Scene Classification with Recurrent Attention of VHR Remote Sensing Images

Qi Wang, *Senior Member, IEEE*, Shaoteng Liu, Jocelyn Chanussot, *Fellow, IEEE*, and Xuelong Li, *Fellow, IEEE*

***Abstract*—Scene classification of remote sensing images has drawn great attention because of its wide applications. In this paper, with the guidance of human visual system (HVS), we explore the attention mechanism and propose a novel end-to-end *Attention Recurrent Convolutional Network (ARCNet)* for scene classification. It can learn to focus selectively on some key regions or locations and just process them at high-level features, thereby discarding the non-critical information and promoting the classification performance. The contributions of this paper are three-fold: First, we design a novel *recurrent attention structure* to squeeze high-level semantic and spatial features into several simplex vectors for the reduction of learning parameters. Second, an end-to-end network named *ARCNet* is proposed to adaptively select a series of attention regions and then to generate powerful predictions by learning to process them sequentially. Third, we construct a new dataset named OPTIMAL-31, which contains more categories than popular datasets and gives researchers an extra platform to validate their algorithms. The experimental results demonstrate that our model makes great promotion in comparison with state-of-the-art approaches.**

***Index Terms*—deep learning, CNN, RNN, LSTM, attention, scene classification, remote sensing**

I. Introduction

WITH the rapid development of remote sensing instruments over recent years [1], [2], very high resolution (VHR) remote sensing images are becoming increasingly available and bringing us the opportunity to try more researches in military and civilian applications, such as natural disaster detection [3], [4], land-cover/land-use classification [5], [6] geographic space object detection [7], [8], geographic image retrieval [9], [10], urban planning, and environment monitoring. As we all know, VHR remote sensing images recognition based on the knowledge of domain experts has high labor cost. Therefore, intelligent scene classification of remote sensing images [11], [12], [13], [14], which categorizes scene images into different classes based on its semantic information, has drawn great attention in remote sensing field. Nevertheless, because of various classes of scenes and complex spatial information of VHR remote sensing images, how to effectively describe and classify the scenes is a pivotal and challenging task.

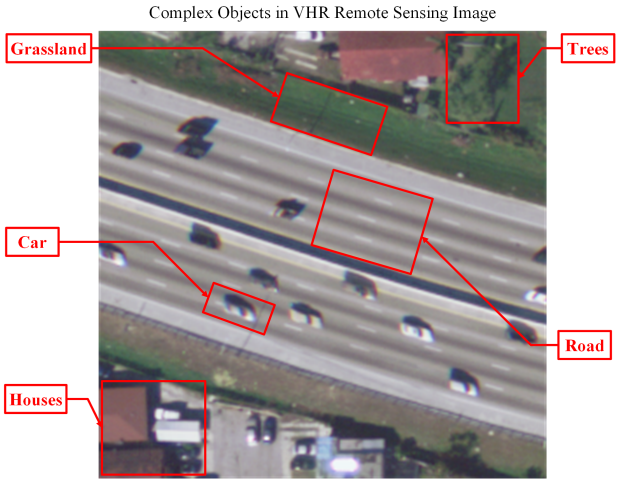


Fig. 1. VHR remote sensing image contains many types of objects and features. For the category of road, only pavement and vehicles are useful for classification, and the rest are redundant parts.

VHR remote sensing images are quite different with normal images due to its unique capture mode. They usually cover a large area with an overhead view, which cause the images contain many types of objects and features. Obviously, not all these spatial information are useful as shown in Fig. 1. So how to focus on the critical parts of images and abandon the useless ones are very crucial. However, most previous works tend to generate a global representation of image with the same contribution of each part [13], [15], [16], [17], [18], in spite of the negative effects of redundant areas. For this reason, we intend to design an attention mechanism to solve this problem with the guidance of human visual system (HVS).

It is universally acknowledged that when our human need to classify scenes, one important property of HVS is that it will not process the entire image at once. Instead we take the initiative to select some key regions and combine them to generate an internal representation of the scene, as shown in Fig. 2. This is more than intelligent because concentrating on parts of the image could save computational resources and enhance classification result as some useless data will not be processed. It is called the visual attention mechanism [19], [20]. In the applications of neural networks, the fundamental function of attention mechanism is to allow the network to

Q. Wang is with the School of Computer Science, and with the Unmanned System Research Institute, and also with the Center for OPTical IMagery Analysis and Learning (OPTIMAL), Northwestern Polytechnical University, Xi'an 710072, China (e-mail: crabwq@gmail.com).

S. Liu is with the the School of Computer Science and with the Center for OPTical IMagery Analysis and Learning (OPTIMAL), Northwestern Polytechnical University, Xi'an 710072, China (e-mail: liushaoteng@live.com).

J. Chanussot is with the Grenoble Images Speech Signals and Automatics Laboratory, Grenoble Institute of Technology, 38400 Grenoble, France, and also with the Faculty of Electrical and Computer Engineering, University of Iceland, 107 Reykjavk, Iceland (e-mail: jocelyn.chanussot@gipsa-lab.grenoble-inp.fr).

X. Li is with the Xi'an Institute of Optics and Precision Mechanics, Chinese Academy of Sciences, Xi'an 710119, China (e-mail: xuelong_li@opt.ac.cn), and the University of Chinese Academy of Sciences, Beijing 100049, China.

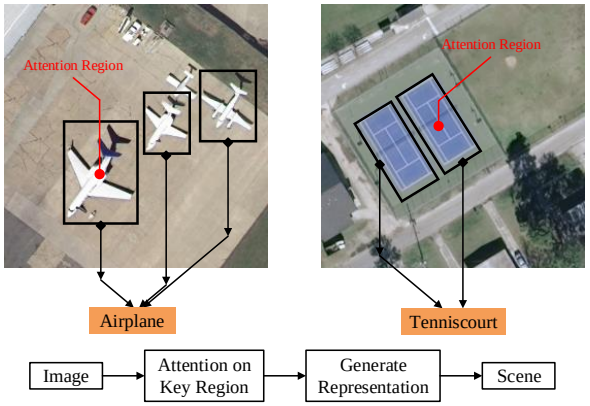


Fig. 2. With the guidance of human visual system, the basic function of attention mechanism and the simple flow chart of *ARCNet*.

select the input data rather than put all of them into a fixed length encoding vector, and most attention mechanisms can be divided into two parts: hard attention and soft attention. For soft attention, the attention weights of each region in the image are series of linear and continuous values. But for hard attention, these values are limited to the determined 0 or 1.

On one hand, attention mechanism can save computing resources and remove redundant data. On the other hand, remote sensing images have complex spatial information so that single key region can not well express the whole image. Under these circumstances, we design a novel *recurrent attention structure* to generate several attention features based on the high-level input, and then squeeze them into some simplex vectors. For better exploiting the advantages of attention mechanism, we propose the *Attention Recurrent Convolutional Network (ARCNet)* which contains *high-level feature extractor*, *recurrent attention structure*, and *sequential representation processor*. Specifically, we employ pre-trained Convolutional Neural Networks (CNN) model [21], [22], [23], [24], [25], [26] as feature extractor to convert remote sensing images to high-level representations, and explore deep Long Short Term Memory (LSTM) networks to process attention features sequentially. Both the number of parameters in our model and the amount of computation can be controlled independently. It is worth mentioning that we design an end-to-end optimization procedure that allows the model to be trained directly with back propagation.

Overall, this paper makes the following contributions.

1) A novel *recurrent attention structure* is proposed. With this structure, classifiers can focus on key areas through learning just as human visual system do. Especially for the remote sensing images which contain complex objects, it can greatly accelerate the convergence rate and improve the accuracy.
2) An end-to-end network named *ARCNet* is proposed. With this network, attention mechanism can adapt to remote sensing images and its training difficulty is greatly reduced. To the best of our knowledge, the proposed *ARCNet* is one of the first successful attempts on attention mechanism for remote sensing scene classification.
3) A new remote sensing scene classification dataset named OPTIMAL-31 is constructed. It contains more categories than popular datasets and gives researchers an extra platform to validate their algorithms.

The structure of this article is as follows. Section I gives a brief introduction of the background and motivation of this paper. Section II introduces the related works. Section III introduces the details of the proposed *ARCNet*. In Section IV, the experimental results and analysis are reported. Finally, the conclusions are made in Section V.

II. Related Works

In this section, the relevant works of remote sensing scene classification and attention mechanism based methods are briefly reviewed.

A. Scene Classification

The previous works about remote sensing scene classification are in varied forms, but can be roughly divided into the following three categories according to the features they used: handcrafted features, unsupervised learning features, and deep learning features [14].

