Multilevel Capsule Weighted Aggregation Network Based on a Decoupled Dynamic Filter for Remote Sensing Scene Classification

CHUNYUAN WANG, YANG WU, YIHAN WANG, YIPING CHEN, AND YAN GAO

1School of Electronic and Electrical Engineering, Shanghai University of Engineering Science, Shanghai 200000, China
2Shanghai Institute of Satellite Engineering, Shanghai 200000, China
Corresponding author: Yihan Wang (ehanmails@163.com)

This work was supported by the National Natural Science Foundation of China under Grant 61801286.

ABSTRACT The improvement of remote sensing scene classification (RSSC) by effectively extracting discriminant representations for complex and diverse scenes remains a challenging task. The capsule network (CapsNet) can encode the spatial relationship of features in an image, which exhibits encouraging performance. Nevertheless, the original CapsNet is unsuitable for RSSC with complex image background. In addition, conventional neural network methods only use the last features from their last convolutional layer and discount the intermediate features with complementary information. To search the additional information in intermediate convolutional layers and increase the performance of feature aggregation, this paper proposes a multilevel capsule weighted aggregation network (MCWANet) based on a decoupled dynamic filter (DDF), in which a new multilevel capsule encoding module and a new capsule sorting pooling (CSPool) method are implemented by combining the advantageous attributes of a residual DDF block, weighted capsule aggregation, and the new CSPool method. Extensive experiments on two challenging datasets, AID and NWPU-RESISC45, demonstrate that multilevel and multiscale features can be extracted and fused to extract semantically strong feature representation and that the proposed MCWANet performs competitively in RSSC.

INDEX TERMS Scene classification, decoupled dynamic filter, weighted aggregation, capsule network.

I. INTRODUCTION A remote sensing image (RSI) contains the detailed spatial structure of Earth’s surface, which is captured by satellite imaging sensors. Remote sensing scene classification (RSSC) has been a popular research topic in the field of high-resolution RSI in recent decades; its purpose is to assign semantic labels to the images [1] automatically. Due to the rapid development of aeronautics and astronautics technology, many RSIs provide us with more applications for RSSCs, such as land planning, urban change, traffic control, disaster monitoring, military needs, agriculture, and forestry [2], [3]. The key to RSSC is to extract robust and distinctive feature representations. However, the special bird’s eye view image acquisition mode makes traditional RSSC methods ineffective. In addition, an RSI might include multiple land cover units of different sizes, or different RSIs might include the same land cover unit, which causes great difficulties in RSSC. Thus, realizing automatic and high-precision RSSC is still a challenge [4].

Robust feature representation is the focal point in RSSC. In accounting for its applications, RSSC techniques can be divided into two categories: human-defined feature extraction and depth feature extraction. Owing to the excellent capability of deep learning, the depth feature approach exhibits better results from RSIs than human-defined features [5], [6]. Therefore, using deep learning technology has been widely considered and proven to be the central topic in RSSC tasks. Because of their advantages in data processing power and hierarchical learning abilities, CNNs (convolutional neural networks) have been extensively applied in information extraction and RSSC and have achieved many state-of-the-art results [7], [8]. Although CNNs perform well in many tasks, they also have many limitations. CNNs cannot explicitly learn...
the relationship between feature locations, and it is difficult to identify similar objects with different location relationships. Moreover, CNN uses a pooling operation to reduce the number of parameters and maintain translation invariance, but conventional pooling methods cause the loss of essential details and location information.

Recently, Hinton et al. [9] advanced the concept of “capsule”, an interpretable deep learning model and is considered a powerful alternative to CNN. Different from the traditional deep learning model based on scalar neurons, the capsule network (CapsNet) uses a vector capsule representation to learn equivariant representation, which is more robust to the spatial relations between objects and the changes in their poses in the image. CapsNet has been extensively applied to classification, recognition, segmentation and prediction tasks [10]–[12]. However, the original CapsNet [13] only employs two convolution layers and uses a great number of training parameters to attempt to explain all the contents of an image, which is unsuitable for RSIs with complex backgrounds. In addition, most of these methods only focus on the feature maps of the last layer and discount the intermediate feature maps. Those feature maps obtained by different intermediate layers possess considerable supplementary information and discrimination ability.

