HDTFF-Net: Hierarchical Deep Texture Features Fusion Network for High-Resolution Remote Sensing Scene Classification

Wanying Song, *Member, IEEE*, Yifan Cong, Shiru Zhang, Yan Wu, *Member, IEEE*, and Peng Zhang, *Member, IEEE*

***Abstract*—Fusing features from different feature descriptors or different convolutional layers can improve the understanding of scene and enhance the classification accuracy. In this article, we propose a hierarchical deep texture feature fusion network, abbreviated as HDTFF-Net, aiming to improve the classification accuracy of high-resolution remote sensing scene classification. The proposed HDTFF-Net can effectively combine the shallow texture information from manual features and the deep texture information by convolutional neural networks (CNNs). First, for deeply excavating the multiscale and multidirectional shallow texture features in images, an improved Wavelet feature extraction module and a Gabor feature extraction module are designed by fully fusing the structural features into the backbone neural network. Then, to make the output texture features more discriminative and interpretative, we incorporate the above texture feature extraction modules into traditional CNNs (Tra-CNNs), and design two improved deep networks, namely Wave-CNN and Gabor-CNN. Finally, according to the Dempster-Shafer evidence theory, the designed Wave-CNN and Gabor-CNN are fused with the Tra-CNN by a decision-level fusion strategy, which can effectively capture the deep texture features by different feature descriptors and improve the classification performance. Experiments on high-resolution remote sensing images demonstrate the effectiveness of the proposed HDTFF-Net, and verify that it can greatly improve the classification performance.**

***Index Terms*—Convolutional neural network (CNN), deep Gabor features, deep wavelet features, Dempster-Shafer (D-S) evidential theory, remote sensing scene classification (RSSC).**

Manuscript received 18 May 2023; revised 4 July 2023; accepted 18 July 2023. Date of publication 25 July 2023; date of current version 11 August 2023. This work was supported in part by the Natural Science Foundation of China under Grant 61901358, Grant 62172321, and Grant 61871312, in part by the Outstanding Youth Science Fund of Xi'an University of Science and Technology under Grant 2020YQ3-09, in part by the Scientific Research Plan Projects of Shaanxi Education Department under Grant 20JK0757, in part by the Ph.D. Scientific Research Foundation under Grant 2019QDJ027, in part by the China Postdoctoral Science Foundation under Grant 2020M673347, in part by the Natural Science Basic Research Plan in Shaanxi Province of China under Grant 2019JZ-14, and in part by the Civil Space Thirteen Five Years Pre-Research Project under Grant D040114. *(Corresponding author: Wanying Song.)*

Wanying Song, Yifan Cong, and Shiru Zhang are with the Xi'an Key Laboratory of Network Convergence Communication, School of Communication and Information Engineering, Xi'an University of Science and Technology, Xi'an 710054, China (e-mail: songwanying0201@163.com; 20207223088@stu.xust.edu.cn; zhangshiru@xust.edu.cn).

Yan Wu and Peng Zhang are with the School of Electronics Engineering, Xidian University, Xi'an 710071, China (e-mail: ywu@mail.xidian.edu.cn; pzhang@xidian.edu.cn).

Digital Object Identifier 10.1109/JSTARS.2023.3298492

I. Introduction

With the ongoing development of the satellite imaging technologies, a great amount of high-resolution remote sensing (RS) images containing complex and detailed spatial structures are obtained by sensors carried on satellites or aircraft. It thus provides a solid source of data support for land-use/land-cover investigation, such as scene classification [1], change detection [2], [3], and RS image registration [4]. And remote sensing scene classification (RSSC) aims to automatically identify and classify different land cover types using image processing and machine learning algorithms. The land cover types generally include natural environments such as forests, grasslands, and water bodies, as well as human-made environments such as farmlands, airports, and factories. It is an important technique in RS image analyzing and interpreting, and has a wide range of applications in fields such as environmental monitoring, urban management, land planning, and disaster assessment [1], [2], [3], [4], [5].

Recently, RSSC has received widespread attentions and various methods [1], [2], [3], [4], [5], [6], [7] have been proposed, thanks to the continuous collection of available datasets and the latest advancements in data-driven algorithms. In literature, effective and discriminative feature representation plays a crucial role in RSSC. According to the way of feature learning, most of the state-of-the-art approaches can be divided into three categories: hand-crafted feature-based methods, unsupervised feature learning-based methods, and deep feature-based methods.

A. Hand-Crafted Feature-Based Methods:

Earlier algorithms are mostly dependent on low-level and hand-crafted features to extract the relevant information from RS images. These hand-crafted features are generally designed to capture specific characteristics in scene such as texture, shape, and color, and can be used to distinguish different land covers [8]. For example, Ojala et al. [6] utilized local binary pattern (LBP) to describe the gray-scale and the rotation invariant texture information in RS images, and Tan et al. [7] introduced a low-rank representation to automatically annotate multilabel RS images. In addition, Dalal et al. [9] used the histogram of oriented gradients (HOG) to capture the shape and texture information in images. And Kobayashi et al. [10] transformed the histogram features by the powerful Dirichlet model, thus improving the

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4.0/

classification performance of RS images. Luo et al. [11] used six traditional feature descriptors including GLCM, Gabor, SIFT, etc., for shallow feature extraction, and then combined them in various ways to enhance the classification performance of RS image. Nevertheless, these handcrafted features generally require domain expertise and a lot of time to manually design and implement the feature extraction process. And they are often designed for a specific task or dataset and may not generalize well to other datasets or tasks. Therefore, with the increasing complexity of RS images, these low-level and handcrafted features may grow weaker and may not adequately represent semantic information.

B. Unsupervised Feature Learning-Based Methods

Different from the supervised methods, the unsupervised ones [12], [13], [14], [15] do not require an enormous amount of labeled training data or are fine-tuned from the pretrained Convolutional neural networks (CNNs). The well-known generative adversarial networks (GANs), that is to learn a generative distribution of data through a two-player minimax game, is one of the most exciting unsupervised algorithm appearing in recent years. For example, Ma et al. [12] proposed a SiftingGan method that can well generate and manipulate labeled samples for data augmentation, thereby improving the performance of the scene classification baseline. Yu et al. [13] utilized an attention GAN for feature learning by incorporating contextual information through expanded convolutional layers and utilizing feature representation to form a content loss. And Kwak et al. [14] presented a novel unsupervised self-training with domain adversarial network by combing the adversarial training to alleviate spectral discrepancy problems with the self-training to automatically generate new training data in the target domain using an existing thematic map or ground truth data. In addition, Romero et al. [15] proposed the use of greedy layerwise unsupervised pretraining coupled with a highly efficient algorithm for unsupervised learning of sparse features. It is rooted on sparse representations and enforces both population and lifetime sparsity of the extracted features, simultaneously.

C. Deep Feature-Based Methods

Compared with the manual features, deep features have shown to be highly effective for RSSC tasks [5], [16], [17], [18], [19], [20], [21], [22], [23]. They are capable of capturing the subtle variations and learning the hierarchical information in RS scenes, thus enabling accurate and efficient analysis in RSSC. For example, Chen et al. [16] proposed an RSSC method via multibranch local attention network, where convolutional local attention module is embedded into all downsampling blocks and residual blocks of ResNet backbone. Xie et al. [17] introduced a scale-free CNN model for RSSC. It avoids losing critical information in image by pretrained CNNs, which deteriorates the classification performance. However, the CNNs generally employ a single receptive field, and thus may not capture the complex structure and the varied textures in high-resolution RS images. What is worse, these deep features obtained by CNNs may not have a specific physical meaning, and may lack of some interpretability.

Fortunately, recent studies [18], [19], [20], [21], [22], [23] have demonstrated that fusing the features by different convolutional neural networks can greatly improve the scene understanding and enhance the classification accuracy. For example, Wang et al. [18] designed a multilevel feature fusion network, which reduced the high-dimensional features through adaptive channel dimensionality for scene classification, and achieved high accuracy and stability. Chu et al. [21] designed a specifical feature fusion algorithm fusing the weights of the global GIST and local SIFT, and obtained better performance in RS scene classification. Ji et al. [22] localized the multiscale discriminative regions in RS images using an attention network, and then integrated the features learned from the localized regions by a classification network. Similarly, recent studies in [19], [20], [23], [24], [25], [26], [27], [28], [29], [30] fused the multilayer or multiscale features by specifical ways and all provided significant achievements, which well demonstrated that feature fusion is an efficient step in feature representation and image classification.

Generally, the deep feature-based methods can get better performance than the handcrafted feature-based unsupervised feature learning-based methods. However, they still have many shortcomings and challenges in real high-resolution RSSC, which are summarized as follows.

1) High-resolution RS images generally contain complex scenes and varied textures, and typically have high with-in class differences and between-class similarities. Therefore, it is often easier to misclassify these images, especially the artificial buildings, thus promoting us to excavate more discriminative deep features for accurate classification.
2) The deep features obtained by CNNs or multilayer networks generally have a common problem that it is difficult to provide a specific physical interpretation from the perspective of physical scattering mechanism in images, especially for RS images with complex structures and varied textures. Fortunately, as artificially designed feature descriptors, Gabor filter and Wavelet transformation have some similar operation modes or effects as that in CNNs, making it possible for us to fuse them into CNNs. In fact, similar effects [31] have been done and have obtained satisfying results. However, focusing on high-resolution RSSC, most of them cannot well integrate Gabor or Wavelet into CNNs or lack the effective preservation of details, but just employ them to generate texture features for CNN inputs or bring edge blurring.
3) What is more, features obtained by different feature descriptors may make different contributions to the final discrimination of categories. Employing only a single receptive field or directly concatenating the multilayer and multiscale features may limit the discriminative characteristic of the deep features to a certain extent. Thus, an effective fusion strategy for the hierarchical features is of great significance in classification. In fact, focusing on the fusion of hierarchical features, the powerful Dempster-Shafer (D-S) evidential theory has been successfully applied to synthetic aperture radar (SAR) image segmentation [32].

Motivated by the limitations and challenges aforementioned, we propose a hierarchical deep texture features fusion network based on D-S evidential theory to improve the performance of RSSC, named as hierarchical deep texture feature fusion network (HDTFF-Net) in this article. The main novelties and contributions of this article are summarized as follows.

1) The proposed HDTFF-Net model contains three subnetworks, namely Tra-CNN, Wave-CNN, and Gabor-CNN, to effectively capture the complex and varied textures within multiple receptive fields, thus providing greater discriminative features for classification. The Tra-CNN employs the powerful DenseNet-201 model as the basic framework, and the improved Wave-CNN and Gabor-CNN integrate the shallow texture feature into the deep feature extraction module for providing specific physical meanings for deep features.
2) In Wave-CNN, the discrete Wavelet decomposition is fused into CNN for enhancing the description of multilevel texture features. And a spatial attention mechanism is fused to pay attention to the locations of textures and details, thus providing greater weights for them.
3) In Gabor-CNN, we propose to replace the convolutional kernel of some convolutional layers in backbone model with a two-dimensional Gabor kernel, so that it can extract the texture features of different directions and scales in frequency domain and thus enhance the deep texture feature representation.
4) Considering the difference and relationship between the hierarchical deep features obtained by Wave-CNN, Gabor-CNN, and Tra-CNN, we propose an effective decision-level fusion strategy based on the D-S evidential theory. Thus, every subnetwork can well make its contribution to the final inference of the attributive class. Extensive comparisons and ablation experiments on NWPU-RESISC45, PatternNet38 and AID30 datasets demonstrate the effectiveness of HDTFF-Net in RSSC.