Handcrafted Features: The methods based on handcrafted features are the earliest in scene classification of remote sensing image. Color histograms and texture descriptors [8], [27] are global features which can be sent to the classifier directly. Scale invariant feature transform (SIFT) [10], [28] and histogram of oriented gradients [29], [30] are local features which usually need mid-level descriptor to generate the entire representation [10], [31]. Recently, a combination of multiple different features is considered as a promising approach to seek further promotion [28], [32], [31]. For example, Zhu et al. [31] propose a local-global feature for bag-of-visual-words scene classifier, which can combine several features by a fusion operation at histogram level. Nevertheless, how to design an effective model to combine these features is very difficult, and the representation ability of handcrafted features becomes weaker with the increasing challenge of this task.

Unsupervised Learning Features: In the domain of remote sensing, many unsupervised methods have been used and have achieved better performance than those based on handcrafted features [33], [34], [13], [35], [36]. For example, Chaib et al. [37] present an informative feature selection method, which applies a sparse PCA to learn informative feature from a dictionary constructed with SIFT. However, because of the less usage of image labels, unsupervised learning cannot guarantee to distinguish the differences between different scenes.

Deep Learning Features: As the active performance of deep learning, almost all state-of-the-art approaches about remote sensing scene classification are based on CNN [15], [16], [38], [17], [39], [25], [26], [40], [41], [42]. In the beginning, many researchers are prone to fully training a CNN using just a few thousand pictures of remote sensing images datasets. But due to the strict requirement of data size and data quality, it is demonstrated to be quite difficult. Under these circumstances, pre-trained CNN have been transferred to this task [7], [39],

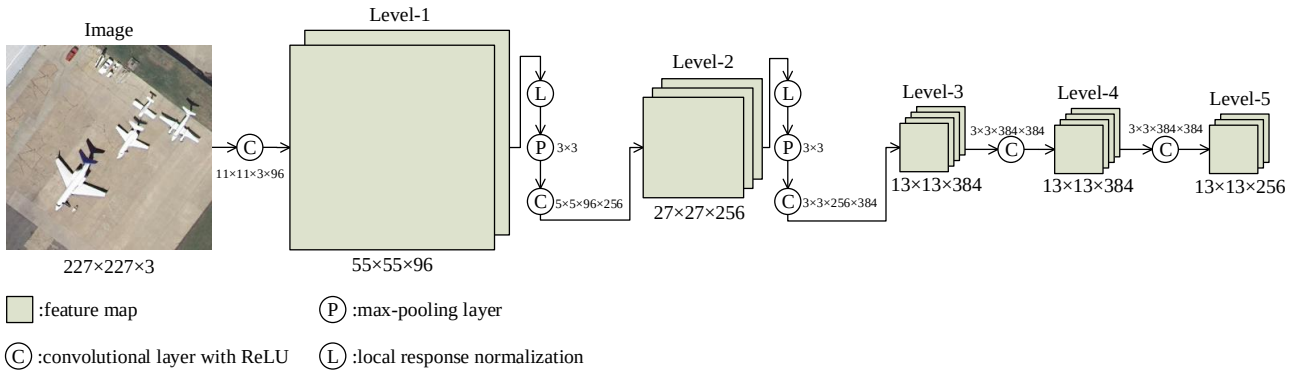


Fig. 3. The overall architecture of the high-level feature extractor. It is composed of five convolutional layers, and we choose the Level-5 feature map as the representation of the image.

[26] which means training the CNN with other lager datasets, e.g., ImageNet, Places205 [43], etc. Now almost every network is based on pre-trained CNN. So employing it to solve the data problem is a fast and efficient way.

B. Attention Mechanism

As far as we know, there is no previous work of using attention mechanism in the domain of remote sensing. The most relevant one is to combine saliency detection into scene classification [44], [45], [46]. Saliency detection means to detect the salient parts of an image, and it is based on the assumption that a region of interest is generally salient. For example, Zhang et al. [44] propose a saliency-guided sampling strategy to extract the representation from a remote sensing image, which has the ability to remove redundant information. Zhang et al. [45] take the extracted saliency map into account primary objects, which is a special feature encoder to assist the classifier.

Superficially, saliency detection has the same idea with attention mechanism. But in fact not all salient parts in an image are important. For saliency detection, it mainly uses the texture information of the image to calculate the salient parts, which does not have the ability to distinguish the degree of importance. For attention mechanism, it will constantly adjust the supervision signal through training and finally learn the region of interest. The former focuses on calculation, while the latter on learning, which is the biggest difference. Therefore, learning-centered attention mechanism is superior to the saliency detection in the case of having sufficient data.

III. METHOD DESCRIPTION

In this section, we will explain the specific details of the proposed *ARCNet* for VHR remote sensing images scene classification, and it has three main components: *high-level feature extractor*, *recurrent attention structure*, and *sequential representation processor*. Our method's core idea is to add a branch into the classifier which can adjust a group of weights by supervise learning, then the new representation is encoded by input image and these weights. Instead of treating each part of image as same, it gives a way to rearrange the input signal through learning. Next we will introduce these parts in succession, and give the overall architecture of *ARCNet* in the end.

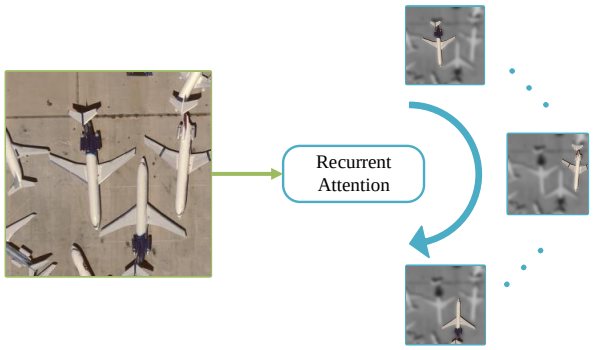


Fig. 4. The effect of the recurrent attention structure, which is discarding useless information and focusing on key regions.

A. High-level Feature Extractor

If the attention operation is performed on the original images, the number of learning weights tends to be quite large as the high resolution, which will increase the amount of computation and the difficulty of training. So we decide to process high-level features which have fewer pixels on a single channel.

Currently, the most effective way to extract high-level features of images is deep convolutional operation. Therefore, we construct our extractor based on it to generate representations and introduce pre-trained operation to solve the problem of less training dataset. Our *ARCNet* is applicable to a variety of CNN, such as AlexNet [21], VGGNet [22], ResNet [23] and so on. In this section, we take AlexNet as an example as shown in Fig. 3.

AlexNet [21] is an innovative deep CNN that combines five convolutional layers and three fully connected layers. Its main advantages include dropout as well as rectified linear unit

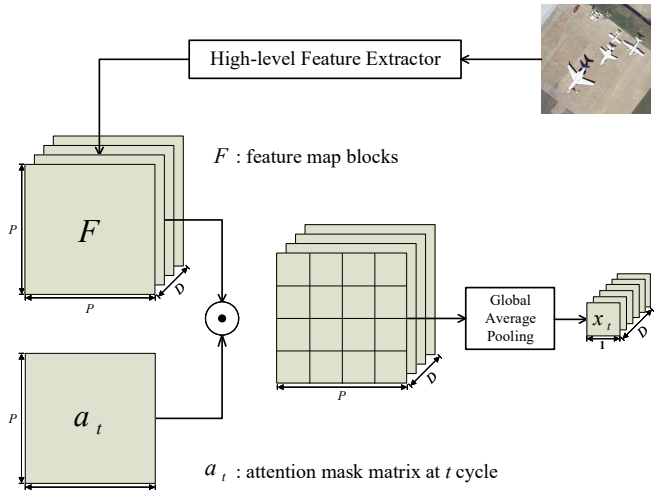


Fig. 5. The sketch map of how the mask matrix works. First, feature block is generated by the pre-trained CNN. Next, the input data of deep LSTM networks is computed by this feature block and mask matrix.

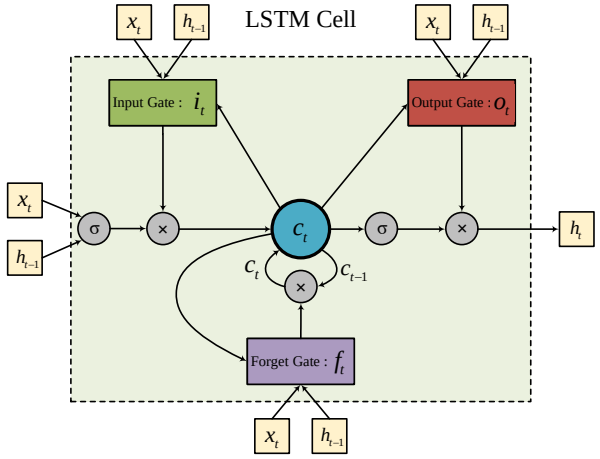


Fig. 6. Each LSTM unit remembers a single floating-point value c_t. This value may be diminished or erased through a multiplicative interaction with the forget gate f_t or additively modified by the current input x_t multiplied by the activation of the input gate i_t. The output gate o_t controls the emission of h_t, and the stored memory c_t transformed by the hyperbolic tangent non-linearity.