This paper proposes a novel CapsNet for RSSC, named MCWANet, which employs a new multilevel capsule encoding module and a new capsule sorting pooling (CSPool); these elements include a residual decoupled dynamic filter (DDF) block, weighted capsule aggregation, and the new CSPool method. A DDF [14] decouples the conventional dynamic filter into spatial and channel dynamic filters with adaptive content and lighter weight, which can enhance the abundant information in different spatial locations and different channels. Here, we construct a residual structure [15] DDF to alleviate the vanishing gradient problem caused by increasing the depth of a deep neural network. The multilevel capsule encoding module is enhanced to generate multilevel capsule aggregation directly from intermediate feature layers to enrich the feature space with different receptive fields. The feature obtained by the final convolution has the largest receptive field and can locate large objects with high-level semantic information. The features obtained by the intermediate layers are more appropriate and robust in the location of minimal objects, containing essential details and low-level texture patterns. We use a weight matrix to condition the aggregation extent for extracting intermediate features. To alleviate the overfitting problem and reduce the number of feature parameters, the CSPool method is proposed to improve the output quality and the robustness of capsules. The novel MCWANet achieves competitive and promising performance in RSSC.

The contributions of this paper are summarized as follows.

1. A new end-to-end MCWANet is structured to extract multilevel and informative features to provide a semantically strong capsule representation that increases RSSC accuracy;

2. A new multilevel capsule encoding module is designed with a residual-structure DDF and a weighted aggregation strategy, which highlights both the spatialwise and channelwise salient and informative features and makes full use of the feature maps of intermediate layers, playing a positive role in improving the ability of capsule representation;

3. A new CSPool method is proposed to provide a high-quality and robust capsule representation that reduces the number of capsules to prevent feature redundancy and destruction, further improving the RSSC result.

4. Our novel model achieves excellent RSSC accuracies compared to those of some state-of-the-art algorithms on two challenging datasets. The structure of this paper is as follows. Related studies are introduced in Section II. Section III describes the novel architecture of the MCWANet in detail. In Section IV, the experimental results are discussed. Finally, the conclusion is given in Section V.

II. RELATED STUDIES

Robust representation of RSIs is the key to improving RSSC accuracy. Human-defined feature descriptors are usually used for RSSC. In [16], the BoVW method was proposed to encode the scale invariant feature transform [17] for local blocks. In [18], two kernels were used to construct spatial relationships: cooccurrence kernels and pyramid matching kernels. Zhao et al. [19] introduced the Fisher kernel [20] and used a gradient vector to encode human-defined features. Negrel et al. [21] studied several second-order methods to fuse the histogram of the directional gradient [22]. However, human-defined methods are challenging to apply in practice because of the complexity and diversity of RSIs, which encourages the development of more robust and accurate recognition schemes [23].

Hu et al. [24] extracted depth features from multiscale images based on CNN and encoded them into scene representations by using traditional feature coding methods. To achieve the high-level feature, Yu and Liu [25] constructed three neural networks with different receptive fields, which were fused by a probability fusion model. In [26], a multilayer stacking covariance gap was advanced to fuse intermediate features, where a covariance matrix was proposed to represent the complementary information of the superimposed feature map. Wang et al. [27] proposed a coding hybrid resolution representation that integrates low-level and high-level features. Zhu et al. [28] integrated multilevel semantics from global texture, visual word packages and preprocessed CNNs and then introduced extra descriptors to optimize several feature representations. Unfortunately, the descriptors usually damage the original structure of feature spatial information by reconstructing new feature tensors. In addition, DenseNet [29] found that the fusion of feature maps generates fusion redundancy.

CapsNet, which was first implemented by Sabour et al. [13], is an effective network for classification. There have been some other innovative achievements. Hinton described a version of the capsule in [9], in which a matrix capsule
that learns entity and posture was proposed to represent the different attributes of an object better. Chen et al. [30] embedded the routing process and all the other neural network parameters into the optimization process, which solves the problem of manually setting the optimal route times. In [31], a hit net was proposed, which is an improved vertex net constructed from a hit or dead middle layer. Rosario et al. [32] introduced MLCN, which is a new type of poverty reduction center network composed of several parallel lanes. It makes parallel computation, high-speed training and minimal training parameters possible. Cheng et al. [33] proposed a complex valued vertex network that uses a multiscale complex convolution layer to extract multiscale complex features and construct complex capsules. Experiments showed better performance than the original CapsNet. However, the design of their methods only adopts the spatial structure feature, which restricts the ability of CapsNet. In addition, Yin et al. [34] applied CapsNet to hyperspectral image classification. Mobiny and Van Nguyen [35] achieved lung cancer screening by using a fast CapsNet. Afshar et al. [36] successfully realized the classification of brain tumor images by using CapsNet.

III. MATH

In this paper, a novel MCWANet is proposed, as shown in Fig. 1. Compared with the original CapsNet, the MCWANet constructs three new modules: a residual DDF module, a multilevel capsule encoding module and a capsule sorting pooling module. Dynamic routing is used between the primary capsule layer (Pri-CapsL) and the digital capsule layer (Dig-CapsL).