The rest of this article is organized as follows: Section II provides a brief review of popular CNN models. Section III details the proposed HDTFF-Net framework, containing the Gabor-CNN, the Wave-CNN, and the fusion strategy by D-S evidential theory. Finally, the RS scene classification results and analysis on several public datasets are given in Section IV to validate the effectiveness of HDTFF-Net, and Section V concludes the article.

II. Related Work

Deep learning employs artificial neural networks (ANNs) containing multiple layers to solve complex problems. It learns features by training a neural network on a large amount of training samples, and then makes predictions or classifications on testing samples. CNN [25], [33], a special type of ANN, is commonly used in deep learning. It can automatically learn features from raw data without manual feature extraction, making it more suitable for complex images, voice, and other fields. Due to its remarkable performance, CNN has occupied a dominant position in multiple application fields, including face detection, disease classification, image segmentation, and scene classification. Most recently, many typical CNNs, such as LeNet, AlexNet, VGGNet, and ResNet have appeared in the field of image understanding and recognition. In literature, these models have also achieved excellent results in RS scene classification.

A typical CNN [33] consists of several key components including convolutional layers, pooling layers, fully connected layers, and softmax layer, which are detailed in the following. Input data is propagated layer by layer and then the class probability is obtained through the fully connected layer.

1) The convolutional layer refers to a module that extracts features by applying a set of learnable filters to perform convolution operations on input data. The convolution operation is the sum of the element-wise products of the filter and the input data. Since the parameters of the filter are learnable, the convolutional layer can automatically identify the important features from inputs, and thus realize efficient feature extraction. After the convolutional layer, an activation function, such as Sigmoid, ReLU and tanh, will be used to introduce nonlinearity to better fit the data.
2) The pooling layer aims to divide the input image into rectangular regions called pooling filters. Each filter aggregates the pixel values within the window, typically by computing the maximum or average value. This can help to reduce the computational cost of subsequent layers in the network.
3) The fully connected layer is responsible for mapping the features extracted by the previous layers to the final output or prediction. In the fully connected layer, every output feature map from the previous layer will be flattened into a one-dimensional vector ξ_s.
4) Finally, the obtained feature vector ξ_s will be fed into the softmax layer, which is used to transform the outputs of the previous layer into a probability distribution over the classes $k \in \{1, 2, \ldots, K\}$

$$p\left(x_s = k|\xi_s, W_s\right) = e^{W_{sk}^T \xi_s} \Big/ \sum_{i=1}^{K} e^{W_{si}^T \xi_s} \tag{1}$$

where x_s indicates the label of image s, and W_s denotes the weight matrices of softmax function.

III. HDTFF-Net Model for RSSC

According to the discussions above, we can conclude the following:

1) Texture representing the natural spatial variation in the backscatter associated with the variability of targets is a crucial and ubiquitous factor in classification, and it is easy to misclassify these images with varied textures.
2) Most of the traditional CNNs may not have specific physical meanings, especially for high-resolution RS images.
3) Employing a single receptive field or directly concatenating the multilayer features may limit the discriminative characteristic of the fusion features to a certain extent.

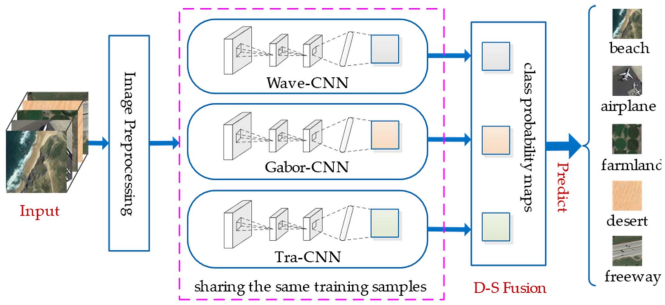

Fig. 1. Architecture of the proposed HDTFF-Net.

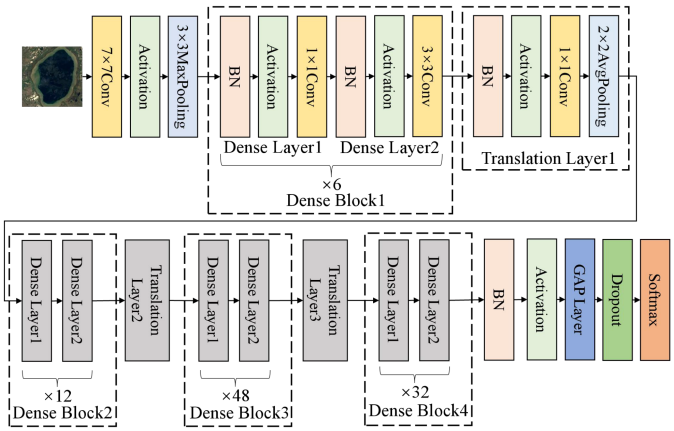

Fig. 2. Structure of Tra-CNN.

Therefore, we are inspired to propose an efficient hierarchical deep texture features fusion network, abbreviated as HDTFF-Net, for deep texture features extraction and high-resolution RSSC. As illustrated in Fig. 1, different from the existing multilevel CNNs fusing deep features from multiple convolutional layers, the proposed HDTFF-Net model consists of three subnetworks, that is, Wave-CNN for learning deep Wavelet features, Gabor-CNN for learning deep Gabor features, and Tra-CNN for learning deep gray features, which will be detailed in the following sections.

First, the HDTFF-Net model performs image enhancement, containing random translation, invertion, rotation, enlargement and minification, on input RS images to make the dataset four times larger than the original one and thus effectively improve the generalization of HDTFF-Net. Then, all measured RS images are resized to $224 \times 224 \times 3$, and then are normalized by $(x - x_{\min})/(x_{\max} - x_{\min})$.

Second, the preprocessed datasets are respectively inputted into the designed Wave-CNN, Gabor-CNN, and Tra-CNN. In HDTFF-Net, each subnetwork is trained on the same data and then their outputs are fused in parallel. And the designed Wave-CNN and Gabor-CNN can effectively consider the low-level texture features in deep learning, thus enhancing the interpretability of deep textures from the perspective of specific physical meaning, and providing greater discriminative features for scene classification. Furthermore, every subnetwork learns its special type of deep features from raw RS images, which can guarantee the effective extraction of textures and detail information more comprehensively.

Finally, the HDTFF-Net aims to estimate the class labels of all images belonging to. However, due to the differences and relationships between the obtained deep features, directly maximizing or integrating the class probability maps by Wave-CNN, Gabor-CNN, and Tra-CNN cannot guarantee their respective contributions to the final inference of the attributive class. Thus, we propose to fuse them according to the D-S evidential theory, which will also be detailed in the following.

A. Tra-CNN for Learning Deep Gray Features

As provided in [34], as a densely connected network model with front and back layers, DenseNet has fewer parameters and stronger ability of deep feature extraction and loss transfer. And it helps to address the vanishing gradient problem caused by the deepening of CNN network layers. The feature extraction module of DenseNet generally contains four Dense Blocks and three Transition Blocks. And the formula of Dense Block is defined as follows:

$$x_L = H_L\left[x_0, x_1, \ldots, x_{L-1}\right] \tag{2}$$

where x_L is the output of layer L, H_L is the nonlinear transformation function, and the output feature map dimension of the network layer before L is kept consistent with x_L by batch normalization.

In literature [34], according to the depth of network, there are DenseNet-121, DenseNet-169, and DenseNet-201. Considering the complex scene in high-resolution RS images, DenseNet-201 with deeper network structure is employed to construct the Tra-CNN, which is shown in Fig. 2. As shown in Fig. 2, the input image with a size of $224 \times 224 \times 3$ is preferentially passed through a 7×7 convolutional kernel to extract the shallow information. Next, the feature map is downsampled by a 3×3 maximum pooling layer, and then the feature results will pass through four dense blocks and three translation layers to obtain the deep features. Here, the transition layer is mainly used to reduce the channel number and the size of the output feature map of the dense block and then to pass the feature map to the next dense block. Finally, a nonlinear activation is performed by BN layer and ReLU activation layer, a global average pooling layer is used to integrate the features, a Dropout layer is used for network connection with consequent cropping to mitigate the occurrence of overfitting, and a softmax layer is used to output class probabilities for classification.

B. Wave-CNN for Learning Deep Wavelet Texture Features

1) Wavelet Transform: Wavelet transform [35] utilizes wavelet basis functions to analyze and represent signals, images and data. These basis functions are theoretically derived from a wavelet called the mother wavelet by translation and dilation operations. It provides a frequency window and a time window that can be modulated. In the field of image processing, discrete Wavelet transform (DWT), consisting of filtering and downsampling, is generally used. Given an image $\boldsymbol{x}$, it passes

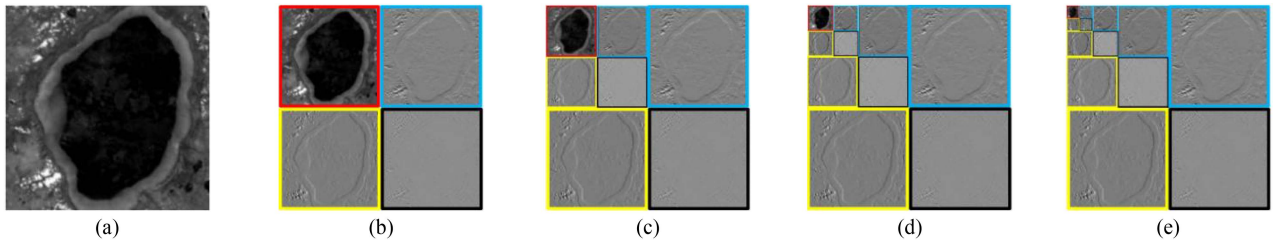

Fig. 3. Example of DWT of different levels. (a) R component of the original image. (b) One-level decomposition result. (c) Two-level decomposition result. (d) Three-level decomposition result. (e) Four-level decomposition result. Note that the red patches refer to the low-frequency component $\boldsymbol{x}_{LL}$, the blue, yellow, and black patches refer to the high-frequency detail components $\boldsymbol{x}_{LH}$, $\boldsymbol{x}_{HL}$, and $\boldsymbol{x}_{HH}$.

through four filters of equal size, i.e., low-pass filter f_{LL}, and high-pass filters, f_{LH}, f_{HL}, f_{HH}, and then the image $\boldsymbol{x}$ will be divided into a low-frequency approximate component x_{LL} and three high-frequency detail components, i.e., $\boldsymbol{x}_{LH}$, $\boldsymbol{x}_{HL}$, and $\boldsymbol{x}_{HH}$. Then, it could continuously decompose the approximate component to capture finer details. Thus, DWT allows for a multiresolution analysis of an image, which means that different scales of the image can be analyzed separately [35].