(ReLU) activating functions for problems with non-linearity. The ReLU has a half-wave rectifier function that contains only positive numbers from the training phase. The dropout method reduces co-adaptations of neurons and substantially abridges over-fitting in fully connected layers.

In our model, we cut the normal AlexNet and take Level-5 convolutional features as the input of attention operation, as shown in Fig. 3. The resolution of these features is 13×13, which is far less than 227×227 of the original images. The extractor produces P^2 vectors, each of which is a D dimensional representation corresponding to a part of the image,

$$\boldsymbol{F} = \{\boldsymbol{f}_1, \boldsymbol{f}_2, \cdots, \boldsymbol{f}_{P\times P}\}, \qquad \boldsymbol{f}_i \in \mathbb{R}^D, \tag{1}$$

where F is the feature block and is converted from scene images by pre-trained AlexNet. P is the width of F, and $\boldsymbol{f}_i$ is the i_{th} feature vector in F. In addition, we select the feature block from the last convolutional layer in contrast to the previous work which chose the vector from the last fully connection layer. Therefore, the size of feature block is $13 \times 13 \times 256$ which means $P = 13$ and $D = 256$ in our model.

B. Recurrent Attention Structure

Recurrent attention structure is the main idea of this paper, and even the entire *ARCNet* is constructed around it. As described above, most previous works tend to generate a global representation of images in spite of the negative effect of redundant areas. So this structure aims at generating several attention representations based on high-level features of remote sensing images, as shown in Fig. 4. The specific implementation is based on mask matrix which has the same size of high-level features. At the same time, this structure will help discard the non-critical information, thereby improving the classification performance and reducing the computational complexity.

As mentioned in the previous section, the feature block extracted from pre-trained CNN is represented as $\boldsymbol{F} = \{\boldsymbol{f}_1, \boldsymbol{f}_2, \cdots, \boldsymbol{f}_{P\times P}\}$. In our model, we employ a mask matrix a_t as the attention weights to make a multiplication with F directly, and the mask matrix will be updated according to the output in each time-step. This operation is shown in Fig. 5, and the calculation function is as follows:

$$a_t = \{a_{t,1}, a_{t,2}, \cdots, a_{t,P\times P}\}, t \in 1 \cdots T, \tag{2}$$

$$\boldsymbol{x}_t = \sum_{i=1}^{P\times P} a_{t,i}\boldsymbol{f}_i, \qquad \boldsymbol{x}_t \in \mathbb{R}^D, \boldsymbol{f}_i \in \mathbb{R}^D, \tag{3}$$

where a_t is the attention mask matrix at t time, and the size of each a_t is $P \times P$. T is the total number of recurrences. $\boldsymbol{f}_i$ is one element in feature block $\boldsymbol{F}$, which has D dimensions as same as $\boldsymbol{F}$. $\boldsymbol{x}_t$ is a $P \times P \times D$ block which is feature block processed by recurrent attention structure at t time.

C. Sequential Representation Processor

For one aspect, *recurrent attention structure* will generate a series of attention representations waiting for processes. For another aspect, the mask matrix mentioned above requires continuous learning and updating. So we need a sequential processor to solve these problems. About this type of processor, RNN is the most powerful one. In our model, we build a deep RNN to find a sequential way to process the recurrent attention features and timely update the mask matrix by learning.

When given an input $\boldsymbol{x} = (x_1, x_2, \cdots, x_T)$ and the hidden layer state $\boldsymbol{h} = (h_1, h_2, \cdots, h_T)$ which will transmit to the next time in deep RNN, the characteristic of deep RNN can be expressed by the following state update function:

$$RNN : h_{t-1}, x_t \rightarrow h_t, \tag{4}$$

where h_t is the hidden state at t time and x_t is the input data at t time. Besides, the hidden layer activation function is the logistic sigmoid function.

Different from normal deep RNN, LSTM networks is a special kind of RNN which can easily memory information for a large number of time-steps. Specifically, LSTM unit uses a vector named memory cell to store long term memory, which allows it to find long range relationships better and achieve great performance than normal deep RNN. When it comes to this, the state update of LSTM unit can be described as follow:

$$LSTM : h_{t-1}, c_{t-1}, x_t \rightarrow h_t, c_t, \tag{5}$$

where c_t is the memory cell state at t time.

In previous works, many LSTM units are proposed. Despite the small differences in connections and activation functions, all LSTM units have the memory cell to store the long range information over times. Moreover, in every time-step, LSTM unit has the function to decide what kind of memory needs to be forgotten and what kind of memory needs to be passed.

In our method, we chose LSTM networks as the deep RNN to give assistance for the implementation of our recurrent attention structure. The hidden layer is computed as follows:

$$i_t = \sigma(W_{xi}x_t + W_{hi}h_{t-1} + W_{ci}c_{t-1} + b_i), \tag{6}$$
$$f_t = \sigma(W_{xf}x_t + W_{hf}h_{t-1} + W_{cf}c_{t-1} + b_f), \tag{7}$$
$$c_t = f_t c_{t-1} + i_t \tanh(W_{xc}x_t + W_{hc}h_{t-1} + b_c), \tag{8}$$
$$o_t = \sigma(W_{xo}x_t + W_{ho}h_{t-1} + W_{co}c_{t-1} + b_o), \tag{9}$$
$$h_t = o_t \tanh(c_t), \tag{10}$$

where i is the *input gate*, f is the *forget gate*, o is the *output gate*, and c is the *memory cell* activation vector. Besides, σ is the logistic sigmoid function. In general, the information which is stored in the *memory cell* can only be added by the *input gate* and be deleted by the *forget gate*. The *output gate* is responsible for the output of *memory cell*. The architectures of LSTM unit is shown in Fig. 6.

D. ARCNet

LSTM unit is applied as the sequential representation processor and the implementation is introduced above. So the next step is to construct the entire network. The overall architecture of the proposed *ARCNet* is shown in Fig. 7. In this network, the output of the previous LSTM layer will influence the input of next layer, and it gives the possibility to readjust the signal of supervision. After several experiments, we decide to use three stacked LSTM layers, each hidden layer with D memory cells, which is the dimension of the input data. At the end of stacked LSTM layers, a softmax layer is set to predict the category of the scene. The output of softmax layer as well as the output of the whole network is described as follows:

$$\boldsymbol{y} = \{\boldsymbol{y}_1, \boldsymbol{y}_2, \cdots, \boldsymbol{y}_T\}, \tag{11}$$
$$\boldsymbol{y}_t = (y_{t,1}, y_{t,2}, \cdots, y_{t,L}), t \in 1 \cdots T, \tag{12}$$

where T is the number of time-step while also is the recurrent time in our deep LSTM networks. $\boldsymbol{y}_t$ is the prediction vector at t time and $\boldsymbol{y}$ is the combination of prediction vector over time. At last, L is the number of labels and $y_{t,l}$ is the probability of the l_{th} label.

In addition, the softmax function about prediction probability is

$$y_{t,i} = \frac{\exp(\boldsymbol{W}_{t,i}\boldsymbol{h}_{t-1})}{\sum_{j=1}^{L} \exp(\boldsymbol{W}_{t,j}\boldsymbol{h}_{t-1})}, i \in 1 \cdots L, \tag{13}$$

where $\boldsymbol{W}_{t,i}$ is the weight block to the i_{th} label at t time.

The architecture to be discussed next is the detail of updating mode for mask matrix a_t. In general, our model will predict the next attention mask matrix through the output of deep LSTM networks. Taking the t time as an example, we send the output of deep LSTM networks $\boldsymbol{h}_t$ to a softmax directly over $P \times P$ which is the size of mask matrix.

$$h_t \xrightarrow{softmax} a_{t+1}. \tag{14}$$

This softmax outputs the probability over $P \times P$ area, and this probability can be regarded as the important degree of each pixel in the feature block. Therefore, the area with high important degree is the key region. The softmax function about mask matrix is

$$a_{t,i} = \frac{\exp(\boldsymbol{W}_{t,i}\boldsymbol{h}_{t-1})}{\sum_{j=1}^{L} \exp(\boldsymbol{W}_{t,j}\boldsymbol{h}_{t-1})}, i \in 1 \cdots P \times P, \tag{15}$$

where $\boldsymbol{W}_{t,i}$ in this place is the weight block of the i_{th} pixel at t time. After generating the probability, the next time-step input of deep LSTM networks $\boldsymbol{x}_t$ is calculated by *recurrent attention structure*.