A. DECOUPLED DYNAMIC FILTER

It is proven that the excellent properties of a DDF are as follows [14]. A DDF provides a spatially variable filter to guarantee content adaptation; it has the same amount of computation as deep convolution, which makes its reasoning speed faster than that of standard convolution and dynamic filters; and it can significantly reduce the memory consumption of dynamic filters, which enables the direct replacement of all standard convolutions with a DDF.

Setting \( F \in \mathbb{R}^{c \times n} \), \( n = h \times w \) as a given input feature, the standard convolution operation of a feature vector \( F_{(:,i)} \) at the \( i \)-th pixel can be described as follows:

\[
F'_{(:,i)} = \sum_j W[p_j - p_i] F_{(:,j)} + b
\]

where \( b \) is the bias vector. \( W \) represents a \( k \times k \) convolution. Here, the filter \( W \) is the same for all pixels; that is, the filter is independent of the input content.

Different from standard convolution, the dynamic filter uses an additional self-network to generate a filter for each pixel; that is, the spatial invariant filter in the above formula becomes a spatially variable filter. Dynamic filters can enhance content-adaptive learning, but they also entail considerable computation and occupation.

To meet the challenge that a single filter should be both “content adaptive” and “lighter than standard convolution”, a DDF achieves both goals. The key is constructing DDFs as both spatial and channel dynamic filters. A DDF is described as follows:

\[
F'_{(r,i)} = \sum_j D_j^{sp} [p_j - p_i] D_r^{ch} [p_i - p_j] F_{(r,j)}
\]

where \( F_{(r,i)} \in \mathbb{R}^c \) and \( F_{(r,j)} \in \mathbb{R}^c \) represent the output feature vector and the input feature vector of the \( i \)-th pixel in the \( r \)-th channel, respectively. \( c' \) and \( c \) represent the output channels and the input channels, respectively, and \( D_j^{sp} [p_j - p_i] \) and \( D_r^{ch} [p_i - p_j] \) represent the spatial dynamic filter and the channel dynamic filter, respectively. The DDF operation and its procedure are shown in Fig.2. and the filter prediction branch is expected to be as light as possible. The channel and the spatial dynamic filters can be predicted from the input, and the output characteristics can be calculated according to the above formula. Compared with the conventional dynamic filter, the DDF spatial domain reduces the original pixel-wise filter values \((n \times c' \times c \times k \times k)\) to \(nk^2\) spatial filters and \(ck^2\) channel dynamic filters.
FIGURE 2. The chart of the DDF operation and procedure. Where ‘GAP’ expresses the global average pooling.

The spatial and channel filters of an attention type of classification prediction are designed according to the correlation between the dynamic filter and its attention focus. For the prediction branch of the spatial filter, only one convolution is used. For the prediction branch of the channel filter, a structure like the squeeze and extraction (SE) [37] block is adopted, which is \( \text{GAP} + \text{FC} + \text{ReLU} + \text{FC} \). Because direct use of these direct prediction filters, which may be too large or too small, would lead to instability of training, the filters are normalized (here, FN is designed concerning batch normalization (BN)):

\[
D_{si}^{sp} = \alpha^{sp} \frac{D_{si}^{sp} - \mu(D_{si}^{sp})}{\delta(D_{si}^{sp})} + \beta^{sp} \tag{3}
\]

\[
D_{ri}^{ch} = \alpha^{ch} \frac{D_{ri}^{ch} - \mu(D_{ri}^{ch})}{\delta(D_{ri}^{ch})} + \beta^{ch} \tag{4}
\]

where \( D_{si}^{sp} \) and \( D_{ri}^{ch} \) are filters before normalization; \( \mu \) and \( \delta \) represent the mean value and the standard deviation, respectively; and \( \alpha \) and \( \beta \) represent the moving standard deviation and moving mean value, respectively. FN helps limit the generated filter values to a reasonable range so that it can avoid the vanishing and exploding gradient problems in the training process.

Inspired by the residual pattern of ResNet, MCWANet employs a residual pattern to join the DDF convolution module, as shown in Fig. 1. Because the input image must undergo a multilevel convolution operation from its initial input to its final output, the receptive field is increased.

B. MULTILEVEL CAPSULES ENCODING MODULE

1) PRINCIPLE OF AGGREGATION

In the conventional CapsNet, the input of a convolutional operation is the output of the previous convolutional operation. The feature obtained from the final convolutional operation is employed as the representative feature constructed the capsules. As shown in [25] and [26], different levels of convolution features contain diverse spatial structure information from the image. The receptive field of low-level convolution is microscopic, which can obtain appearance information from images. In contrast, high-level convolution has a large receptive field that can obtain the spatial structure information between objects.