In this article, for the implementation of two-dimensional (2-D) DWT, the Haar Wavelet function is adopted. Specifically, each feature map is down-sampled by a factor of 2 when it passes through four mutually orthogonal filters, and then four different components will be generated. These four filters can be defined as follows:

$$\boldsymbol{f}_{LL} = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}, \quad \boldsymbol{f}_{LH} = \begin{bmatrix} -1 & -1 \\ 1 & 1 \end{bmatrix},$$
$$\boldsymbol{f}_{HL} = \begin{bmatrix} -1 & 1 \\ -1 & 1 \end{bmatrix}, \quad \boldsymbol{f}_{HH} = \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}. \tag{3}$$

Then, the DWT operation [41] can be determined by the following:

$$\boldsymbol{x}_{LL} = (f_{LL} * \boldsymbol{x}) \downarrow_2, \quad \boldsymbol{x}_{LH} = (f_{LH} * \boldsymbol{x}) \downarrow_2,$$
$$\boldsymbol{x}_{HL} = (f_{HL} * \boldsymbol{x}) \downarrow_2, \quad \boldsymbol{x}_{HH} = (f_{HH} * \boldsymbol{x}) \downarrow_2 \tag{4}$$

where $*$ is convolutional operator, and $\downarrow_2$ is downsampling operator with the stride of 2. Moreover, according to the theory of Haar transform, the (i, j)th value of $\boldsymbol{x}_{LL}$, $\boldsymbol{x}_{LH}$, $\boldsymbol{x}_{HL}$ and $\boldsymbol{x}_{HH}$ after 2-D Haar transform can be written as follows:

$$\begin{aligned}
\boldsymbol{x}_{LL}(i,j) &= \boldsymbol{x}(2i-1, 2j-1) + \boldsymbol{x}(2i-1, 2j) \\
&\quad + \boldsymbol{x}(2i, 2j-1) + \boldsymbol{x}(2i, 2j) \\
\boldsymbol{x}_{LH}(i,j) &= -\boldsymbol{x}(2i-1, 2j-1) - \boldsymbol{x}(2i-1, 2j) \\
&\quad + \boldsymbol{x}(2i, 2j-1) + \boldsymbol{x}(2i, 2j) \\
\boldsymbol{x}_{HL}(i,j) &= -\boldsymbol{x}(2i-1, 2j-1) + \boldsymbol{x}(2i-1, 2j) \\
&\quad - \boldsymbol{x}(2i, 2j-1) + \boldsymbol{x}(2i, 2j) \\
\boldsymbol{x}_{HH}(i,j) &= \boldsymbol{x}(2i-1, 2j-1) - \boldsymbol{x}(2i-1, 2j) \\
&\quad - \boldsymbol{x}(2i, 2j-1) + \boldsymbol{x}(2i, 2j)
\end{aligned} \tag{5}$$

where $i = 1, 2, \ldots, V/2$ and $j = 1, 2, \ldots, H/2$ are the coordinates of each pixel in image. Thus, the width and height of the output component of each level after DWT will be 1/2 of the input image. Then, according to the characteristics of multiresolution analysis of DWT, the sub-band image x_{LL} containing low-frequency approximate information will be continuously decomposed by DWT.

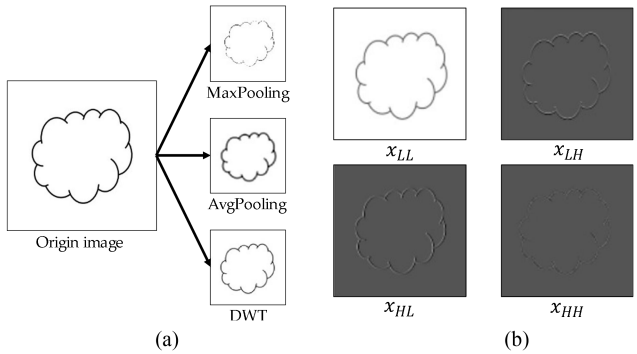

Fig. 4. Results of different pooling methods. (a) Outputs of different downsampling methods. (b) Four different components obtained by 2-D DWT.

To demonstrate the effectiveness of 2-D DWT in capturing different levels of texture features, we provide an example of an RS image decomposition, as displayed in Fig. 3. As shown in Fig. 3(a), the original RS image's R component is first extracted. Fig. 3(b) shows the first-level decomposition results, where the low-frequency component x_{LL}, enclosed in the red rectangle, mainly preserves the main information of the input feature map. Meanwhile, the structural information in image in horizontal, vertical, and diagonal directions can be obtained from different high-frequency sub-bands x_{LH}, x_{HL}, and x_{HH}, respectively shown in blue, yellow, and black rectangles. By continuously decomposing the low-frequency component, higher level detail structures can be obtained, as shown in Fig. 3(c) and (d) for the second and third level of decomposition. Based on the demonstrated capability of DWT in capturing multiscale texture features, we plan to incorporate it into deep learning frameworks to excavate deeper texture features from RS image analysis.

As we know, traditional CNNs generally employ max pooling and average-pooling for downsampling, which can effectively reduce the image size and suppress the noise. However, they easily get into such problems as shown in Fig. 4(a). Depending on the data, the Max pooling may overemphasize on details, thus easily erasing some details in image especially when the pixel

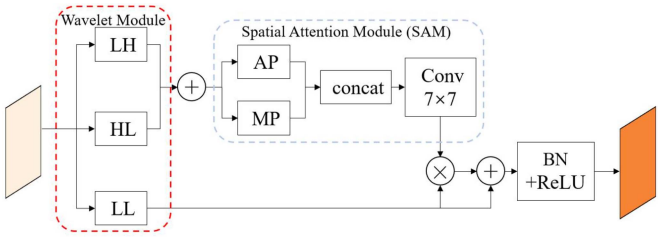

Fig. 5. Discrete wavelet decomposition fused by spatial attention module (DWT-SAM).

values in background are close to that in details. And depending on the data, the average pooling performs mean filtering and downsampling on the image, thus obviously diluting the pertinent details in image.

Therefore, this article aims to design a more suitable module that not only has the dimension reduction function of the pooling layer but also can well retain textures in image.

Here, we obtain four different components by 2-D DWT, as shown in Fig. 4(b). It is clear that the main information in image is well preserved in low-frequency component, and the details are well preserved in different high-frequency components. In conclusion, the 2-D DWT can not only obtain the same low-frequency component as CNN, but also well maintain better details than CNN.

Inspired by the discussions above, we mean to improve the traditional CNN by using DWT to replace traditional pooling layers. Considering the advantages of DWT, the fusion of DWT into CNN can help to enhance the description of texture features, and further improve the performance of scene classification. In the next section, we will provide a detailed introduction of how to integrate DWT into deep learning models.

2) Construction of Wave-CNN: As discussed above, the input RS images can be decomposed into low-frequency and high-frequency components by 2-D DWT. Generally, the low-frequency components contain the smoothing information, and the high-frequency components contain the varied textures and details in image. Considering that DWT has the characteristic of multiresolution analysis and can better preserve the textures and details in classification, the Wave-CNN model is constructed based on DenseNet. In Wave-CNN, the multilevel Wavelet texture features are first decomposed and then the redundant information is filtered. In addition, a spatial attention module (SAM) is fused to pay attention to the locations of textures, which is shown in Fig. 5, providing greater weights for textures. And the cascade fusion is used to strengthen the reuse of features and improve the classification ability of Wave-CNN. In the following, Fig. 6 illustrates the structure of the proposed Wave-CNN.

1) All of the maximum and average pooling layers in the traditional DenseNet are replaced by Wavelet decomposition to capture the multiscale texture features in RS image. As we know, the input image in DenseNet first passes through a convolutional module and a 3×3 maximum pooling layer, and then completes the down-sampling for subsequent propagation in the Translational Block by an average pooling layer. In the proposed Wave-CNN, these pooling layers are replaced with 2-D Wavelet transform layers to obtain the low-frequency component x_{LL} and the high-frequency ones x_{LH}, x_{HL}, x_{HH}.

   Empirically, the diagonal component x_{HH} contains less effective information in RS image and generally brings more redundant noise. Thus, for reducing the negative impact of x_{HH} on texture feature extraction, this article removes it and only retains x_{LL}, x_{LH}, and x_{HL}. Specifically, the low-frequency component x_{LL} is used as the result of downsampling the input feature map, and the features are nonlinearly activated by adding a BN layer and a ReLU activation layer. And the x_{LH} component and x_{HL} component are fused into a high frequency detail texture component H by Add layer.

2) *Multistage Wavelet feature extraction:* The 2-D DWT is employed to extract the first-stage Wavelet components $(x_{LL1\_1}, x_{LH1\_1}, x_{HL1\_1})$, and $x'_{\text{dense}(1)}$ is assumed to be the output of Dense Block1. Then, we will obtain the first-stage Wavelet feature Wavelet_{1_1} by the proposed DWT-SAM module as shown in Fig. 5. Next, the first-stage Wavelet component x_{LL1_1} will be further decomposed into the second-stage Wavelet components $(x_{LL1\_2}, x_{LH1\_2}, x_{HL1\_2})$ by 2-D DWT, and then the second-stage Wavelet feature Wavelet_{1_2} will be obtained by the same method as Wavelet_{1_2}. Thus, the Dense Block N will perform a total of $N-4$ Wavelet decompositions.

3) *Discrete Wavelet decomposition fused by spatial attention module (DWT-SAM):* To pay attention to the locations of textures, an SAM is fused into the traditional 2-D DWT, which can effectively provide greater weights for textures. Here, the feature maps pass through an average pooling layer and a maximum pooling layer, respectively, along the channel axis, and then we can obtain two feature maps with a dimension of 1. After the superimposition by the Concat channel, the spatial weight matrix will be obtained by a 1×1 convolutional layer. The spatial matrix is dotted with the x_{LL} component and is accessed in the network layer using a shortcut connection. Fig. 5 shows the structure of DWT-SAM, and the module formula is defined as follows:

$$x_H = x_{LH} + x_{HL} \tag{6}$$

$$x_S = x_{LL} \otimes f_{\text{Sigmoid}}\left(\text{conv}^1_{7\times7}\left(\text{concat}\begin{pmatrix}\text{AvgPool}\,(x_H),\\ \text{MaxPool}\,(x_H)\end{pmatrix}\right)\right)$$

$$= x_{LL} \otimes f_{\text{Sigmoid}}\left(\text{conv}^1_{7\times7}\left(\text{concat}\left(F^S_{\text{Avg}}, F^S_{\text{Max}}\right)\right)\right) \tag{7}$$

$$\text{Wavelet} = \text{BN}\left(f_{\text{ReLU}}\left(x_{LL} \oplus x_S\right)\right) \tag{8}$$

where f_{ReLU} denotes the activation function, BN represents the batch normalization, and x_S represents the spatial attention module composed of components. AvgPool and MaxPool respectively represent average and maximum pooling. Conv is the convolutional layer used for channel transformation.