Now back to the label prediction function, $\boldsymbol{y}$ is the combination of each prediction vector over time, but the final one is not calculated. In our model, we try several operations to get the final prediction, just as shown in Fig. 8. For better description, the final prediction vector is

$$\boldsymbol{V} = (v_i, v_2, \cdots, v_L), \tag{16}$$

where $\boldsymbol{V}$ is the final prediction vector and L is the total number of labels.

The three combination approaches we propose are as follows.

1) Take the last time prediction vector as the final one:

$$\boldsymbol{V} = \boldsymbol{y}_T. \tag{17}$$

2) Sum the prediction vector over time:

$$\boldsymbol{V} = \sum_{i=1}^{t} \boldsymbol{y}_i. \tag{18}$$

3) Sum the prediction vector with linear weight function:

$$g_t = \frac{t}{T}, t \in 1 \cdots T, \tag{19}$$
$$\boldsymbol{V} = \sum_{i=1}^{t} g_i \boldsymbol{y}_i. \tag{20}$$

In the end, the prediction is the one which has the highest probability in the final prediction vector $\boldsymbol{V}$.

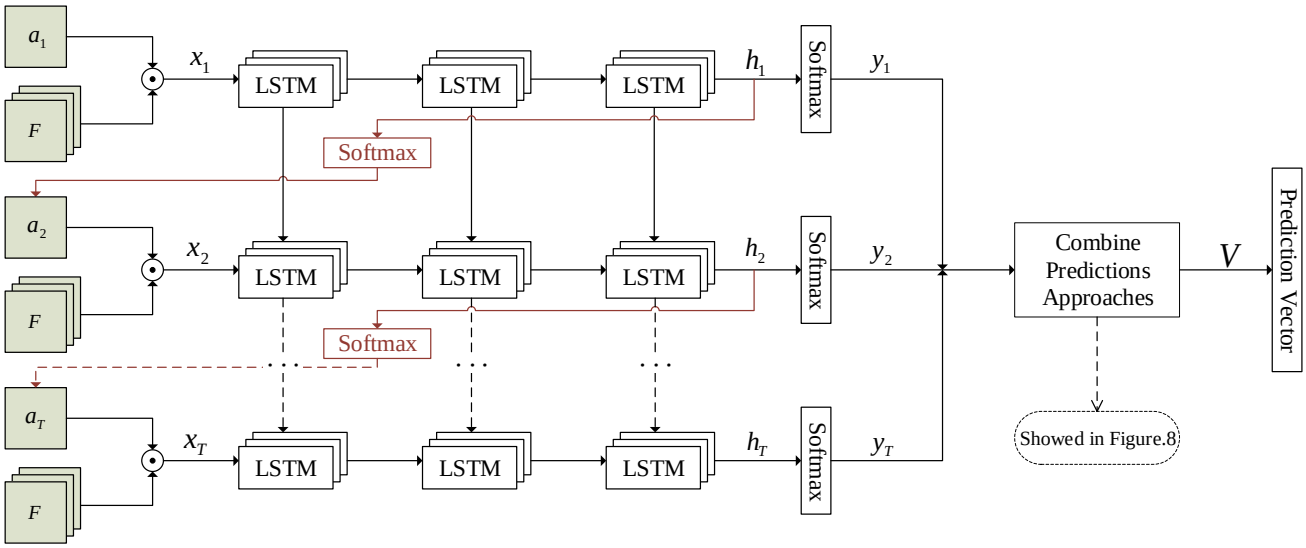


Fig. 7. The overall architecture of the *ARCNet*.

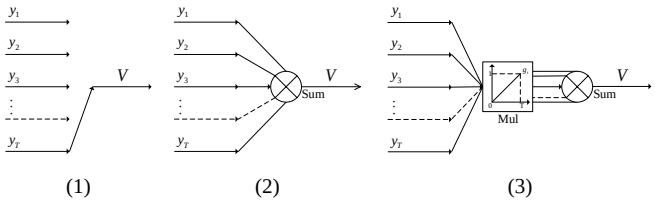


Fig. 8. Three approaches to combine all prediction vectors to the final prediction vectors. (1) Take the last time prediction vector as the final one. (2) Sum the prediction vector over time. (3) Sum the prediction vector with linear weight function.

IV. EXPERIMENT

In this section, extensive experiments are conducted to confirm the effectiveness of the proposed *ARCNet*. First, the datasets and evaluation metrics we used to validate the method are introduced. Second, the details of parameter setting are explained. Finally, the superiority of the proposed method is discussed in comparison with some state-of-the-art algorithms.

A. Experimental Dataset

1) UC Merced Land-Use Dataset: The UC Merced Land-Use (UCM) dataset [47] is the first ground truth dataset derived from a publicly available high resolution overhead image. It was manually extracted from aerial orthography and downloaded from the United States Geological Survey (USGS) National Map. This dataset contains 21 typical land-use scene categories, including agricultural, airplane, baseball diamond, beach, buildings, chaparral, dense residential, forest, freeway, golf course, harbor, intersection, medium density residential, mobile home park, overpass, parking lot, river, runway, sparse residential, storage tanks, and tennis courts. Each class consists of 100 images measuring 256×256 pixels with a pixel resolution of 30 cm in the RGB color space. The classification of UCM dataset is challenging because of the high inter-class similarity among categories such as medium residential and dense residential areas.

2) WHU-RS19 Dataset: The WHU-RS19 (RS19) dataset [34] is a new publicly available dataset wherein all the images are collected from Google Earth (Google Inc. Mountain View, CA, USA). This dataset consists of 950 images with a size of 600×600 pixels distributed among 19 scene classes, including airport, beach, bridge, commercial area, desert, farmland, football field, forest, industrial area, meadow, mountain, park, parking lot, pond, port, railway station, residential area, river, and viaduct. It can be seen that, compared to the UCM dataset, the scene categories in the WHU-RS dataset are more complicated due to variations in scale, resolution, and viewpoint-dependent appearance.

3) Aerial Image Dataset: The Aerial Image Dataset (AID) [17] is a large-scale dataset for aerial scene classification which has a number of 10000 images with a size of 600×600 pixels. This dataset is made up of the following 30 aerial scene types: airport, bare land, baseball field, beach, bridge, center, church, commercial, dense residential, desert, farmland, forest, industrial, meadow, medium residential, mountain, park, parking, playground, pond, port, railway station, resort, river, school, sparse residential, square, stadium, storage tanks and viaduct. All the images are labelled by the specialists in the field of remote sensing image interpretation.

4) OPTIMAL-31 Dataset: As the lack of experimental data in the domain of remote sensing, we construct a new dataset for scene classification named as OPTIMAL-31, where the images contained are also collected from Google Earth. In this dataset, 31 classes are constructed and each class is formed by 60 images with the size of 256×256 pixels, so it has a total of 1860 images. In addition, the classes of our dataset in-

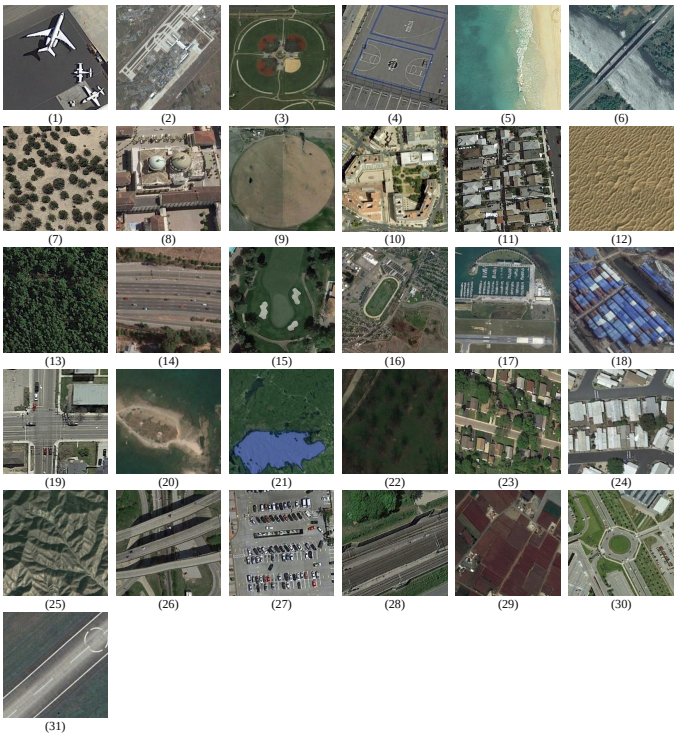


Fig. 9. Example of OPTIMAL-31 dataset. (1)-(31) means airplane, airport, baseball filed, basketball court, beach, bridge, bushes, church, round farmland, business district, dense houses, desert, forest, freeway, golf field, playground, harbor, factory, crossroads, island, lake, meadow, medium houses, mobile house area, mountain, overpass, parking lot, railway, square farmland, roundabout, and runway.