To integrate the intermediate convolutional feature maps from multiple convolutional layers, a multilevel capsule encoding module is proposed, as shown in Fig. 1. The feature aggregation can be represented as:

\[
MF = \sum_{i=1}^{n} Ag(CF_i, W_i) \tag{5}
\]

where \( Ag \) refers to the aggregation method, \( CF_i \) represents the capsule feature maps of the \( i \)-th layer, \( W_i \) expresses the corresponding weight parameters, and \( n \) is the aggregation layer number.

Multilevel feature aggregation is realized by stacking all the feature maps in a parallel connection form. Affected by the stacking aggregation, some useless information will inevitably be fused, which would lead to redundancy, destruction of features and reduced efficiency. On the one hand, there will be considerable redundant information in the stacking feature maps from overlapping adjacent convolutional layers. On the other hand, the change of some pixels can destroy the original feature structure, especially the immature abstraction of the features transferred from the previous layer to the rearwards layer. We designed a weight matrix to address these issues and enhance the aggregation effectiveness to condition the aggregation extent.

2) WEIGHT MATRIX FOR THE AGGREGATION

The weight matrix is used to control the degree of feature map participation in the aggregation. Each level of a feature map is assigned a weight \( W_i \) to control its aggregation extent. When the weight is small, the feature map controlled by the weight has little influence on the aggregation and can reduce redundant information and structural destruction. If the weight is large, the ratio of representative feature maps is large, which is beneficial to aggregation effectiveness. The principle of the prediction vector computed with different
the extent to which the capsule is activated.

\[
\hat{C}_F = \tilde{W}_1 \cdot CF_1 \\
\hat{C}_F = \tilde{W}_2 \cdot CF_2 \\
\vdots \\
\hat{C}_F = \tilde{W}_i \cdot CF_i \\
MF_{stack} = \hat{C}_F_1 \cup \hat{C}_F_2 \cup \cdots \hat{C}_F_i
\]  

(6)

where \(W_1, W_2\) and \(W_i\) are the weight matrices between \(CF_1, CF_2, CF_i\) and \(\hat{C}_F_1, \hat{C}_F_2, \hat{C}_F_i\) respectively. \(\hat{C}_F_i\) is the prediction vector of the capsules obtained by the DDF. And \(MF_{stack}\) represents the output of the multilevel capsule encoding module which is then used as the proposal primary capsules (ProPri-Caps). The variety of feature representations is encoded by using the weight matrix between the original capsules and the ProPri-Caps. In the training process, the part-whole relationship of every capsule pair is learned by conditioning the weight matrix \(W_1, W_2\) and \(W_i\).

Parallel stacking can collect the characteristics of different feature maps and preserve the original pixels. In the final aggregation, the output capsules of the DDF blocks realize multiscale feature extraction and achieve the combination of different numbers and sizes of receptive fields, where the image’s structural information and semantic information.

### C. CAPSULES POOLING

The multilevel capsule encoding module creates much redundant information, so a new capsule pooling method is proposed to lessen the parameter numbers of MCWANet, which can prevent common overfitting and maintain accuracy. The standard pooling method always operates on each element of a capsule, which can change the capsule vector direction and lead to a change in the entity attributes that the capsule expresses and cause difficulty in learning. Therefore, different from the standard pooling method, and considering that each capsule in CapsNet is a vector, capsule pooling operates on the entire capsule vector instead of a single element to preserve feature information (location, direction, etc.). We regard each capsule as a whole and subsample along the channel direction to protect the complete expression of object parts. This ensures that the direction of the capsule remains unchanged.

In the ProPri-Caps layer, the ProPri-Caps are reshaped from the convolutional results of the DDF blocks where the prediction branch of channel filters has gone through an attention structure like SE. As a result, important channels are given more weight than useless channels that represent redundant information. Therefore, a response factor \(V_{rxy}\) is defined as the energy value of a capsule vector, which represents the extent to which the capsule is activated.

\[
V_{rxy} = \sqrt{\sum_n (v_{rxy})^2}
\]

(7)

where \(v_{rxy}\) is each element value of a capsule vector at location \((x,y)\), and \(n\) is the dimension of the capsule vector.

For every capsule at the same location in all the capsule decks, we sequence the \(V_{rxy}\) and \(k\) capsules as follows

\[
k = \text{MIN}[\text{top\_index}(V_{rxy}), N/2]
\]

(8)

where \(N\) is the capsule number at position \((x,y)\), \(\text{top\_index}\) implies how many capsule responses are larger than the mean capsule response at position \((x,y)\) in all capsule decks, and MIN signifies taking a small value. Therefore, \(k\) capsules are passed to the next layer at a pixel location as the primary capsules (Pri-Caps). Since the direction of the capsule vector is not changed, the new Caps-pooling algorithm is more suitable for CapsNet.

