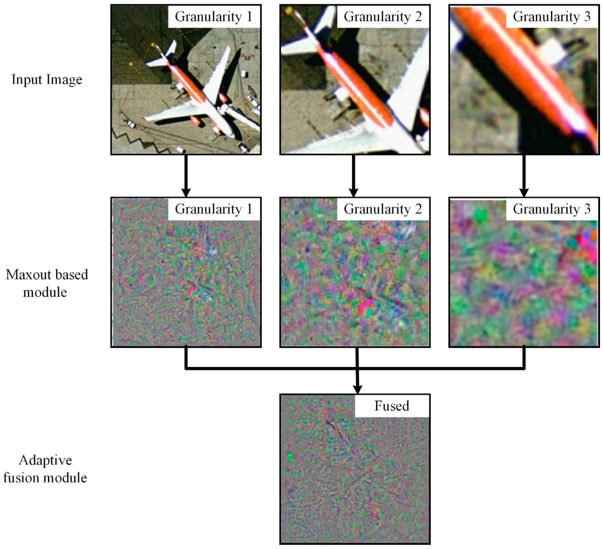

FIGURE 8. The visualization of the outputs of maxout-based module and adaptive fusion method procedure. The method introduced in [35] is used for visualization.

The proposed MGFN achieves an accuracy of 90.92% with a 10% training split, and 93.41% with a 20% training ratio, thus achieving better performance than other deep learning-based methods. This result is possibly due to the following reasons: (1) the granularity extraction module provides a chance to extract features of the same input image hierarchically, reducing visual-semantic discrepancy; (2) the maxout-based module extracts the optimal second-order features and selects the most distinguishable features, describing the latent ontological essence of input images; (3) the adaptive fusion method fuses multiple granularities with emphasis on the most distinguishing diversity; and (4) the number of input images is 3 times the number of transformed images, and they play a similar role to that of a data argument.

To verify the effectiveness of the proposed MGFN adequately, we compare it with existing deep learning methods. The CA and SD of different deep learning strategies for remote sensing scene classification are shown in3.

As shown in Table3, all the deep learning models performs better on UCM dataset than on other two datasets AID and RN45. That is because the UCM dataset has a fixed spatial resolution 0.3m. Images with 0.3m spatial resolution can describe a lot of geomorphic details, leading to the high classification accuracy for nearly all the deep learning models. Nevertheless, the special resolution of AID changes from approximately 0.5 m to 8 m and the spatial resolution of NR45 changes from 0.2 m to 30 m. Different spatial resolutions bring great difficulties for image classification. High intraclass diversity and low interclass similarity of AID and NR45 increase the difficulty of classification.

TABLE 2. Overall Classification Accuracy(CA) of different models on the most challenging dataset NR45. "SD" denotes the standard deviation of the classification accuracy.

	Models	TR=10%		TR=20%	
		CA (%)	SD	CA (%)	SD
Handcrafted Feature	GIST [21]	15.90	0.23	17.88	0.22
	LBP [21]	19.20	0.41	21.74	0.18
	Colour histograms [21]	24.84	0.22	27.52	0.14
Unsupervised Feature	BoVM+SPM [21]	27.83	0.61	32.96	0.47
	LLC [21]	38.81	0.23	40.03	0.34
	BoVM [21]	41.72	0.21	44.97	0.28
Deep learning	AlexNet [31]	76.69	0.21	79.85	0.13
	GoogLeNet [31]	76.47	0.18	79.79	0.15
	VGG-D [31]	76.19	0.38	78.48	0.26
	Proposed MGFN	99.00	0.10	93.45	0.11

TABLE 3. Overall Classification Accuracy of different models on different datasets.

Models	UCM		AID		RN45	
	CA (%)	SD	CA (%)	SD	CA (%)	SD
AlexNet+SVM [31]	94.42	0.10	84.23	0.10	81.22	0.19
GoogLeNet+SVM [31]	96.82	0.20	87.51	0.11	82.57	0.12
VGG-D+SVM [31]	97.14	0.10	89.33	0.23	87.15	0.45
MSCP with AlexNet [44]	97.29	0.63	88.99	0.38	81.70	0.23
MSCP+MARwith AlexNet [44]	97.32	0.52	90.65	0.19	88.31	0.23
MSCP with VGG-D [44]	98.36	0.58	91.52	0.21	85.33	0.17
MSCP+MAR with VGG-D [44]	98.40	0.34	92.21	0.17	88.07	0.18
DCNN with AlexNet [31]	96.67	0.10	85.62	0.10	85.56	0.20
DCNN with GoogLeNet [31]	97.07	0.12	88.79	0.10	86.89	0.10
DCNN with VGG-D [31]	98.93	0.10	90.82	0.16	89.22	0.50
RTN with VGG-D [4]	98.96	–	92.44	–	89.90	–
Proposed MGFN	99.00	0.10	93.45	0.11	90.92	0.15