cludes airplane, airport, baseball filed, basketball court, beach, bridge, bushes, church, round farmland, business district, dense houses, desert, forest, freeway, golf field, playground, harbor, factory, crossroads, island, lake, meadow, medium houses, mobile house area, mountain, overpass, parking lot, railway, square farmland, roundabout, and runway. The examples of every class is shown in Fig. 9. OPTIMAL-31 contains more categories than popular datasets so that it has a higher degree of difficulty. The dataset has been uploaded to the OneDrive and all researches can download it from this address: https://1drv.ms/u/s!Ags4cxbCq3lUguxW3bq0D0wbm1zCDQ

B. Evaluation Metrics

In the task of image classification, the most widely used evaluation metrics are overall accuracy, average accuracy and confusion matrix [14].

Overall Accuracy (OA): No matter which class the images belong to, take the number of all correctly classified images divided by the the number of whole dataset.

Average Accuracy (AA): It means averaging the prediction accuracy of every class.

Confusion Matrix (CM): It is a particular matrix to show the performance of algorithm in a visual way, which is widely used in supervised learning. In this matrix, each row represents the actual categories, and each column represents the predicted value. Therefore, it can be very easy to show that whether these multiple categories have confused or not.

In this work, all the experimental datasets have the same number of images in each class, so the OA is equivalent to the AA. Thus, we only use OA and CM in this paper to make the evaluation.

In actual training, the results are usually good for the training set, but the fitting precision for other data is typically not so satisfactory. So we usually don't take all the dataset for training, but take a part of it as validation set to validate the models. This is called cross validation and we will use 5-fold cross validation in most cases, which means 80% of the dataset for training and the rest for validation. In the meanwhile, the influence of different ratio of training set and validation set will also be discussed in this paper.

C. Training Details

In this section, the basic training parameter setting will be explained. Moreover, some important parameters and different network structures will be discussed emphatically including the number of recurrences, the number of LSTM layers, different combination approaches and high-level feature extractors. For these parameters, we do several experiments to choose the most suitable one on the UCM dataset.

1) Training Parameter: For the pre-trained CNN, we use the pre-trained weights which can be downloaded at http://places.csail.mit.edu/downloadCNN.html. In particular, for all datasets the dimensionality of the LSTM hidden size were set to 256. We set the batch size to 32 with the learning rate to 0.0001. Moreover, all models are trained using Adam optimization algorithm [48] with the weight decay penalty of 10^{-5} for 50 epochs. We also use dropout [21] of 0.5 at all connections without LSTM networks. In addition, our implementation is based on Pytorch with the NVIDIA Titan X.

2) Recurrence Number: As we all know, LSTM networks is a type of RNN, and the unique characteristic of RNN is recurrence. In this context, the number of recurrences in our network is an important parameter, which is related to the complexity and the expressive power of the model. Theoretically speaking, the increasing number of recurrence will enhance the ability to generate suitable representation, but it will also increase the difficulty of training model. In this paper, we select several values of this parameter and make the comparison among classification accuracy and loss, which can be seen in Fig. 10 and Table I. According to the experimental results, the classification accuracy tends to increase and the loss descent tends to be fast with large values of this parameter, but this effect becomes not obvious when it increases to a certain level. In this case, we apply 20 recurrences in our network.

3) Layer Number: The number of layers is very similar to the recurrence number. The former determines the horizontal depth of our network and the latter determines the longitudinal depth. Experimental results about this parameter can be seen in Fig. 11 and Table II. In this case, too deep network will cause classification accuracy decrease and loss descent slowness. So we stack 3 LSTM layers in our network.

TABLE I
THE INFLUENCE OF RECURRENCE NUMBER ON THE UC MERCED LAND-USE DATASET

Recurrence Number	OA(%)
3	96.8
5	96.9
10	97.2
20	97.6
30	97.7

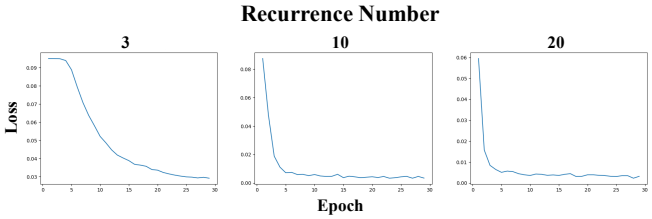


Fig. 10. The comparison of loss descent when changing recurrence number of deep LSTM network.

4) Combination Approach: The three approaches we proposed to combine the outputs of LSTM network are shown in Fig. 8, and the experimental results can be seen in Fig. 12. In this case, the classification accuracy using these three approaches tends to be very similar, but the first approach causes a slower loss descent than the other two. Therefore, we choose the third one in our network.

5) Convolutional Feature Extractor: Different structures of CNN will generate different representations of the same image. In the meanwhile, our *ARCNet* can be applied to any type of CNN. So we test several CNN as our feature extractor in order to find a proper one for the task of remote sensing scene classification. The first CNN we choose is AlexNet, a classic network for the challenge of ImageNet, which has

TABLE II
THE INFLUENCE OF LAYER NUMBER ON THE UC MERCED LAND-USE DATASET

Layer Number	OA(%)
1	97.3
3	97.6
5	96.6

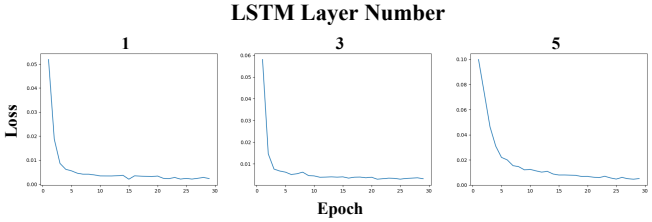


Fig. 11. The comparison of loss descent when changing layer number of deep LSTM network.

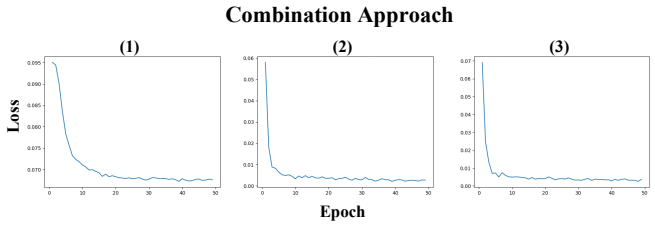


Fig. 12. The comparison of loss descent when changing combination approach of deep LSTM network.

TABLE III
THE INFLUENCE OF CONVOLUTIONAL FEATURES EXTRACTOR ON THE UC MERCED LAND-USE DATASET

pre-trained CNNs	OA(%)
AlexNet	97.6
VGGNet16	99.5
ResNet34	98.8

relatively small depth than other networks. VGGNet is the second network we applied. This network can be set to different layers and we choose VGG-16 in this paper. The last network we test is ResNet, which is famous for its ingenious structure called residual. This structure can let the gradient be better transferred to achieve a better learning. It can also be set to different layers, and we choose 34 layers which is called ResNet-34. The classification accuracy comparison of different CNN feature extractors is shown in Table III, and it is not difficult to find that VGGNet-16 can achieve the highest performance. In this case, we choose VGGNet-16 as our CNN feature extractor.

D. Performance

In this section, the performance comparison between our network and some state-of-the-art methods will be discussed. The details of our *ARCNet* are shown in Table IV, and the evaluation metrics are overall accuracy and confusion matrix.