### D. DYNAMIC ROUTING

The Pri-Caps \(u\) are received after the CSpool operation. As shown in Fig.3, between Pri-CapsL and Dig-CapsL, the dynamic routing algorithm [13], which is a kind of information selection mechanism, is used to iteratively update coupling coefficients, where all the Pri-Caps of feature maps in all the levels are collected as the input. The coupling coefficients \(c_{ij}\) are obtained with a softmax function

\[
c_{ij} = \frac{\exp(b_{ij})}{\sum_n b_{ij}}
\]

(9)

where \(b_{ij}\) is initialized to 0 before the training begins.

In the dynamic routing algorithm, the coefficient \(b_{ij}\) is iteratively refined by measuring the agreement between the Pri-Caps vectors \(u_{ij}\) and \(v_j\). Dynamic routing adds a routing coefficient value to a \(j\)-digital capsule by using the inner product between two vectors:

\[
b_{ij} = b_j + v_j \cdot u_j
\]

(10)

If a good agreement is reached, Pri-Caps \(u_{ij}\) makes a good prediction for the digital capsule \(v_j\). As a result, the coefficient \(b_{ij}\) is significantly increased. By using a squash function, the output of capsule \(v_j\) can be obtained from input \(s_j\). The equation used here is as follows

\[
v_j = \frac{||s_j||^2 \cdot s_j}{1 + ||s_j||^2 \cdot ||s_j||}
\]

(11)
After normalization, the short vector is close to 0, and the long vector is almost 1. The loss function calculated with the margin loss, is given as:

\[
L_m = T_m \max\left(0, k^+ - ||v_m||^2\right) + \lambda (1 - T_m) \max\left(0, ||v_m|| - k^-\right)^2 \tag{12}
\]

When the m class really exists, then \(T_m = 1\) and \(k^+, k^-\) and \(\lambda\) are parameters that are established during training. The overall loss is the sum of the marginal losses of all outputs in the final layer.

IV. EXPERIMENTAL RESULTS AND ANALYSIS

In this section, our MCWANet is used on public datasets. First, the datasets are presented. Second, the experimental settings are introduced. Third, the experimental results are shown.

A. DATASETS

Two challenging public datasets are used to estimate the proposed MCWANet method. Sample images of the two datasets are displayed in Fig.4. Then, each data set is separately introduced.

AID dataset: This dataset is composed of 10000 samples (600 \(\times\) 600 pixels). There are 30 classes, each of which has 220 to 420 images. The resolutions of these images are between 0.5 m and 8 m. This large data set is available from the website http://www.captain-whu.com/project/AID/.

NWPU-RESISC45 dataset: This dataset contains 31500 images, including 45 classes, where every class has 700 images (256 \(\times\) 256 pixels). The dataset was obtained from Google Earth. The resolutions range from 0.2 m to 30 m. The dataset can be acquired at http://www.escience.cn/people/JunweiHan/NWPU-RESISC45.html.

B. EXPERIMENTAL DESIGN

In this paper, all experiments are carried out on a personal computer with an Intel 4.2 GHz 4core i7-7700 CPU and an NVIDIA RTX 2070 GPU with 16 GB of memory. The PyTorch framework is adopted to conduct the experiments. All images are scaled to 224 \(\times\) 224 pixels before training. The learning rate is initialized as 0.001 and plus 0.1 when the loss per epoch does not decrease. The weight decay parameter is set to 0.0005. For the convenience of feature aggregation and pooling operations, the stride is set to 1. Batch normalization (BN) is used after each convolutional operation, and the ReLU activation function is adopted.

The basic CapsNet contains three layers: two convolutional layers and one capsule layer. The convolution kernel of the first convolutional layer is 256, 9 \(\times\) 9 with a stride of 1. The second layer is a Pri-CapsL with 32 capsules of 8-dimensional vectors at each pixel position, which is obtained from the results of the preceding layer by 8, 9 \(\times\) 9 convolution kernels with a stride of 2. The final layer is an FC-CapsL with a 16D capsule representing a class. Table 1 describes the parameter settings of the proposed MCWANet in detail.

| Layer       | Channel No. | Activation Function |
|-------------|-------------|---------------------|
| DDF1        | 64          | ReLU                |
| DDF2        | 128         | ReLU                |
| DDF3        | 256         | ReLU                |
| DDF4        | 512         | ReLU                |
| ProPri Caps | 120         | /                   |
| Digital Caps| 8 \(\times\) 16| Squash             |

Three evaluation criteria were used: overall accuracy (OA), standard deviation (SD) and confusion matrix (CM). OA is the number of correctly recognized images divided by the total number of test images. The CM is a kind of information table where the columns represent the ground truth, and rows represent the prediction.