4) *Multilevel Wavelet cascade fusion:* In order to incorporate the multilevel Wavelet texture features into the initial network, a multilevel Wavelet cascade fusion is proposed with reference

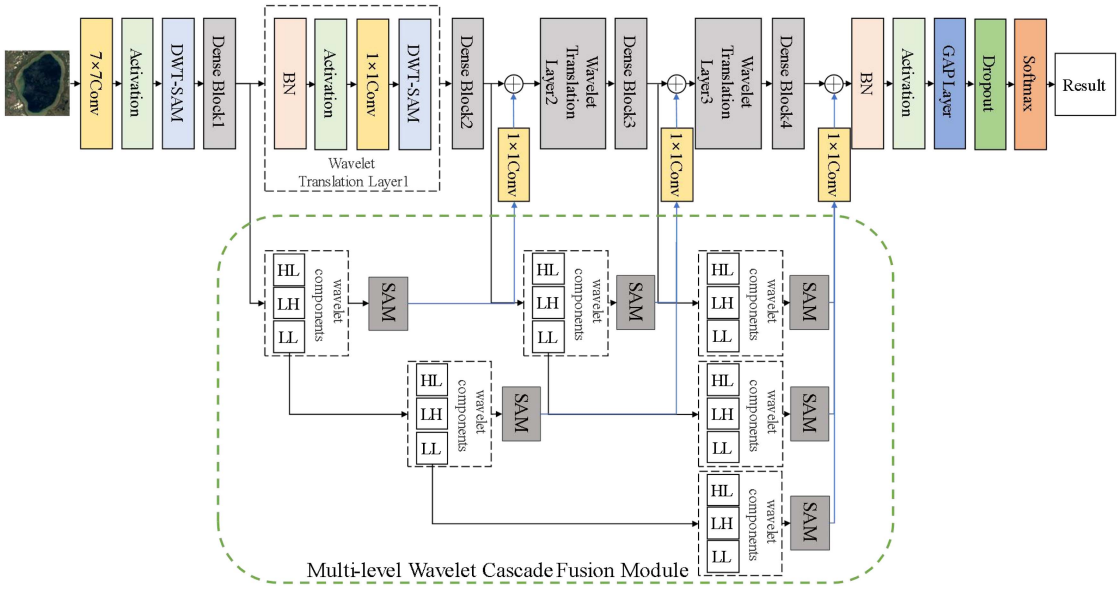

Fig. 6. Structure of Wave-CNN.

to DenseNet structure characteristics. It aims to allow the output features of each block in the network to additionally obtain the Wavelet texture output of the previous layer, thus achieving further enhancement of the texture features. Additionally, the cascade fusion is used to strengthen feature reuse and improve the classification ability of Wave-CNN. As illustrated in Fig. 6, starting from Dense Block2, the output features of each Dense Block will be added to the multilevel Wavelet features of the previous Dense Block, and the Dense Block N is integrated into the Wavelet feature output of each of the $N-1$ previous Dense Blocks. The fusion formula is defined as follows:

$$x'_{\text{dense}(n)} = H_n\left[x_{\text{dense}(n)}, \text{Wavelet}_{n-1}, \ldots, \text{Wavelet}_1\right], \quad n \geq 2 \tag{9}$$

where $x_{\text{dense}(n)}$ and $x'_{\text{dense}(n)}$, respectively, denotes the input and output features of Dense Block n in the backbone model. $Wavelet_{n-1}$ denotes the Wavelet feature of Dense Block $n-1$, and H_n is the nonlinear transformation function. And the feature dimension of the previous Wavelet feature is kept consistent with $x_{\text{dense}(n)}$ by batch normalization, ReLU activation function, and convolutional layer.

C. Gabor-CNN for Learning Deep Gabor Texture Features

1) Gabor Features: Gabor [37] filter employs several Gabor kernels to perform short-time windowed Fourier transform on the signal in the frequency domain to filter the input image. And then it extracts the information that matches the frequency range of the filter, and suppresses the information that exceeds the frequency range. In the field of image processing, the formula of 2-D Gabor filter is defined as follows:

$$g\left(x, y, \lambda, \theta, \varphi, \sigma, \gamma\right) = e^{\left(-\left(\frac{x'^2+\gamma^2 y'^2}{2\sigma^2}\right)\right)} \cdot e^{\left(i\left(2\pi\frac{x'}{\lambda}+\varphi\right)\right)} \tag{10}$$

$$g_R\left(x, y, \lambda, \theta, \varphi, \sigma, \gamma\right) = e^{\left(-\left(\frac{x'^2+\gamma^2 y'^2}{2\sigma^2}\right)\right)} \cdot \cos\left(2\pi\frac{x'}{\lambda}+\varphi\right) \tag{11}$$

$$g_I\left(x, y, \lambda, \theta, \varphi, \sigma, \gamma\right) = e^{\left(-\left(\frac{x'^2+\gamma^2 y'^2}{2\sigma^2}\right)\right)} \cdot \sin\left(2\pi\frac{x'}{\lambda}+\varphi\right) \tag{12}$$

$$x' = x\cos\theta + y\sin\theta,$$
$$y' = -x\sin\theta + y\cos\theta \tag{13}$$

where λ is the wavelength. We can obtain the feature maps of different scales by adjusting λ. θ denotes the direction of the Gabor kernel function. And we can obtain the characteristic graphs in different directions by changing θ. φ is the phase offset, and it is generally set to a constant. σ is the spatial aspect ratio and γ represents the standard deviation of the Gabor function's Gaussian factor. The real component $g_R(.)$ and imaginary component $g_I(.)$ are obtained by Euler Formula. $g_R(.)$ is used to eliminate redundant features and smooth the image, and $g_I(.)$ is to extract the edge information in image.

In order to visually observe what the CNN and Gabor has learned from image data, we visualize the pretrained DenseNet kernels and Gabor kernels in the first few layers, as shown in Fig. 7(a) and (b). It can be seen that: 1) Many learned filters by CNN and Gabor are similar, laying a good foundation for the replacement of convolution kernel by Gabor kernel. 2) The learned filters by Gabor show a regular characteristic of direction in the shallow layers, however the learned filters by CNN do not.

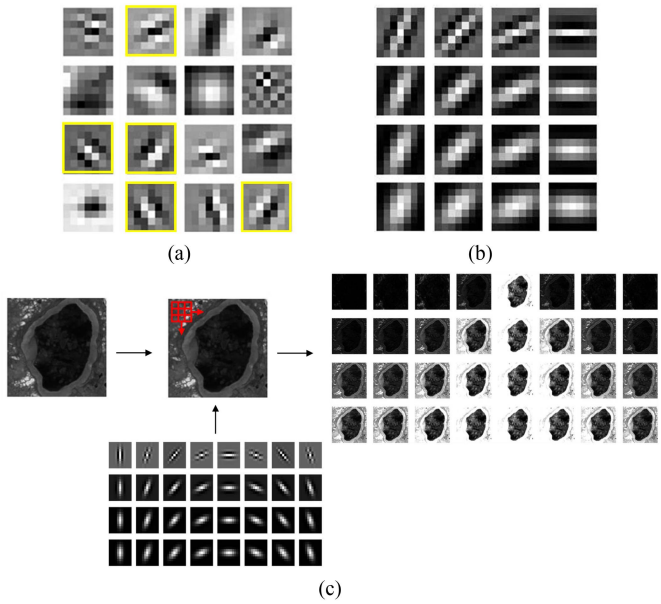

Fig. 7. (a) CNN kernels (b) Gabor kernels. (c) Texture feature extraction by 2-D Gabor filter.

That is to say, filters by Gabor with different orientations can well help to analyze the varied edges and textures in images. What is more, Gabor filter has another advantage that its parameters can be artificially adjusted, thus enhancing its ability to decompose signals in different scales and directions. Here, Fig. 7(c) illustrates the texture feature extraction by 2-D Gabor filter.

Thus, considering the advantages of Gabor filter, we are sure that fusing it into CNNs can help to enhance the robustness of deep features to the variations of orientation and scale in the shallow feature maps extracted by the first few convolution modules, and thus help to enhance the representation of deep texture features. It is the main motivation for us to construct the Gabor-CNN, which will be detailed in the following.

2) Construction of Gabor-CNN: Considering the advantages of 2-D Gabor filter on feature extraction, we propose to replace the feature extraction function of standard convolutional kernel with 2-D Gabor filter to place more emphasis on the extraction of textures in high-resolution RS images. So, it can effectively extract the texture features of different directions and scales in the frequency domain and thus enhance the deep texture feature representation. After filtering, Gabor feature maps will be activated by the ReLU function, and then be transferred to the next layer. Finally, the trainable convolutional kernel is added for updating the weights of the network in back propagation. In detail, we add a learning weighted filter bank after the Gabor convolutional layer, which is used to change the number of feature channels and obtain the final feature map of the entire module. The filter bank consists of convolutional kernel with a dimension of $C_I \times C_O$, where (C_I, C_O) denotes the number of channels in the input signature graph and the number of filter convolutional kernels, And the size of convolutional kernels is 1×1, which is consistent with the Gabor filter kernels. The GaborConv2D layer is obtained by passing the activated Gabor feature maps through a 1×1 convolutional layer, which is

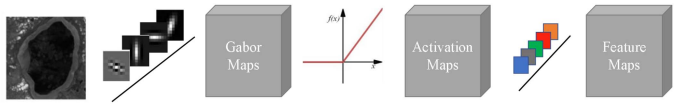

Fig. 8. Deep Gabor feature module.

defined as follows:

$$x_{l+1}^{j} = \mathrm{Conv}_{1\times1}^{n} \otimes f_{\mathrm{ReLU}}\left(\mathrm{Gabor}_{k\times k}^{j} \otimes x_{l}^{m}\right) \tag{14}$$

where x_l^m is the input feature map, and m is the number of channels. $\mathrm{Gabor}_{k\times k}^{j}$ is the $k \times k$ Gabor convolutional layer, and j is the number of Gabor kernels. And the value of j depends on the direction and scale of Gabor kernel. Assume that the direction of Gabor kernel is d and the scale is s, $j = s \times d$. The ReLU function f_{ReLU} is used to nonlinearly activate the Gabor features. $Conv_{1\times1}^{n}$ is a standard convolutional kernel of 1×1, and n is its number of kernels. Through (14), we can obtain the output feature map x_{l+1}^{n}, and n is the number of channels.

In a standard convolutional layer, for j standard $k \times k$ convolutional kernels, the number of trainable parameters of this convolutional layer is $(k \times k \times m + 1) \times j$ with the number of input feature maps m. However, in the proposed GaborConv2D layer, it is not necessary to update the parameters of kernels, and thus the number of trainable parameters is $(j + 1) \times n$.

Fig. 8 shows the deep Gabor feature module. As shown in Fig. 8, the network needs to update the parameters of each layer by backpropagation after propagating through the forward direction of the GaborConv2D layers. Thus, only the latter convolutional layer with a size of 1×1 can be learned. Therefore, this layer only needs to update the learned convolutional kernel Conv in the backpropagation of the neural network, and let the loss be L, the learning rate be η. And the gradient update equation is as follows:

$$\mathrm{Conv}' = \mathrm{Conv} + \eta \cdot \frac{\partial L}{\partial\left(\mathrm{Conv}\right)}. \tag{15}$$

The structure of Gabor-CNN is illustrated in Fig. 9. In Gabor-CNN, some of the standard convolutional layers in the DenseNet network are replaced with GaborConv2D. It sets the number of Gabor kernels to 40 (extracting texture feature maps in 8 directions and 5 scales) and varies the number of channels by 1×1 convolutional layers to ensure matching the number of channels with the baseline. The first layer in Gabor-CNN uses GaborConv2D with 7×7 Gabor kernels, and the stride is set to 1, so as to maintain the texture features in image. Besides, we use a 3×3 convolutional layer with the stride of 2 to extract deep features and reduce the size of the image. Finally, we use a 3×3 maximum pooling layer to reduce dimension.