1) UC Merced Land Use Dataset: We perform a comparative evaluation of the proposed *ARCNet* against some state-of-the-art remote sensing scene classification methods on the UCM dataset, as shown in Table V. According to the discussion in the previous section, we apply the pre-trained VGGNet-16 as our feature extractor and name the proposed *ARCNet* as ARCNet-VGG16. In the mean time, this dataset is

TABLE IV
THE DETAILS OF *ARCNet*

CNN Feature Extrctor	VGGNet-16
Layer Number	3
Recurrence Number	20
Feature Size	$7 \times 7 \times 512$
Attention Size	$7 \times 7 \times 20$
Hidden Size	256

the most widely used dataset, so there are a lot of methods to compare including handcrafted features based, unsupervised learning features based and deep learning feature based. As we can see in Table V, our network obviously outperforms all other scene classification methods in OA whether its training ratio is 80% or 50%. When using 80% labeled images for training, our ARCNet-VGG16 makes an increase of 1.04 percentage points over the second best method named as Combing Scenarios I and II [16]. When using 50% labeled images for training, this type of increase is 1.99 percentage points compared with SalM3LBPCLM [18]. Fusion by Addition [25] is the most effective method for remote sensing scene classification, and the proposed ARCNet-VGG16 can makes an increase of 1.7% percentage points than this method. In addition, our ARCNet-VGG16 is based on the high-level features extracted from VGGNet-16, so the comparison with normal VGGNet-16 is very necessary. As shown in Table V, VGG-VD-16 [17] gets $95.21 \pm 1.20\%$ at 80% training samples and $94.14 \pm 0.69\%$ at 50% training samples, which confirms attention mechanism actually makes a great contribution to the promotion of classification accuracy.

We also make a confusion matrix to further analyze the effect of the proposed *ARCNet*, as shown in Fig. 13. In this figure, one freeway image is incorrectly classified as overpass and one medium residential image is classified as sparse residential. These two categories are very confusing so that other methods usually get much lower accuracy. For example, Fine-tuning GoogLeNet [15] only achieves 75% for the class of freeway, but our method get 95%. Fusion by Addition [25] only achieves 65% for the class of dense residential, but our method get 100%. It confirms that our *ARCNet* is very good at handling detailed information, and this is also attributed to the application of *recurrent attention structure* which can focus on key regions of the scene.

According to the experimental results on UCM dataset, it can be concluded that the proposed *ACRNet* has great superiority compared to other state-of-the-art methods and the attention mechanism indeed makes an important role in learning scenes. Besides, it is necessary to further explain the computational efficiency of the proposed ARCNet-VGGNet16. We set the batchsize to 1 and use 2000 images for inference. After calculating the average time, our network needs about $61.29ms$ per image when inference.

2) WHU RS19 Dataset: In order to further evaluate the performance of the proposed *ACRNet*, we make a comparison of OA with several state-of-the-art scene classification methods on the RS19 dataset, as shown in Table VI. For this dataset, the most important differences compared on UCM dataset are the larger resolution and smaller number of images per class. Moreover, because of the relatively less usage than the previous dataset, there are fewer comparison methods. This experiment is also divided into two parts, 60% samples for training and 40% for training. From the results, we can find that this dataset is less challenging than the previous one because the same method usually gets higher accuracy. It needs to be mentioned that our ARCNet-VGG16 also wins the first place than all other methods, and the best result achieves 100%

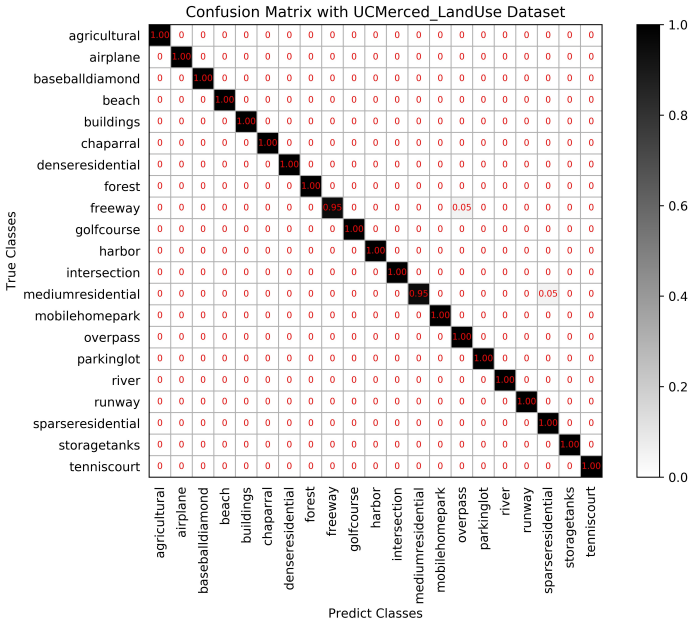


Fig. 13. The confusion matrix on UC Merced Land Use dataset under the training ratio of 80%.

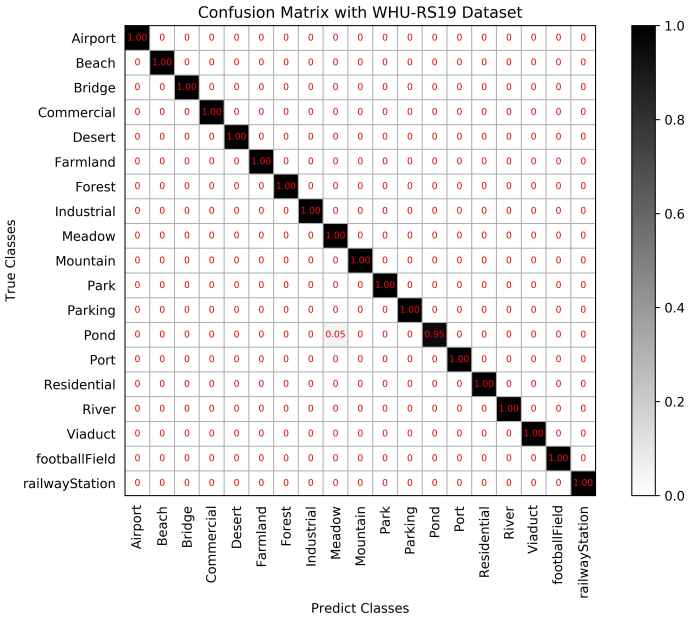


Fig. 14. The confusion matrix on WHU RS19 dataset under the training ratio of 60%.

on the validation dataset.

The confusion matrix on this dataset is shown in Fig. 14, and there is only one misclassified image. According to the results of other methods, this dataset is simpler than the previous one so that most methods usually get higher accuracy as well as our *ARCNet*. In conclusion, this experiment further proves the effectiveness of the proposed method.

3) Aerial Image Dataset: The number of images in previous two datasets is very limited, so it is necessary to evaluate our *ACRNet* on an larger dataset. That is also the reason why we choose AID as the experimental dataset. AID consists of

TABLE V
OVERALL ACCURACY (%) OF DIFFERENT METHODS ON THE UC MERCED LAND-USE DATASET

Methods	80% images for training	50% images for training
ARCNet-VGG16	**99.12** $\pm$ 0.40	**96.81** $\pm$ 0.14
Combing Scenarios I and II [16]	98.49	
Fusion by Addition [25]	97.42 $\pm$ 1.79	
CNN-NN [38]	97.19	
Fine-tuning GoogLeNet [15]	97.10	
GoogLeNet [17]	94.31 $\pm$ 0.89	92.70 $\pm$ 0.60
CaffeNet [17]	95.02 $\pm$ 0.81	93.98 $\pm$ 0.67
Overfeat [39]	90.91 $\pm$ 1.19	
VGG-VD-16 [17]	95.21 $\pm$ 1.20	94.14 $\pm$ 0.69
MS-CLBP+FV [49]	93.00 $\pm$ 1.20	88.76 $\pm$ 0.76
Gradient Boosting Random CNNs [50]	94.53	
SalM3LBPCLM [18]	95.75 $\pm$ 0.80	94.21 $\pm$ 0.75
Partlets-based [51]	91.33 $\pm$ 1.11	
Multifeature Concatenation [52]	92.38 $\pm$ 0.62	
Pyramid of Spatial Relations [53]	89.10	
Saliency-guided Feature Learning [44]	82.72 $\pm$ 1.18	
Unsupervised Feature Learning [13]	81.67 $\pm$ 1.23	
BoVW [15]	76.81	

TABLE VI
OVERALL ACCURACY (%) OF DIFFERENT METHODS ON THE WHU-RS19 DATASET

Methods	60% images for training	40% images for training
ARCNet-VGG16	**99.75** $\pm$ 0.25	**97.50** $\pm$ 0.49
Combing Scenarios I and II [16]	98.89	
Fusion by Addition [25]	98.65 $\pm$ 0.43	
VGG-VD-16 [17]	96.05 $\pm$ 0.91	95.44 $\pm$ 0.60
CaffeNet [17]	96.24 $\pm$ 0.56	95.11 $\pm$ 1.20
GoogLeNet [17]	94.71 $\pm$ 1.33	93.12 $\pm$ 0.82
SalM3LBPCLM [18]	96.38 $\pm$ 0.82	95.35 $\pm$ 0.76
Multifeature Concatenation [52]	94.53 $\pm$ 1.01	
MS-CLBP+FV [49]	94.32 $\pm$ 1.02	
MS-CLBP+BoVW [49]	89.29 $\pm$ 1.30	
Bag of SIFT [53]	85.52 $\pm$ 1.23	