The accuracy of deep learning models on these two datasets is 84% 92% and 81% 89%, much lower than that of UCM. Since the accuracy of UCM is high, the improvement of proposed MGFN is relatively small compared with AID and NR45. And as for AID and NR45, the proposed MGFN gains 1.01% and 1.02% respectively, confirming the validity of proposed MGFN.Compared to the performance of AlexNet, GoogLeNet and VGG-D in Table2, the deep learning models perform better on the same NR45 dataset. These differences primarily arise in the classification layer: the models in Table2 are combined with the softmax classifier, and those in Table3 are combined with the SVM classifier. Models with the SVM classifier perform better than those with softmax. The proposed MGFN achieves the best CA on all datasets; thus, these experimental results demonstrate the effectiveness of the proposed MGFN adequately.

V. CONCLUSION

In this paper, we proposed a novel MGFN framework to address the visual-semantic discrepancy and variance challenges in remote sensing scene classification. We designed the granularity extraction module to learn hierarchical features to reduce visual-semantic discrepancies and built the maxout-based module to extract the optimal second-order features and select the most distinguishable features to describe the latent ontological essence of input images. Then, we proposed the adaptive fusion method to fuse multiple granularities with emphasis on the most distinguishing diversity. Experiments demonstrated that we seek a solution for the two challenges mentioned above. In the future, we plan to investigate compressing this model to accelerate the training process while maintaining classification accuracy.

ACKNOWLEDGMENT

The authors would like to thank all of the team members of the D605 Laboratory.

REFERENCES

[1] 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.

[2] Q. Tan, J. Ling, J. Hu, X. Qin, and J. Hu, "Vehicle detection in high resolution satellite remote sensing images based on deep learning," *IEEE Access*, vol. 8, pp. 153394–153402, 2020.

[3] L. Ma, Y. Liu, X. Zhang, Y. Ye, G. Yin, and B. A. Johnson, "Deep learning in remote sensing applications: A meta-analysis and review," *ISPRS J. Photogramm. Remote Sens.*, vol. 152, pp. 166–177, Jun. 2019.

[4] P. Li, D. Zhang, P. Chen, X. Liu, and A. Wulamu, "Multi-adversarial partial transfer learning with object-level attention mechanism for unsupervised remote sensing scene classification," *IEEE Access*, vol. 8, pp. 56650–56665, 2020.

[5] S. Jia, H. Liu, and F. Sun, "Aerial scene classification with convolutional neural networks," in *Proc. Int. Symp. Neural Netw.*, 2015, pp. 258–265.

[6] J. Wang, Q. Qin, Z. Li, X. Ye, J. Wang, X. Yang, and X. Qin, "Deep hierarchical representation and segmentation of high resolution remote sensing images," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, Jul. 2015, pp. 4320–4323.

[7] W. Yang, X. Yin, and G.-S. Xia, "Learning high-level features for satellite image classification with limited labeled samples," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 8, pp. 4472–4482, Aug. 2015.

[8] M. Vakalopoulou, K. Karantzalos, N. Komodakis, and N. Paragios, "Building detection in very high resolution multispectral data with deep learning features," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, Jul. 2015, pp. 1873–1876.

[9] L. Khelifi and M. Mignotte, "Deep learning for change detection in remote sensing images: Comprehensive review and meta-analysis," *IEEE Access*, vol. 8, pp. 126385–126400, 2020.

[10] G. Sumbul and B. Demir, "A deep multi-attention driven approach for multi-label remote sensing image classification," *IEEE Access*, vol. 8, pp. 95934–95946, 2020.

[11] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, p. 436, 2015.

[12] H. Zhao, F. Liu, H. Zhang, and Z. Liang, "Research on a learning rate with energy index in deep learning," *Neural Netw.*, vol. 110, pp. 225–231, Feb. 2019.

[13] F. Hu, G.-S. Xia, J. Hu, Y. Zhong, and K. Xu, "Fast binary coding for the scene classification of high-resolution remote sensing imagery," *Remote Sens.*, vol. 8, no. 7, p. 555, Jun. 2016.

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

[15] K. Qi, W. Liu, C. Yang, Q. Guan, and H. Wu, "Multi-task joint sparse and low-rank representation for the scene classification of high-resolution remote sensing image," *Remote Sens.*, vol. 9, no. 1, p. 10, Dec. 2016.

[16] Y. Liu, Y. Zhong, F. Fei, and L. Zhang, "Scene semantic classification based on random-scale stretched convolutional neural network for high-spatial resolution remote sensing imagery," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, Jul. 2016, pp. 763–766.

[17] H. Zhao, F. Liu, H. Zhang, and Z. Liang, "Convolutional neural network based heterogeneous transfer learning for remote-sensing scene classification," *Int. J. Remote Sens.*, vol. 40, no. 22, pp. 8506–8527, Nov. 2019.