Each Dense Layer in Dense Block contains two convolutional layers with different kernel sizes and different number of convolutional kernels. The former one contains 128 convolutional kernels, and the size of convolutional kernel is 1×1. It is often used to limit the dimension of feature maps and reduce the computational amount of subsequent convolution layer. And the later one generally contains 32 convolutional kernels, and

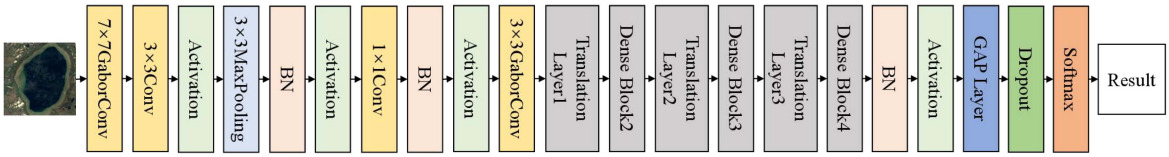

Fig. 9. Structure of Gabor-CNN.

TABLE I
PARAMETERS OF GABOR CONVOLUTIONAL KERNEL

Kernel_size	θ	λ	σ	γ
same as the original convolutional layer	$[0,\pi)$	$[2,6]$	1	0.5

the size of convolutional kernel is 3×3. It is to extract the deep features in image. The Dense Block1 mainly extracts the shallow features of the image, including corner points, texture, light and dark features, etc., and the Gabor convolutional kernel is mainly used for the extraction of details and texture features in image. Therefore, only the 3×3 convolutional kernels in Dense Block1 of original DenseNet model are replaced with Gabor convolutional kernels, and the later Dense Blocks still employ the standard convolutional kernel to extract deep features. And the parameters for each Gabor convolutional kernel are set as that in Table I.

D. Hierarchical Deep Texture Feature Fusion By D-S Evidential Theory

According to the discussions above, we can obtain several class probabilities by Wave-CNN, Gabor-CNN, and Tra-CNN. This section will focus on fusing the class probabilities and predicting the class labels of the input images. Considering the differences and relationships between the deep features by Wave-CNN, Gabor-CNN, and the traditional CNN (Tra-CNN), directly maximizing or integrating the class probability maps cannot guarantee the semantic description. Thus, according to the powerful D-S evidential theory [32], we propose a decision-level fusion to integrate the hierarchical deep features. In this way, the HDTFF-Net can fully combine the advantages of different deep features, and thus every subnetwork can well make its contribution to the final inference of the attributive class. At the same time, each sub-network in HDTFF-Net shares the same training dataset, which helps to reduce the requirements for data preparation and further to enhance the overall generalization of the HDTFF-Net in classification.

The class probability corresponding to the subnetwork in HDTFF-Net is regarded as the basic belief assignment (BBA) function in D-S evidential theory, which is defined as follows:

$$\begin{cases} m(\emptyset) = 0 \\ \sum_{A \subseteq \Theta} m(A) = 1 \end{cases} \tag{16}$$

where m is a BBA function. $\Theta = \{e_1, e_2, \ldots, e_L\}$ is a recognition frame containing L elements, and every element consists of a binary vector of length L. If one of the components takes the value of 1, the remaining ones will take the value of 0. A denotes the focal element within the recognition frame, indicating the class of scenes to which different images belong. And $m(A) > 0$ is satisfied, where $m(A)$ represents the strength of evidence in support for hypothesis A according to the specified evidence subject, and indicates the degree of confidence of the evidence in A. In HDTFF-Net, A contains only one element, and then $m(A)$ can be interpreted as a probability function, which is commonly referred to as the Bayesian BBA.

For each image prediction, the class probabilities m_1, m_2, and m_3 obtained by Wave-CNN, Gabor-CNN, and Tra-CNN can be considered as a set of BBAs. According to the D-S evidential theory, the hierarchical deep texture feature fusion is implemented as follows:

$$\begin{aligned} m(e_1) &= m_1 \oplus m_2 \oplus m_3(e_1) \\ &= \frac{m_1(e_1) m_2(e_1) m_3(e_1)}{\sum_{j=1}^{L} m_1(e_j) m_2(e_j) m_3(e_j)} \end{aligned} \tag{17}$$

where $\oplus$ expresses the class probability fusion of sub- networks in HDTFF-Net.

In conclusion, the HDTFF-Net with three subnetworks can fully capture the multiscale, multidirectional, and multitype deep features in high-resolution RS images, thus providing more discriminative features for classification.

IV. EXPERIMENTAL RESULTS AND ANALYSIS

A. Datasets Description for Experiments

In the experiments, three public high-resolution RS image datasets are used to demonstrate the performances of the proposed HDTFF-Net in RSSC. These three datasets all contain rich scene changes, leading to high within-class differences and between-class similarities and making them more challenging. And the descriptions are provided in the following.

NWPU-RESISC45 [38]: The NWPU-RESISC45 dataset is a diverse benchmark dataset for RSSC, containing 45 scene categories, including lake, farmland, airport, etc. Each category contains 700 images, resulting in a total of 31 500 images. The images were acquired by Google Earth and have a resolution of 256×256 pixels. The spatial resolution of the images varies between 0.25 and 30 m/pixel. Fig. 10 provides some example images of NWPU-RESISC45 dataset.

AID30 [39]: The aerial image dataset (AID) consists of 10 000 aerial images covering 30 classes, with each class containing about 300 to 400 RGB images with a resolution of 600×600 pixels captured by different sensors and platforms. The spatial resolution varies from 8 to 0.5 m/pixel. Some example images of AID30 dataset are shown in Fig. 11.

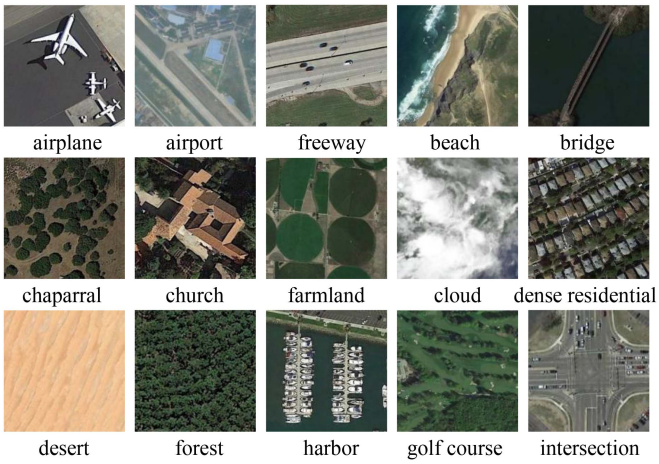

Fig. 10. Some example images of NWPU-RESISC45 dataset.

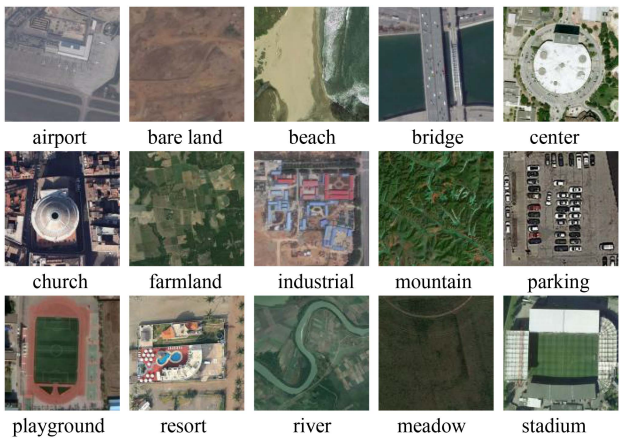

Fig. 11. Some example images of AID30 dataset.

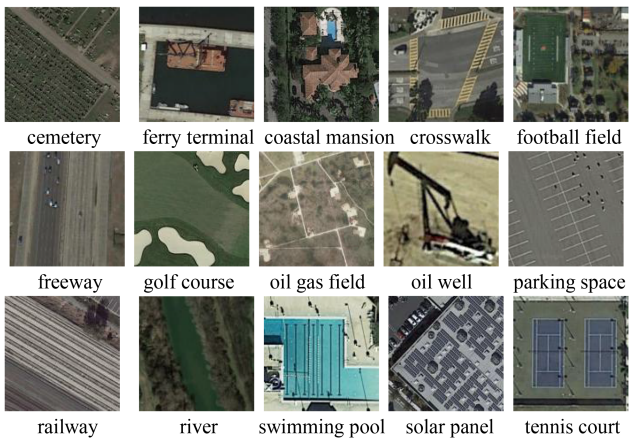

Fig. 12. Some example images of PatternNet38 dataset.

PatternNet38 [40]: The PatternNet38 was acquired from Google Map. It contains a total of 30 400 RGB images with a resolution of 256 × 256 pixels. The spatial resolution of the image ranges from 0.062 to 4.693 m/pixel. Each image in PatternNet38 is labeled with one of 38 land cover categories, including cemetery, oil gas field, tennis court, and so on. Fig. 12 shows some example images of PatternNet38 dataset.

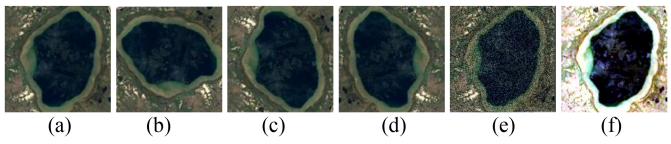

Fig. 13. Data Enhancement. (a) Original image. (b) Random rotation. (c) Horizontal flip. (d) Vertical flip. (e) Gaussian noise. (f) Contrast adjustment.

TABLE II
MODEL TRAINING PARAMETERS

Parameter	Pre-training Dataset	Batch size	Epoch	Loss Function	Learning Rate	Optimizer
Value	ImageNet	32	100 (Early-stop) +100	Sparse Categorical Crossentropy	0.001 +0.0001	SGD

B. Experiment Setting and Objective Evaluation

1) Data Enhancement: In the experiments, random rotation, horizontal flip, vertical flip, Gaussian noise, and contrast adjustment are used for data enhancement in order to increase the sample diversity and enhance the generalization of the proposed HDTFF-Net in RSSC. The first three types, that is random rotation, horizontal flip and vertical flip, belong to geometric transformations and are the most commonly used data enhancement methods. They mainly focus on the position bias in the training data, and can effectively solve the problem of insufficient training samples. However, the geometric transformation only considers the original position bias of samples, thus having limited ability in enriching the sample diversity of RS images. Therefore, we additionally employ Gaussian noise and contrast adjustment, belonging to color transformation, to rich the sample diversity for accurate classification. Fig. 13 shows the results of different enhancement methods.

2) Experiment Environment: We conduct the scene classification experiments on high- resolution RS images using a PC with a 11th Gen Intel(R) Core(TM) i9-11900F @ 2.50 GHz CPU, a NVIDIA GeForce RTX 3090 GPU and 32 GB memory. And the software environment is Python 3.7.16 + Tensorflow (GPU) 2.9.1.

3) Parameter Setting: The model training parameters are set as that in Table II, and each subnetwork in HDTFF-Net is pretrained on the ImageNet. The weight of the network layer after the Dense Block 4 is not updated. During the migration learning, the momentum of the optimizer SGD is set to 0.9, and the weight Decay is set to 0.0001. The first 100 rounds only update the weight of the network layer after the Dense Block 4. The Early-Stop strategy is adopted. When the loss of the test set does not decrease for five consecutive rounds, the training will automatically stop. Then, we will train the entire network for 100 epochs, and this training phase adopts a learning rate decay strategy. When the loss of the test set does not decrease for five consecutive rounds, the learning rate is reduced by 0.5, and the minimum learning rate is set to 0.000001.