To thoroughly test the effectiveness of the advanced MCWANet, we adopted 10-fold cross validation in the experiment. Furthermore, two sets of training-testing ratios are adopted. For the AID dataset, the training-testing ratios are set to 5:5 and 2:8. For the NWPU-RESISC45 dataset, the ratios are 2:8 and 1:9. Each data set is randomly divided into 10 parts without repeated sampling, and all experiments are performed 10 times. Finally, the mean results of the OA and SD were reported as the final results.
C. RESULTS AND ANALYSIS

1) EXPERIMENTAL RESULTS FROM THE AID DATASET

The AID dataset is a large-scale RSI dataset in which at least 100 images in every scene class are used for training models. All the compared algorithms are based on deep convolutional networks. Table 2 shows the comparison results of the different algorithms. As shown, our MCWANet achieves the highest OA of $95.89\% \pm 0.23\%$ and $93.62\% \pm 0.18\%$ with training ratios of 5:5 and 2:8, respectively. In particular, the CNN-CapsNet that combines VGG_16 and CapsNet achieves excellent performance on the AID dataset. For training rates of 5:5 and 2:8, the performance of our MCWANet is 1.15% and 1.99% higher than that of CNN-CapsNet, respectively. Furthermore, the recognition OA of the proposed network is 1.05% and 2.28% higher than that of MF2Net, respectively; MF2Net is one of the newest RSSC methods. This emphasizes that MCWANet can learn more robust feature representations from RSIs. The CM of MCWANet is shown in Fig. 5 for the 5:5 training ratio. The data in each row of the CM represent the prediction category of each image. Our model has a better capability of processing most categories. Some misclassified scenes are difficult to distinguish for human interpreters.

2) EXPERIMENTAL RESULTS FROM THE NWPU-RESISC45 DATASET

This dataset is a large-scale dataset and is one of the most challenging at present. The OA and SD for different methods are shown in Table 3, in which the proposed MCWANet exhibits the best performance. Compared with the second-best model, the enhancement achieved by MCWANet with a 1:9 training ratio is 0.75% (RAN), and with a 2:8 training
TABLE 2. OA (%) and SD of different methods under 5:5 and 2:8 training ratios on the AID dataset.

| Method                  | OA(TP5:5) | OA(TP2:8) |
|-------------------------|-----------|-----------|
| GoogLeNet[38]           | 86.39 ± 0.55 | 83.44 ± 0.40 |
| VGG-16[38]              | 89.64 ± 0.36 | 86.59 ± 0.29 |
| Fusion by addition[39]  | 91.87 ± 0.36 | /          |
| Fusion by concatenation[39] | 91.86 ± 0.28 | /          |
| RAN[40]                 | 93.66 ± 0.28 | 92.18 ± 0.42 |
| Multilevel Fusion[25]   | 94.17 ± 0.32 | /          |
| MF²Net[41]              | 94.84 ± 0.27 | 91.34 ± 0.35 |
| S-CNN[42]               | 95.24 ± 0.18 | 92.38 ± 0.13 |
| VGG-16-CapsNet[43]      | 94.74 ± 0.17 | 91.63 ± 0.19 |
| MCANet                  | 95.89 ± 0.23 | 93.62 ± 0.18 |

The OA and SD results obtained by inserting different modules into the original CapsNet [18] are shown in Table 4, and the training times of an epoch and the number of training parameters for the AID dataset are listed in Table 5.

3) EFFECTIVENESS OF DIFFERENT SUBPARTS

As described in Section III, our MCWANet includes three main improvements: a residual-structure DDF module, a multilevel capsule encoding module, and a CSPool strategy. To validate their effectiveness in MCWANet, we implement the following ablation experiments.

The OA and SD results obtained by inserting different modules into the original CapsNet [18] are shown in Table 4, and the training times of an epoch and the number of training parameters for the AID dataset are listed in Table 5.

“Convs-Caps” identifies the model that substitutes the 2 ConvLayers with a 9 × 9 convolution kernel for the 4 ConvLayers with 3 × 3 convolution kernels. “Res-Caps” identifies the model that substitutes the 3 × 3 convolution kernels in “Convs-Caps” for three residual-structure cascade 1 × 1 convolution kernels. “SF-Caps” identifies the model that substitutes convolutions for spatial dynamic filters, and “CF-Caps” identifies the model that substitutes convolutions for channel dynamic filters. “DDF-Caps” identifies the model that uses a residual-structure DDF, as shown in Fig.1. “DDF-MCE-Caps” identifies the model that adds the multilevel capsule encoding module to the “DDF-Caps” model, where the aggregation method is divided into a weighted matrix and a nonweighted matrix for analysis. “DDF-MCE-Pool” identifies the model that adopts the conventional maxpooling method after the multilevel capsule encoding module. “DDF-MCE-CPool” identifies the model that uses a weighted matrix and a nonweighted matrix for analysis.
that adopts our cap-sorting pooling method in “DDF-MCE-Caps” mode, which is also called MCWANet.