[18] K. Makantasis, K. Karantzalos, A. Doulamis, and N. Doulamis, "Deep supervised learning for hyperspectral data classification through convolutional neural networks," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, Jul. 2015, pp. 4959–4962.

[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. X. Zhu, D. Tuia, L. Mou, G.-S. Xia, L. Zhang, F. Xu, and F. Fraundorfer, "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.

[21] 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.

[22] B. Yuan, S. Li, and N. Li, "Multiscale deep features learning for land-use scene recognition," *J. Appl. Remote Sens.*, vol. 12, no. 1, p. 1, Feb. 2018.

[23] N. Xin-Lin, "An introduction to cognitive linguistics," *J. Chengdu College Educ.*, vol. 17, no. 8, pp. 1245–1253, 2006.

[24] F. G. D. Matos, "Cognitive linguistics: An introduction," *Delta Documentao De Estudos Em Lingüística Teórica E Aplicada*, vol. 23, no. 2, pp. 397–398, 2006.

[25] G. S. Xia, J. Hu, F. Hu, B. Shi, X. Bai, Y. Zhong, L. Zhang, and X. Lu, "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.

[26] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2012, pp. 1097–1105.

[27] J. Deng, W. Dong, R. Socher, L. J. Li, K. Li, and F. F. Li, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2009, pp. 248–255.

[28] M. Castelluccio, G. Poggi, C. Sansone, and L. Verdoliva, "Land use classification in remote sensing images by convolutional neural networks," 2015, *arXiv:1508.00092*. [Online]. Available: http://arxiv.org/abs/1508.00092

[29] F. Hu, G.-S. Xia, Z. Wang, X. Huang, L. Zhang, and H. Sun, "Unsupervised feature learning via spectral clustering of multidimensional patches for remotely sensed scene classification," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 8, no. 5, pp. 2015–2030, May 2015.

[30] G. Wang, B. Fan, S. Xiang, and C. Pan, "Aggregating rich hierarchical features for scene classification in remote sensing imagery," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 10, no. 9, pp. 4104–4115, Sep. 2017.

[31] 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.

[32] M. Xie, N. Jean, M. Burke, D. Lobell, and S. Ermon, "Transfer learning from deep features for remote sensing and poverty mapping," in *Proc. 30th AAAI Conf. Artif. Intell.*, 2016, pp. 3929–3935.

[33] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional neural networks," in *Proc. Eur. Conf. Comput. Vis.*, Sep. 2014, pp. 947–958.

[34] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, "Caffe: Convolutional architecture for fast feature embedding," in *Proc. 22nd ACM Int. Conf. Multimedia*, 2014, pp. 675–678.

[35] A. Mahendran and A. Vedaldi, "Understanding deep image representations by inverting them," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 5188–5196.

[36] Y. Liu and C. Huang, "Scene classification via triplet networks," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 11, no. 1, pp. 220–237, Jan. 2018.

[37] I. J. Goodfellow, D. Warde-Farley, M. Mirza, A. Courville, and Y. Bengio, "Maxout networks," in *Proc. Int. Conf. Mach. Learn.*, 2013, pp. 1319–1327.

[38] B. Du, W. Xiong, J. Wu, L. Zhang, L. Zhang, and D. Tao, "Stacked convolutional denoising auto-encoders for feature representation," *IEEE Trans. Cybern.*, vol. 47, no. 4, pp. 1017–1027, Apr. 2017.

[39] Y. Yang and S. Newsam, "Bag-of-visual-words and spatial extensions for land-use classification," in *Proc. 18th SIGSPATIAL Int. Conf. Adv. Geographic Inf. Syst. (GIS)*, 2010, pp. 270–279.

[40] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556v6*. [Online]. Available: http://arxiv.org/abs/1409.1556v6

[41] W. Zhou, Z. Shao, C. Diao, and Q. Cheng, "High-resolution remote-sensing imagery retrieval using sparse features by auto-encoder," *Remote Sens. Lett.*, vol. 6, no. 10, pp. 775–783, Oct. 2015.

[42] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2005, pp. 886–893.

[43] J. Sivic, "Video Google: A text retrieval approach to object matching in videos," in *Proc. 9th IEEE Int. Conf. Comput. Vis.*, Oct. 2003, p. 1470.

[44] 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.

ZHIGUO ZENG was born in Hunan, China, in 1977. He received the M.S. degree in information and communication engineering from Central South University, in 2011. He is currently pursuing the Ph.D. degree with Air and Missile Defense College, Air Force Engineering University. His main research interests include deep learning in convolutional neural networks and computer vision.

XIHONG CHEN was born in 1961. He is currently a Professor and a Ph.D. Supervisor with Air and Missile Defense College, Air Force Engineering University.

ZHIHUA SONG was born in Hebei, China, in 1982. He received the B.S. degree in missile engineering, and the M.S. and Ph.D. degrees in military operations research from Air Force Engineering University, in 2004 and 2010, respectively.