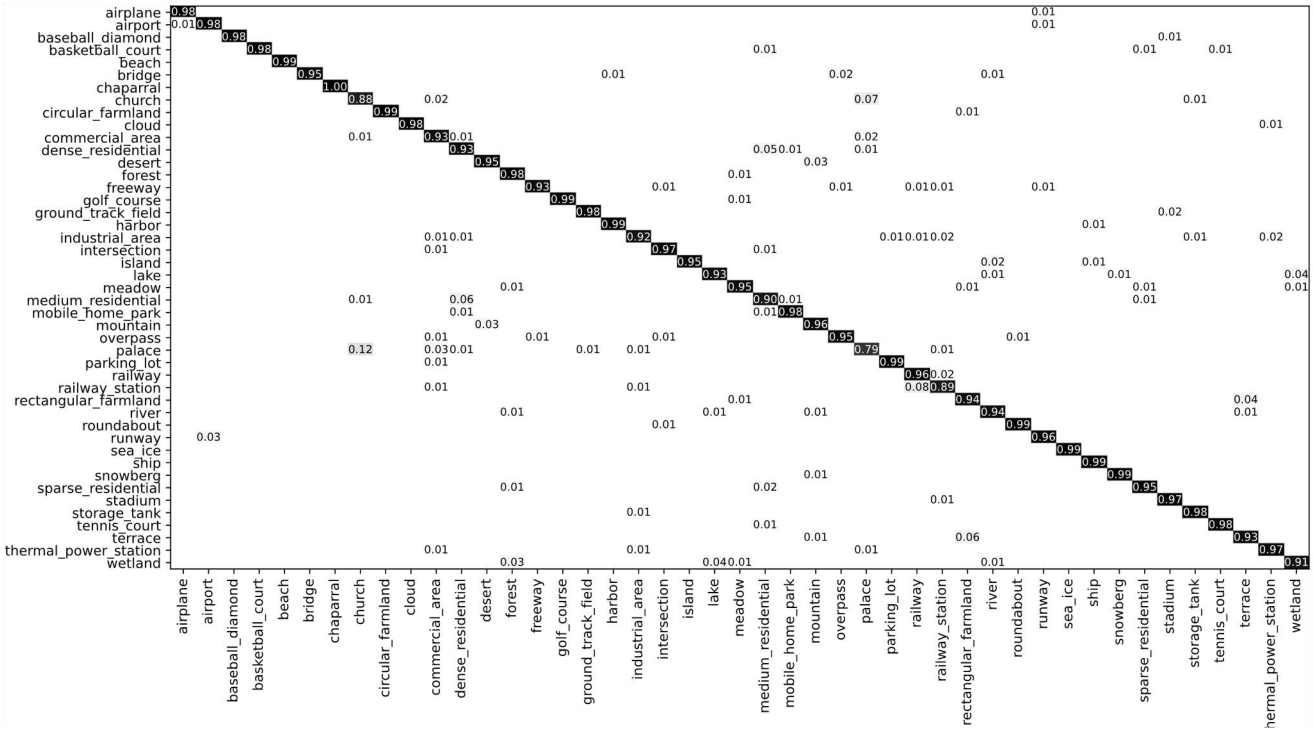

Fig. 14. Confusion Matrix on NWPU-RESISC45.

4) Experiment Evaluation Metrics: The classification accuracy and the confusion matrix are taken as the quantitative evaluations of scene classification results, calculated by comparing the classification maps with the reference labels. The specific definitions are provided as follows.

1) Accuracy (Acc) measures the percentage of correctly classified images out of all the images in the dataset. A higher value of accuracy indicates a better performance of the model in classification. Mathematically, it is expressed as follows:

$$\mathrm{Acc} = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{FP} + \mathrm{TN} + \mathrm{FN}} \tag{18}$$

where TP and TN, respectively, represent the number of positive samples that are correctly and incorrectly judged. FP and FN, respectively, represent the number of negative samples that are correctly and incorrectly judged.

2) The standard deviation σ represents the amount of variation or dispersion in a set of data values from the mean or average value, and is expressed as follows:

$$\mathrm{Acc}' = \frac{1}{n}\sum_{i=1}^{n} \mathrm{Acc}_i$$

$$\sigma = \sqrt{\frac{1}{n}\sum\nolimits_{i=1}^{n} \left(\mathrm{Acc}_i - \mathrm{Acc}'\right)^2} \tag{19}$$

where n is the number of experiments, Acc_i is the accuracy of the ith experiment, Acc' is the average accuracy of the experiments, and σ is the standard deviation of the accuracy of all experiments.

TABLE III
COMPARISON OF ACC AND σ ON NWPU-RESISC45

Method	*Acc* (%)± σ
Two-Stream Fusion [41]	83.02±0.14
SCCov [42]	92.10±0.25
MF^{2}Net [43]	92.73±0.21
SEMSDNet [44]	93.89±0.63
MSA-Network [45]	93.52±0.21
Attention CNN + H-GCN [46]	93.62±0.28
LCNN-CMGF [47]	94.18±0.35
LSRL [48]	94.27±0.44
HDTFF-Net (ours)	**94.47**±0.26

3) Confusion matrix compares the predicted class labels with the true class labels to calculate the number of correct and incorrect predictions. It provides a summary of the predictions made by the model against the actual or true labels of the data, and helps in estimating the performance of classification models. Generally, the row elements in confusion matrix correspond to the actual labels, while the column elements corresponding to the predicted labels.

C. Scene Classification Results and Analysis

In this article, we assess the performance of HDTFF-Net using the selected dataset and compare it with some existing scene classification methods that operate in the same environment. In experiments, we randomly selected samples from the dataset as the training set and the remaining samples as the test set. We

TABLE IV
COMPARISON OF ACC AND σ ON AID30

Method	*Acc* (%)± σ
SEMSDNet [44]	94.23±0.23
SCCov [42]	96.10±0.16
EFPN-DSE-TDFF [49]	94.50±0.30
DMA [50]	96.12±0.14
MF^2Net [43]	95.93±0.23
MSA-Network [45]	96.01±0.43
Attention CNN + H-GCN [46]	95.78±0.37
ACGLNet [51]	96.10±0.10
LSRL [48]	97.36±0.21
HDTFF-Net(ours)	**97.46**±0.16

The bold values average accuracy to emphasize the superiority of the model's classification performance.

TABLE V
COMPARISON OF ACC AND σ ON PATTERNNET38

Method	*Acc* (%)± σ
RS-COCL-NLF [52]	99.26±0.07
TPENAS [53]	99.05%
RS-DARTS [54]	99.43%
GPAS [55]	98.25%
ACGLNet [51]	99.50±0.11
HDTFF-Net(ours)	**99.64**±0.03

The bold values average accuracy to emphasize the superiority of the model's classification performance.

conduct the experiments five times and take the average accuracy of these experiments as the final accuracy.

1) Classification Results and Analysis on NWPU-RESISC45: The comparison of Acc and σ on the NWPU-RESISC45 dataset by different algorithms are provided in Table III. From Table III, we can see that HDTFF-Net achieves the best average accuracy of 94.47%, when the proportion of training set is set to be 20%, demonstrating the value of HDTFF-Net in fusing hierarchical deep features and in high-resolution RS scene classification.

Fig. 14 shows the confusion matrix of HDTFF-Net on the NWPU-RESISC45 dataset, where the values on the main diagonal are the classification accuracies of all scenes. According to Fig. 14, we can see that the classification accuracies of 42 scenes out of 45 categories are greater than 90%. In the comparison of different classification methods, it is easily to misclassify the "palace" and the "church" due to that they are both classical buildings and with similar styles and the scene structures. The suboptimal algorithm LSRS integrates semantic features into the network model, and the accuracies of "palace" and "church" are respectively 82% and 79%. In contrast, the proposed HDTFF-Net fuses hierarchical deep features by Wave-CNN, Gabor-CNN, and Tra-CNN by D-S evidential theory. Theoretically, it can learn more discriminative features from images, and further enhance the performance of scene classification, And the approving results are shown in Fig. 14, illustrating the accuracies of "palace" and "church" are 88% and 79%, respectively. Especially, the accuracy of "church" has been greatly improved, which can fully verify the effectiveness of HDTFF-Net in scene classification.

2) Classification Results and Analysis on AID30: The AID30 contains 30 categories, and the number of images in every class is not uniform. Thus, the problem of with-in class differences and between-class similarities in it is more serious, making the task of classification more difficult. As shown in Table IV, the HDTFF-Net in this article has achieved an average accuracy of 97.46% when the training set proportion is set to 50%. Compared with the dual-stream deep network models, such as LSRL, ACGLNet, Attention CNN+H-GCN, and Two-Stream Fusion, the classification accuracy in this article has been respectively improved by 0.10%, 0.36%, 1.68%, and 2.88%, which proves that the proposed fusion method can achieve significant improvement on large RS scene datasets.

Fig. 15 shows the confusion matrix generated by HDTFF- Net on the AID30. By observing the values on the diagonal, we can see that the classification accuracies of 28 categories exceed 90%. The classification accuracies of nine categories, such as "baseball field," "beach," "meadow," "forest," etc., reaches 100%. And the performance of some confusing categories, such as "airport," "bare land," "bridge," "dense residential," "stadium," and "sparse residential," is also excellent, reaching 98% or more. In addition, the algorithm also achieves better classification results for the scenes "resort" and "school," which are easily confused by most of the reference algorithms. HDTFF-Net extracts different types of deep features and then fuses them based on an effective fusion criterion, which alleviates the influence of large intraclass variability in the AID dataset and improves the accuracy.

3) Classification Results and Analysis on PatternNet38: The scene categories in the PatternNet38 dataset are uniform, and are with great variability among categories. Therefore, we employ 40% of the dataset for training the network and 60% of the dataset for testing the performance of HDTFF-Net and other advanced classification algorithms. The comparison of Acc and σ on PatternNet38 by different methods is provided in Table V. It is clear that the HDTFF-Net proposed in this article obtains an average accuracy of 99.64%, which is greater than other comparison models. The suboptimal model ACGLNet, which integrates the global and the local features by combining a variety of convolutional neural networks, achieves an average accuracy of 99.50% with 40% training set ratio. The HDTFF-Net algorithm proposed in this paper not only combines the shallow texture features and deep features, but also considers the differences and relationships between the features obtained by different deep feature descriptors. And the accuracy is improved by 0.14%, indicating that the HDTFF-Net can effectively discriminate the PatternNet38 dataset.