TABLE VII
OVERALL ACCURACY (%) OF DIFFERENT METHODS ON THE AID

Methods	50% images for training	20% images for training
ARCNet-VGG16	**93.10** $\pm$ 0.55	**88.75** $\pm$ 0.40
Fusion by Addition [25]	91.87 $\pm$ 0.36	
VGG-VD-16 [17]	89.64 $\pm$ 0.36	86.59 $\pm$ 0.29
CaffeNet [17]	89.53 $\pm$ 0.31	86.86 $\pm$ 0.47
GoogLeNet [17]	86.39 $\pm$ 0.55	83.44 $\pm$ 0.40

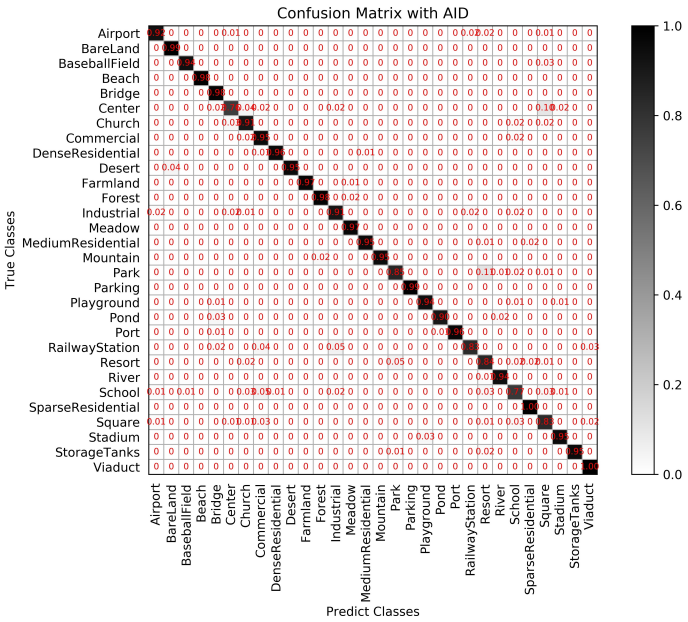


Fig. 15. The confusion matrix on AID under the training ratio of 50%.

TABLE VIII
OVERALL ACCURACY (%) OF DIFFERENT METHODS FOR ON OPTIMAL-31 DATASET

Methods	80% images for training
ARCNet-VGGNet16	**92.70** $\pm$ 0.35
ARCNet-ResNet34	91.28 $\pm$ 0.45
ARCNet-Alexnet	85.75 $\pm$ 0.35
VGG-VD-16 [17]	89.12 $\pm$ 0.35
Fine-tuning VGGNet16	87.45 $\pm$ 0.45
Fine-tuning GoogLeNet	82.57 $\pm$ 0.12
Fine-tuning AlexNet	81.22 $\pm$ 0.19

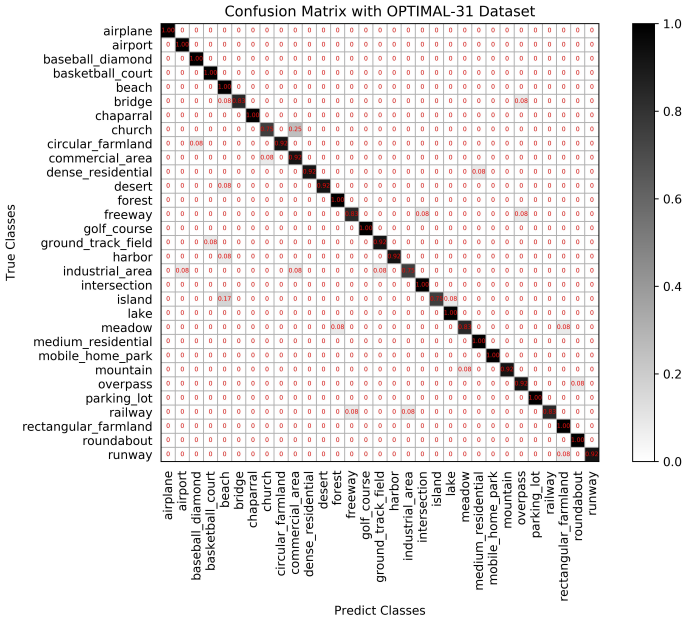


Fig. 16. The confusion matrix on OPTIMAL-31 dataset under the training ratio of 80%.

10000 images distributed among 30 classes, and each image has a size of 600×600 pixels, and we choose 50% and 20% as the training ratio. As we can be seen in Table VII, under the training ratio of 50%, the proposed *ACRNet* makes an increase of 1.23 percentage points over the second best method named as Fusion by Addition [25]. Besides, when using 40% samples for training, the proposed *ACRNet* also achieves the first place than other stat-of-the-art methods.

From the experimental results we can find that AID is much more difficult than the previous two datasets, and the performance of our method is not perfect. Therefore, it is necessary to re-analyze the experimental result through the confusion matrix which is shown in Fig. 15. From this figure we can find that center and school are two confusing categories which get the lowest classification accuracy among the whole 30 classes. Both of these two categories are part of the class of architecture so that there exist many similar ones. That is the reason why these two cannot achieve better performance.

4) OPTIMAL-31 Dataset: In the previous two experiments, the superiority of the proposed *ARCNet* has been well demonstrated. When we validate it on the OPTIMAL-31 dataset constructed by ourselves, the same conclusion is obtained. Therefore, we only list the experimental results of our method to save space. We use 80% images for training and test the *ARCNet* with different CNN feature extractor named as ARCNet-VGG16, ARCNet-ResNet34 and ARCNet-AlexNet, as shown in Table VIII. For the experimental results, the best classification accuracy is 92.70 ± 0.35 which is far below the other two datasets, and this also illustrates the difficulty of our dataset on the other hand. The comparing algorithm we validated on this dataset is VGG-VD-16 [17], and we also trained some baseline networks (Fine-tuning VGGNet16, Fine-tuning GoogLeNet and Fine-tuning AlexNet) for better explaining the effectiveness of the proposed *ARCNet*. Our network makes an increase of 3.58 percentage points than the VGG-VD-16 [17] and almost 5 percentage points than the corresponding baseline networks. This proves that the attention mechanism can indeed improve the classification accuracy in the domain of remote sensing scene classification.

As for the confusion matrix shown in Fig. 16, misclassification appears in many categories especially in these two: church and island. The phenomenon of low accuracy on this dataset is caused by many reasons. First, this dataset contains many confusing categories leading to the misclassification directly. Second, it has 31 classes which is more complex than the previous ones. Finally, each class has 60 images and only 48 for training, which is not large enough to learn enough distinguishing features with large class number. There are two ways to solve these problems. One is to employ a more powerful CNN to extract high-level features, and the other is to increase the size of dataset to ensure the enough training set.

V. CONCLUSION

In this paper, we propose a novel *recurrent attention structure* for remote sensing scene classification and then construct an effective *ARCNet* based on it. The proposed

method is inspired by the attention mechanism of human visual system and enables classifiers focusing on key areas through learning, which can greatly accelerate the convergence rate and improve the accuracy. We also construct a new dataset named OPTIMAL-31 to validate the proposed *ARCNet*. The experimental results show that it outperforms the current state-of-the-art methods, and it is effective to apply attention mechanism to the task of remote sensing scene classification. In the future work, we will further explore the attention mechanism and design more powerful model for other remote sensing applications.

REFERENCES

[1] L. Gómez-Chova, D. Tuia, G. Moser, and G. Camps-Valls, "Multimodal classification of remote sensing images: A review and future directions," *Proceedings of the IEEE*, vol. 103, no. 9, pp. 1560–1584, 2015.

[2] P. Gamba, "Human settlements: A global challenge for eo data processing and interpretation," *Proceedings of the IEEE*, vol. 101, no. 3, pp. 570–581, 2013.

[3] G. Cheng, L. Guo, T. Zhao, J. Han, H. Li, and J. Fang, "Automatic landslide detection from remote-sensing imagery using a scene classification method based on bovw and plsa," *International Journal of Remote Sensing*, vol. 34, no. 1, pp. 45–59, 2013.