From the ablation test results, some conclusions can be drawn. First, among the top five models, “DDF-Caps” has the best behavior from using the residual-structure DDF. When only the spatial dynamic filter is used, the model’s performance decreases significantly, and when only the channel dynamic filter is used, the performance of the model is improved by 0.96% on the AID dataset and 0.98% on the NWPU-RESISC45 dataset. When the spatial and channel dynamic filters are used at the same time, the performance of the model is improved on the two datasets by 1.33% and 1.46%, respectively. These results show that joining the spatial and channel dynamic filters is helpful in the learning of scene representation, and the local features are very useful for scene understanding.

The multilevel capsule coding module is the crucial part of end-to-end MCAWNet, which is used to aggregate the convolutional features of the intermediate layers. Next, we study the effects of the multilevel capsule coding module based on weight adjustment. It can be seen from Table 4 that compared with the “DDF-Caps” model, the multilevel capsule encoding module improves the OAs from both datasets. In particular, the “DDF-MCE-Caps” model with weight adjustment achieves more significant improvement. This means that different intermediate convolutional layers have different contributions to different scene objects, and different levels of feature detail can compensate for the information loss of high-level features. For the NWPU-RESISC45 dataset, the OA with the weight matrix is much higher than that without the weight matrix, which is up to 0.83%. These results show that weight adjustment can effectively remove feature redundancy and aid in learning differentiated scene representations.

When the new CSPool method is used in “DDF-MCE-CPool”, the model’s performance is increased for the two challenging datasets, and at the same time, the number of training parameters is lessened. Many small convolution kernels in MCAWNet replace the large convolution kernels in the original CapsNet. This reduces the number of parameters, improves the capability of depth feature extraction, and indicates that the CSPool method plays an active role in RSSC.

According to the experimental results, the residual-structure DDF module, multilevel capsule encoding module, and CSPool strategy can extract robust features representing most of the classified scenes. The strongest Pri-Caps are employed to structure the digital capsules to conduct classification. All these strategies help increase the capability of the original CapsNet.

V. CONCLUSION

In this paper, we proposed a novel CapsNet, named MCWANet, in which a new multilevel capsule encoding module and a new capsule sorting pooling method are incorporated to improve RSSC accuracy by making use of the superiority of CapsNet, the advantages of the residual DDF block, the weighted capsule aggregation strategy, and the new CSPool method. The proposed MCWANet adopted a deep capsule architecture, which has good application prospects for extracting capsule features and restoring high-quality and strong semantic feature representation. By weighted integration of the DDF features of intermediate layers, the MCWANet enhanced feature representation functionality and robustness. Moreover, by incorporating the CSPool module, the MCWANet largely preserved descriptive information while remaining computationally and memory efficient. Experimental results on two challenging datasets clearly show that the MCWANet can produce competitive results compared with those from a set of existing algorithms. The highest OAs for the two datasets were 95.89% and 92.15%, respectively. Our future work will embed a fine-tuning strategy into our end-to-end classification framework to further improve its classification performance.

ACKNOWLEDGMENT

The authors would like to thank the anonymous reviewers for their valuable comments on the paper and thank the builders of the datasets.

REFERENCES

[1] 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.

[2] Y. Liu, Y. Zhong, and Q. Qin, “Scene classification based on multiscale convolutional neural network,” IEEE Trans. Geosci. Remote Sens., vol. 56, no. 12, pp. 7109–7121, Jul. 2018.
[3] X. Tang, Q. Ma, X. Zhang, F. Liu, J. Ma, and L. Jiao, “Attention consistent network for remote sensing scene classification,” IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens., vol. 14, pp. 2030–2045, 2021.

[4] 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. 16–26, Jan. 2020.

[5] B. Luo, S. Jiang, and L. Zhang, “Indexing of remote sensing images with different resolutions by multiple features,” IEEE J. Sel. Topics Appl. Earth Obs. Remote Sens., vol. 6, no. 4, pp. 1899–1912, Aug. 2013.

[6] C. Chen, B. Zhang, H. Su, W. Li, and L. Wang, “Land-use scene classification using multi-scale completed local binary patterns,” Signal Image Video Process., vol. 10, no. 4, pp. 745–752, 2016.

[7] G. Cheng et al., “Remote sensing image scene classification meets deep learning: Challenges, methods, benchmarks, and opportunities,” IEEE J. Sel. Topics Appl. Earth Obs. Remote Sens., 2020.

[8] 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.