Fig. 16 shows the confusion matrix of HDTFF-Net on the PatternNet38 dataset. According to the diagonal data in Fig. 16, we can see that the accuracies of 38 scene categories all reach more than 95%, among which the discrimination of 30 scene categories is completely correct. The most easily confused category in the PatternNet38 dataset, i.e., "sparse residential,"

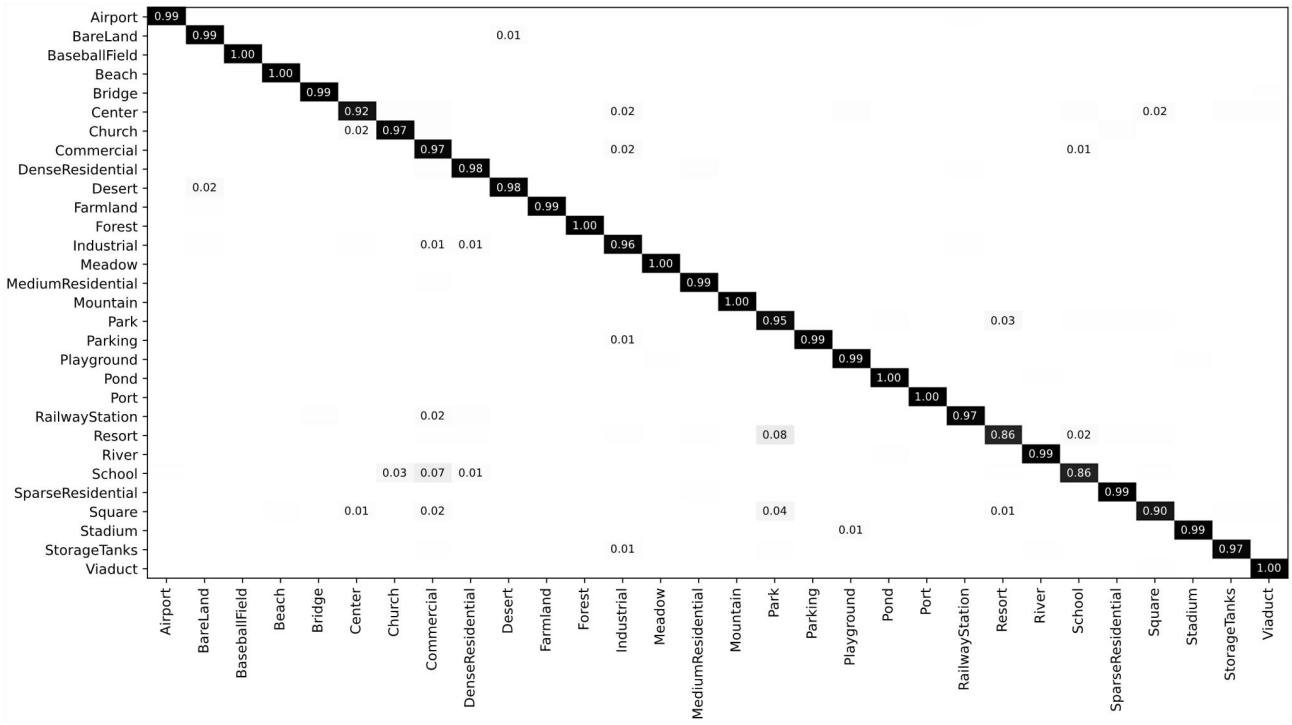

Fig. 15. Confusion Matrix on AID30.

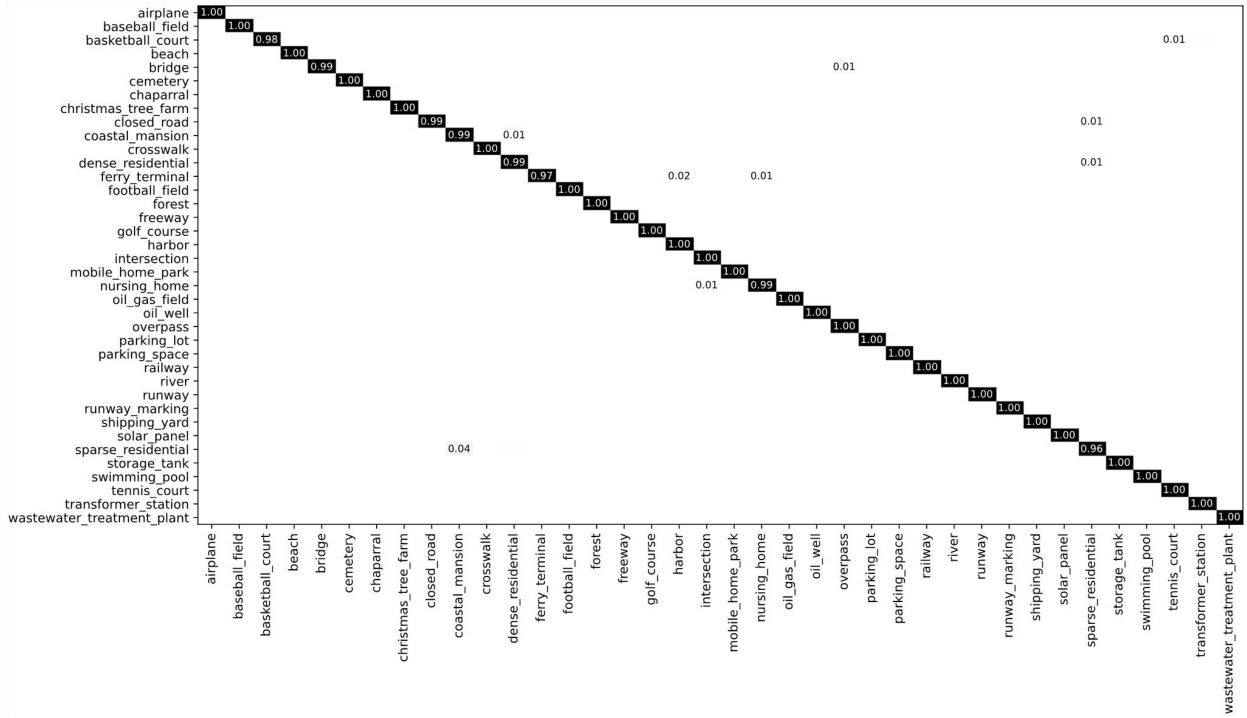

Fig. 16. Confusion Matrix on PatternNet38.

also has an accuracy of 96%. In conclusion, the proposed HDTFF-Net can provide more discriminative features with specific physical meanings and obtain better performance in high-resolution RSSC.

D. Ablation Experiments

According to discussions above, the HDTFF-Net is derived by integrating the three subnetworks, that is Tra-CNN, Gabor-CNN, and Wave-CNN, at an effective decision- level based on

TABLE VI
ABLATION EXPERIMENTS OF DIFFERENT VARIANTS

Method	NWPU RESISC45 (20%)	AID30 (50%)	PatternNet38 (40%)
1. Tra-CNN	93.35	95.48	99.15
2. Gabor-CNN (DenseNet+Gabor)	94.08	96.48	99.45
3. Wave-CNN (DenseNet+Wavelet)	94.12	96.86	99.42
4. Tra-CNN+ Gabor-CNN	94.24	97.02	99.53
5. Tra-CNN+ Wave-CNN	94.37	97.30	99.52
6. Gabor-CNN+ Wave-CNN	94.40	97.38	99.62
7.HDTFF-Net(ours)	**94.47**±0.26	**97.46**±0.16	**99.64**±0.03

The bold values average accuracy to emphasize the superiority of the model's classification performance.

TABLE VII
CLASSIFICATION RESULTS BY DIFFERENT FUSION STRATEGIES

Method	NWPU RESISC45 (20%)	AID30 (50%)	PatternNet38 (40%)
Feature Concatenation	93.85%	96.64%	99.10%
Decision-level Weighting	94.16%	96.92%	99.39%
Decision-level Voting	94.26%	97.14%	99.46%
HDTFF-Net	**94.47**±0.26	**97.46**±0.16	**99.64**±0.03

The bold values average accuracy to emphasize the superiority of the model's classification performance.

the D-S evidential theory. In order to validate the effectiveness of incorporating shallow texture features and multiple deep texture features for enhancing the classification performance, ablation experiments are conducted. Specifically, the performance in RSSC is evaluated by comparing Tra-CNN, Gabor-CNN, Wave-CNN, and the combination of different neural networks using D-S evidential fusion. The results of ablation experiments are presented in Table VI.

As shown in Table VI, the Tra-CNN, that is DenseNet201, achieves 93.35%, 95.48%, and 99.15%, respectively, on NWPU-RESISC45, AID30, and PatternNet38. As discussed above, the Gabor-CNN and the Wave-CNN are enhanced by considering the various textures in images. Fortunately, the Gabor-CNN improves the performance by 0.73%, 1.00%, and 0.30%, and Wave-CNN by 0.77%, 1.38%, and 0.27%. In addition, we also provide the result by fusing any two networks, that is, Tra-CNN+Gabor-CNN, Tra-CNN +Wave-CNN, and Tra-CNN+Wave-CNN, and obtain satisfying performance. All of these can fully demonstrate the advantages of excavating various textures in traditional CNNs for accurate scene classification. Finally, we provide the results by HDTFF-Net, which fuses Tra-CNN, Gabor-CNN, and Wave-CNN by D-S evidential theory. As shown in Table VI, the classification accuracies, respectively, reach 94.47, 97.46, and 99.64, proving that the proposed fusion strategy based on D-S evidential theory can well improve the classification performance of the proposed HDTFF-Net model.

Furthermore, we compare the proposed fusion strategy based on D-S evidence theory with some traditional fusion methods. The reference fusion strategies are specified as follows.

1) Feature vector concatenation fusion.
2) Decision-level weighting fusion (Without loss of generality, we set all weights for Tra-CNN, Gabor-CNN, and Wave-CNN to be 1/3);
3) Decision-level voting fusion. The classification results by different fusion strategies are provided by Table VII. It is clear that directly integrating the three submodels by feature concatenation and decision-level weighting and voting fusions can also obtain satisfying classification accuracy. However, compared with HDTFF-Net, they still have weaker performance. What is more, their motivations and considerations are different from that embodying in the proposed HDTFF-Net.

V. CONCLUSION

In this article, we have designed the HDTFF-Net by fusing the deep features from Tra-CNN, Gabor-CNN, and Wave-CNN for high-resolution RSSC. In HDTFF-Net, DenseNet201 is employed as the backbone network. And then Gabor-CNN and Wave-CNN are designed by fusing the various low-level textures into traditional CNNs, thus providing specific physical meanings for the deep features. Specifically, Gabor-CNN employs the 2-D Gabor filter to extract multiscale and multidirectional texture features, and uses it to replace a portion of the convolutional kernel in the backbone network. And Wave-CNN considers the downsampling characteristics of Wavelet transform and the characteristics of low-frequency and high-frequency components, and also improves its attention to the texture position by a spatial attention mechanism. In addition, the HDTFF-Net model fuses Tra-CNN, Gabor-CNN, and Wave-CNN based on D-S evidence theory by considering their contributions to the final inference of the attributive class. In the end, we have performed abundant experiments on three public datasets, that is NWPU-RESISC45, AID30, and PatternNet38. The comparison results have verified that the HDTFF-Net outperforms other algorithms in excavating effective and discriminative features for high-resolution RSSC. Future works will consider improving our model for RSSC by extracting the nonredundant sparse features to further deal with the overfitting of deep networks and the inefficiency of deep features.

ACKNOWLEDGMENT

The authors would like to thank all the anonymous reviewers for their constructive comments to improve the quality of this paper.

REFERENCES

[1] J. Feng, D. Wang, and Z. Gu, "Bidirectional flow decision tree for reliable remote sensing image scene classification," *Remote Sens.*, vol. 14, no. 16, Aug. 2022, Art. no. 3943.
[2] J. Wang, F. Gao, J. Dong, Q. Du, and H. C. Li, "Change detection from synthetic aperture radar images via dual path denoising network," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 15, pp. 2667–2680, 2022.
[3] J. Wang, F. Gao, J. Dong, S. Zhang, and Q. Du, "Change detection from synthetic aperture radar images via graph-based knowledge supplement network," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 15, pp. 1823–1836, 2022.

[4] J. Fan, Y. Ye, J. Li, G. Liu, and Y. Li, "A novel multiscale adaptive binning phase congruency feature for SAR and optical image registration," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5235216.

[5] C. Shi, T. Wang, and L. Wang, "Branch feature fusion convolution network for remote sensing scene classification," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 13, pp. 5194–5210, 2020.

[6] 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.

[7] Q. Tan, Y. Liu, X. Chen, and G. Yu, "Multi-label classification based on low rank representation for image annotation," *Remote Sens.*, vol. 9, no. 2, Jan. 2017, Art. no. 109.

[8] F. Naccari, S. Battiato, A. Bruna, A. Capra, and A. Castorina, "Natural scenes classification for color enhancement," *IEEE Trans. Consum. Electron.*, vol. 51, no. 1, pp. 234–239, Feb. 2005.

[9] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2005, vol. 1, pp. 886–893.

[10] T. Kobayashi, "Dirichlet-based histogram feature transform for image classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2014, pp. 3278–3285.

[11] B. Luo, S. Jiang, and L. Zhang, "Indexing of remote sensing images with different resolutions by multiple features," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 6, no. 4, pp. 1899–1912, Apr. 2013.

[12] D. Ma, P. Tang, and L. Zhao, "SiftingGAN: Generating and sifting labeled samples to improve the remote sensing image scene classification baseline in vitro," *IEEE Geosci. Remote Sens. Lett.*, vol. 16, no. 7, pp. 1046–1050, Jul. 2019.

[13] 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.

[14] G. Kwak and N. Park, "Unsupervised domain adaptation with adversarial self-training for crop classification using remote sensing images," *Remote Sens.*, vol. 14, no. 18, Sep. 2022, Art. no. 4639.

[15] A. Romero, C. Gatta, and G. Camps-Valls, "Unsupervised deep feature extraction for remote sensing image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 3, pp. 1349–1362, Mar. 2016.

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

[17] J. Xie, N. He, L. Fang, and A. Plaza, "Scale-free convolutional neural network for remote sensing scene classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 9, pp. 6916–6928, Apr. 2019.

[18] X. Wang, L. Duan, A. Shi, and H. Zhou, "Multilevel feature fusion networks with adaptive channel dimensionality reduction for remote sensing scene classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, 2022, Art. no. 8010205.

[19] C. Ma, X. Mu, R. Lin, and S. Wang, "Multilayer feature fusion with weight adjustment based on a convolutional neural network for remote sensing scene classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 18, no. 2, pp. 241–245, Feb. 2021.

[20] C. Shi, X. Zhang, J. Sun, and L. Wang, "Remote sensing scene image classification based on dense fusion of multi-level features," *Remote Sens.*, vol. 13, no. 21, Oct. 2021, Art. no. 4379.

[21] J. Chu and G. Zhao, "Scene classification based on SIFT combined with GIST," *Int. Conf. Inf. Sci., Electron. Elect. Eng.*, vol. 1, pp. 331–336, Apr. 2014.

[22] J. Ji, T. Zhang, L. Jiang, W. Zhong, and H. Xiong, "Combining multilevel features for remote sensing image scene classification with attention model," *IEEE Geosci. Remote Sens. Lett.*, vol. 17, no. 9, pp. 1647–1651, Nov. 2020.

[23] 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, Jun. 2017.

[24] 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.

[25] J. Wang, W. Li, Y. Gao, M. Zhang, R. Tao, and Q. Du, "Hyperspectral and SAR image classification via multiscale interactive fusion network," *IEEE Trans. Neural Netw. Learn. Syst.*, to be published, doi: 10.1109/TNNLS.2022.3171572.

[26] W. Li, J. Wang, Y. Gao, M. Zhang, R. Tao, and B. Zhang, "Graph-feature-enhanced selective assignment network for hyperspectral and multispectral data classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5526914.

[27] R. Hang, P. Yang, F. Zhou, and Q. Liu, "Multiscale progressive segmentation network for high-resolution remote sensing imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5412012.

[28] F. Zhou, C. Xu, R. Hang, R. Zhang, and Q. Liu, "Mining joint intraimage and interimage context for remote sensing change detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, 2023, Art. no. 4403712.

[29] R. Hang, G. Li, M. Xue, C. Dong, and J. Wei, "Identifying oceanic eddy with an edge-enhanced multiscale convolutional network," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 15, pp. 9198–9207, 2022.

[30] J. Wang, F. Gao, J. Dong, and Q. Du, "Adaptive DropBlock-enhanced generative adversarial networks for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 6, pp. 5040–5053, Jun. 2021.

[31] S. Luan, C. Chen, B. Zhang, J. Han, and J. Liu, "Gabor convolutional networks," *IEEE Trans. Image Process.*, vol. 27, no. 9, pp. 4357–4366, May 2018.

[32] Y. Jiang, M. Li, P. Zhang, X. Tan, and W. Song, "Hierarchical fusion convolutional neural networks for SAR image segmentation," *Pattern Recognit. Lett.*, vol. 147, pp. 115–123, Jul. 2021.

[33] R. Cao, L. Fang, T. Lu, and N. He, "Self-attention-based deep feature fusion for remote sensing scene classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 18, no. 1, pp. 43–47, Jan. 2021.

[34] G. Huang, Z. Liu, L. V. D. Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 4700–4708.

[35] Z. Tao, T. Wei, and J. Li, "Wavelet multi-level attention capsule network for texture classification," *IEEE Signal Process. Lett.*, vol. 28, pp. 1215–1219, 2021.

[36] Q. Li, L. Shen, S. Guo, and Z. Lai, "Wavecnet: Wavelet integrated cnns to suppress aliasing effect for noise-robust image classification," *IEEE Trans. Image Process.*, vol. 30, pp. 7074–7089, 2021.

[37] S. E. Grigorescu, N. Petkov, and P. Kruizinga, "Comparison of texture features based on Gabor filters," *IEEE Trans. Image Process.*, vol. 11, no. 10, pp. 1160–1167, Oct. 2002.

[38] 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, Apr. 2017.

[39] G. 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.

[40] W. Zhou, S. Newsam, C. Li, and Z. Shao, "PatternNet: A benchmark dataset for performance evaluation of remote sensing image retrieval," *ISPRS J. Photogrammetry Remote Sens.*, vol. 145, pp. 197–209, Nov. 2018.

[41] Y. Yu and F. Liu, "A two-stream deep fusion framework for high-resolution aerial scene classification," *Comput. Intell. Neurosci.*, vol. 2018, Jan. 2018, Art. no. 8639367.

[42] N. He, L. Fang, S. Li, J. Plaza, and A. Plaza, "Skip-connected covariance network for remote sensing scene classification," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 5, pp. 1461–1474, May 2019.

[43] 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.

[44] T. Tian, L. Li, W. Chen, and H. Zhou, "SEMSDNet: A multiscale dense network with attention for remote sensing scene classification," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 5501–5514, 2021.

[45] G. Zhang et al., "A multiscale attention network for remote sensing scene images classification," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 9530–9545, 2021.

[46] Y. Gao, J. Shi, J. Li, and R. Wang, "Remote sensing scene classification based on high-order graph convolutional network," *Eur. J. Remote Sens.*, vol. 54, pp. 141–155, Feb. 2021.

[47] C. Shi, X. Zhang, and L. Wang, "A lightweight convolutional neural network based on channel multi-group fusion for remote sensing scene classification," *Remote Sens.*, vol. 14, no. 1, Jan. 2022, Art. no. 9.

[48] S. Yang, F. Song, G. Jeon, and R. Sun, "Scene changes understanding framework based on graph convolutional networks and swin transformer blocks for monitoring LCLU using high-resolution remote sensing images," *Remote Sens.*, vol. 14, no. 15, Aug. 2022, Art. no. 3709.

[49] 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.
[50] J. Shen et al., "A dual-model architecture with grouping-attention-fusion for remote sensing scene classification," *Remote Sens.*, vol. 13, no. 3, Jan. 2021, Art. no. 433.
[51] J. Shen, T. Yu, H. Yang, R. Wang, and Q. Wang, "An attention cascade global-local network for remote sensing scene classification," *Remote Sens.*, vol. 14, no. 9, Apr. 2022, Art. no. 2042.
[52] Q. Li, Y. Chen, and P. Ghamisi, "Complementary learning-based scene classification of remote sensing images with noisy labels," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, Sep. 2021, Art. no. 8021105.
[53] L. Ao et al., "TPENAS: A two-phase evolutionary neural architecture search for remote sensing image classification," *Remote Sens.*, vol. 15, no. 8, Apr. 2023, Art. no. 2212.
[54] Z. Zhang, S. Liu, Y. Zhang, and W. Chen, "RS-DARTS: A convolutional neural architecture search for remote sensing image scene classification," *Remote Sens.*, vol. 14, no. 1, Dec. 2021, Art. no. 141.
[55] C. Peng, Y. Li, L. Jiao, and R. Shang, "Efficient convolutional neural architecture search for remote sensing image scene classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 7, pp. 6092–6105, Jul. 2021.

Wanying Song (Member, IEEE) was born in 1988. She received the B.S. degree in electrical science and technology from Shandong University of Science and Technology, Qingdao, China, in 2012, and the M.S. and Ph.D. degrees in signal and information processing from Xidian University, Xi'an, China, in 2015 and 2018, respectively.

She currently works with the School of Communication and Information Engineering, Xi'an University of Science and Technology, Xi'an. She has authored more than 30 technical papers. Her main research interests include remote sensing image analysis and interpretation, and statistical learning theory.

Yifan Cong was born in 1998. He received the B.S. degree in communication engineering from JiangSu Normal University KeWen College, Jiangsu, China, in 2020. He is currently working toward the M.S. degree with the Xi'an Key Laboratory of Network Convergence Communication, Xi'an University of Science and Technology, Xi'an, China.

His main research interests include remote sensing image analysis and interpretation.

Shiru Zhang (used name Minrui Zhang) received the B.S., the M.S. degrees in communications and electronics systems, and the Ph.D. degree in signal and information processing, all from Xidian University, Xi'an, China, in 1987, 1993, and 2005, respectively.

She was a Visiting Scholar with the University of Illinois at Urbana-Champaign in USA from 2008 to 2009, and a Visiting Professor with Chin-Yi University of Technology, Taiwan, during 2014 to 2015. She is currently with the Xi'an University of Science and Technology in China, Xi'an, China, where she is currently a Professor. Her research interests include image processing, signal processing for traditional Chinese medicine, and information hiding.

Yan Wu (Member, IEEE) received the B.S. degree in information processing, and the M.S. and Ph.D. degrees in signal and information processing from Xidian University, Xi'an, China, in 1987, 1998, and 2003, respectively.

From 2003 to 2005, she was a Postdoctoral Fellow with the National Laboratory of Radar Signal Processing. Since 2005, she has been a Professor with the School of Electronic Engineering, Xidian University. She has authored more than 60 technical papers. Her broad research interests include remote sensing image analysis and interpretation, data fusion of multi-sensor images, SAR auto target recognition, and statistical learning theory and application.

Peng Zhang (Member, IEEE) received the B.S. degree in electronic and information engineering, the M.S. and Ph.D. degrees in signal and information processing from Xidian University, Xi'an, China, in 2006, 2009 and 2012, respectively. He currently works with National Laboratory of Radar Signal Processing, Xidian University. His main research interests include SAR image analysis and interpretation and statistical learning theory.