[9] G. E. Hinton, A. Krizhevsky, and S. D. Wang, “Transforming auto-encoders,” in Proc. Int. Conf. Artif. Neural Netw., 2011, pp. 44–51.

[10] R. LaLonde and U. Bagci, “Capsules for object segmentation,” 2018, arXiv:1804.04241 [Online]. Available: https://arxiv.org/abs/1804.04241

[11] F. Deng, S. Pu, X. Chen, Y. Shiy, T. Yuan, and S. Pu, “Hyperspectral image classification with capsule network using limited training samples,” Sensors, vol. 18, no. 9, p. 3153, Sep. 2018.

[12] W. Zhao, J. Ye, M. Yang, Z. Lei, S. Zhang, and Z. Zhao, “Investigating capsule networks with dynamic routing for text classification,” 2018, arXiv:1804.00538 [Online]. Available: https://arxiv.org/abs/1804.00538

[13] S. Sabour, N. Frosst, and G. E. Hinton, “Dynamic routing between capsules,” in Proc. Int. Conf. Neural Inf. Process. Syst., Nov. 2017, pp. 3856–3866.

[14] J. Zhou, V. Jampani, Z. Pi, Q. Liu, and M.-H. Yang, “Decoupled dynamic filter networks,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2021, pp. 6647–6656.

[15] 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.

[16] Y. Yang and S. Newsam, “Bag-of-visual-words and spatial extensions for land-use classification,” in Proc. 18th SIGSPATIAL Int. Conf. Adv. Geograph. Inf. Syst. (GIS), San Jose, CA, USA, 2010, pp. 270–279.

[17] D. G. Lowe, “Distinctive image features from scale-invariant keypoints,” Int. J. Comput. Vis., vol. 60, no. 2, pp. 91–110, 2004.

[18] S. Lazebnik, C. Schmid, and J. Ponce, “Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories,” in Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR), vol. 2, Jun. 2006, pp. 2169–2178.

[19] B. Zhao, Y. Zhong, L. Zhang, and B. Huang, “The Fisher kernel coding framework for high spatial resolution scene classification,” Remote Sens., vol. 8, no. 2, pp. 157–176, Feb. 2016.

[20] F. Perronnin and C. Dance, “Fisher kernels on visual vocabularies for scene classification in remote sensing imagery,” IEEE Trans. Geosci. Remote Sens. Lett., vol. 10, no. 11, pp. 1894–1899, Jan. 2020.

[21] X. Liu, Y. Zhou, J. Zhao, R. Yao, B. Liu, and Y. Zheng, “Chinese convolutional neural networks for remote sensing scene classification,” IEEE Geosci. Remote Sens. Lett., vol. 16, no. 8, pp. 1977–1981, Aug. 2019.

[22] W. Zhang, P. Tang, and L. Zhao, “Remote sensing image scene classification using CNN-capsnet,” Remote Sens., vol. 11, no. 5, p. 494, 2019.

[23] 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.

[24] Y. Yu and F. Liu, “A two-stream deep fusion framework for high-resolution aerial scene classification,” Comput. Intell. Neurosci., vol. 2018, pp. 1–13, Jan. 2018.

CHUNYUAN WANG received the B.S. and M.S. degrees in instrument science and technology and the Ph.D. degree in information and communication engineering from Harbin Institute of Technology, Harbin, China, in 2004, 2006, and 2014, respectively. She is currently a Lecturer with Shanghai University of Engineering Science, Shanghai, China. Her current research interests include multi-source remote sensing image pre-processing, classification, and interpretation.
YANG WU received the B.S., M.S., and Ph.D. degrees in instrument science and technology from Harbin Institute of Technology, Harbin, China, in 2004, 2006, and 2011, respectively. He is currently a Senior Engineer with Shanghai Institute of Satellite Engineering, Shanghai, China. His current research interests include remote sensing information acquisition and object recognition.

YIHAN WANG received the B.S. degree in computer science and technology and the M.S. degree in computer applied technology from Donghua University, Shanghai, China, in 2003 and 2006, respectively. He is currently a Lecturer with the College of Electronic and Electrical Engineering, Shanghai University of Engineering Science, Shanghai. His current research interests include embedded systems, neural networks, and game theory.

YIPING CHEN received the Ph.D. degree in electronic engineering from Shanghai Jiao Tong University, China. He is currently a Lecturer with the Department of Electronic Information, Shanghai University of Engineering Science. He is engaged in the research and development of wireless communication, the IoT, and cloud computing. Previously, he was a Project Manager at Datang Telecom Technology and Industry Group, China.

YAN GAO received the Ph.D. degree in control theory and control engineering from Donghua University, in 2013. She is currently a Teacher with Shanghai University of Engineering Science. Her research interests include stability, synchronization, control of neural networks, and complex networks.

* * *