[4] T. R. Martha, N. Kerle, C. J. van Westen, V. Jetten, and K. V. Kumar, "Segment optimization and data-driven thresholding for knowledge-based landslide detection by object-based image analysis," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 49, no. 12, pp. 4928–4943, 2011.

[5] X. Yao, J. Han, G. Cheng, X. Qian, and L. Guo, "Semantic annotation of high-resolution satellite images via weakly supervised learning," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 54, no. 6, pp. 3660–3671, 2016.

[6] S. Cui, "Comparison of approximation methods to kullback–leibler divergence between gaussian mixture models for satellite image retrieval," *Remote Sensing Letters*, vol. 7, no. 7, pp. 651–660, 2016.

[7] G. Cheng, P. Zhou, and J. Han, "Learning rotation-invariant convolutional neural networks for object detection in vhr optical remote sensing images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 54, no. 12, pp. 7405–7415, 2016.

[8] S. Bhagavathy and B. S. Manjunath, "Modeling and detection of geospatial objects using texture motifs," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 44, no. 12, pp. 3706–3715, 2006.

[9] Y. Wang, L. Zhang, X. Tong, L. Zhang, Z. Zhang, H. Liu, X. Xing, and P. T. Mathiopoulos, "A three-layered graph-based learning approach for remote sensing image retrieval," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 54, no. 10, pp. 6020–6034, 2016.

[10] Y. Yang and S. Newsam, "Geographic image retrieval using local invariant features," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 51, no. 2, pp. 818–832, 2013.

[11] J. Muñoz-Marí, F. Bovolo, L. Gómez-Chova, L. Bruzzone, and G. Camp-Valls, "Semisupervised one-class support vector machines for classification of remote sensing data," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 48, no. 8, pp. 3188–3197, 2010.

[12] D. Tuia, F. Pacifici, M. Kanevski, and W. J. Emery, "Classification of very high spatial resolution imagery using mathematical morphology and support vector machines," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 47, no. 11, pp. 3866–3879, 2009.

[13] A. M. Cheriyadat, "Unsupervised feature learning for aerial scene classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 52, no. 1, pp. 439–451, 2014.

[14] G. Cheng, J. Han, and X. Lu, "Remote sensing image scene classification: Benchmark and state of the art," *Proceedings of the IEEE*, vol. 105, no. 10, pp. 1865–1883, 2017.

[15] M. Castelluccio, G. Poggi, C. Sansone, and L. Verdoliva, "Land use classification in remote sensing images by convolutional neural networks," *arXiv preprint arXiv:1508.00092*, 2015.

[16] F. Hu, G.-S. Xia, J. Hu, and L. Zhang, "Transferring deep convolutional neural networks for the scene classification of high-resolution remote sensing imagery," *Remote Sensing*, vol. 7, no. 11, pp. 14 680–14 707, 2015.

[17] G.-S. Xia, J. Hu, F. Hu, B. Shi, X. Bai, Y. Zhong, L. Zhang, and X. Lu, "Aid: A benchmark data set for performance evaluation of aerial scene classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 55, no. 7, pp. 3965–3981, 2017.

[18] X. Bian, C. Chen, L. Tian, and Q. Du, "Fusing local and global features for high-resolution scene classification," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 10, no. 6, pp. 2889–2901, 2017.

[19] R. A. Rensink, "The dynamic representation of scenes," *Visual Cognition*, vol. 7, no. 1-3, pp. 17–42, 2000.

[20] M. Corbetta and G. L. Shulman, "Control of goal-directed and stimulus-driven attention in the brain," *Nature reviews neuroscience*, vol. 3, no. 3, pp. 201–215, 2002.

[21] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems*, 2012, pp. 1097–1105.

[22] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[23] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 770–778.

[24] Q. Wang, J. Gao, and Y. Yuan, "Embedding structured contour and location prior in siamesed fully convolutional networks for road detection," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 1, pp. 230–241, 2018.

[25] S. Chaib, H. Liu, Y. Gu, and H. Yao, "Deep feature fusion for vhr remote sensing scene classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 55, no. 8, pp. 4775–4784, 2017.

[26] S. Chaib, H. Yao, Y. Gu, and M. Amrani, "Deep feature extraction and combination for remote sensing image classification based on pre-trained cnn models," in *Ninth International Conference on Digital Image Processing (ICDIP 2017)*, vol. 10420. International Society for Optics and Photonics, 2017, p. 104203D.

[27] J. A. dos Santos, O. A. B. Penatti, and R. da Silva Torres, "Evaluating the potential of texture and color descriptors for remote sensing image retrieval and classification." in *VISAPP (2)*, 2010, pp. 203–208.

[28] V. Risojević and Z. Babić, "Fusion of global and local descriptors for remote sensing image classification," *IEEE Geoscience and Remote Sensing Letters*, vol. 10, no. 4, pp. 836–840, 2013.

[29] G. Cheng, J. Han, P. Zhou, and L. Guo, "Multi-class geospatial object detection and geographic image classification based on collection of part detectors," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 98, pp. 119–132, 2014.

[30] G. Cheng, P. Zhou, J. Han, L. Guo, and J. Han, "Auto-encoder-based shared mid-level visual dictionary learning for scene classification using very high resolution remote sensing images," *IET Computer Vision*, vol. 9, no. 5, pp. 639–647, 2015.

[31] Q. Zhu, Y. Zhong, B. Zhao, G.-S. Xia, and L. Zhang, "Bag-of-visual-words scene classifier with local and global features for high spatial resolution remote sensing imagery," *IEEE Geoscience and Remote Sensing Letters*, vol. 13, no. 6, pp. 747–751, 2016.

[32] J. Zou, W. Li, C. Chen, and Q. Du, "Scene classification using local and global features with collaborative representation fusion," *Information Sciences*, vol. 348, pp. 209–226, 2016.

[33] M. L. Mekhalfi, F. Melgani, Y. Bazi, and N. Alajlan, "Land-use classification with compressive sensing multifeature fusion," *IEEE Geoscience and Remote Sensing Letters*, vol. 12, no. 10, pp. 2155–2159, 2015.

[34] G. Sheng, W. Yang, T. Xu, and H. Sun, "High-resolution satellite scene classification using a sparse coding based multiple feature combination," *International Journal of Remote Sensing*, vol. 33, no. 8, pp. 2395–2412, 2012.

[35] Q. Wang, J. Wan, and Y. Yuan, "Locality constraint distance metric learning for traffic congestion detection," *Pattern Recognition*, vol. 75, pp. 272–281, 2018.

[36] B. Du, M. Zhang, L. Zhang, R. Hu, and D. Tao, "Pltd: Patch-based low-rank tensor decomposition for hyperspectral images," *IEEE Transactions on Multimedia*, vol. 19, no. 1, pp. 67–79, 2017.

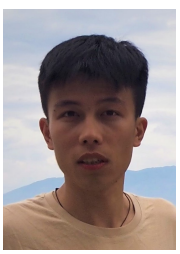
[37] S. Chaib, Y. Gu, and H. Yao, "An informative feature selection method based on sparse pca for vhr scene classification," *IEEE Geoscience and Remote Sensing Letters*, vol. 13, no. 2, pp. 147–151, 2016.

[38] E. Othman, Y. Bazi, N. Alajlan, H. Alhichri, and F. Melgani, "Using convolutional features and a sparse autoencoder for land-use scene classification," *International Journal of Remote Sensing*, vol. 37, no. 10, pp. 2149–2167, 2016.

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

[40] Q. Wang, J. Wan, and Y. Yuan, "Deep metric learning for crowdedness regression," *IEEE Transactions on Circuits and Systems for Video Technology*, DOI: 10.1109/TCSVT.2017.2703920, 2018.
[41] Q. Wang, J. Gao, and Y. Yuan, "A joint convolutional neural networks and context transfer for street scenes labeling," *IEEE Transactions on Intelligent Transportation Systems*, 2017.
[42] B. Du, W. Xiong, J. Wu, L. Zhang, L. Zhang, and D. Tao, "Stacked convolutional denoising auto-encoders for feature representation," *IEEE transactions on cybernetics*, vol. 47, no. 4, pp. 1017–1027, 2017.
[43] B. Zhou, A. Lapedriza, J. Xiao, A. Torralba, and A. Oliva, "Learning deep features for scene recognition using places database," in *Advances in Neural Information Processing Systems*, 2014, pp. 487–495.
[44] F. Zhang, B. Du, and L. Zhang, "Saliency-guided unsupervised feature learning for scene classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 53, no. 4, pp. 2175–2184, 2015.
[45] H. Zhang, J. Zhang, and F. Xu, "Land use and land cover classification base on image saliency map cooperated coding," in *Image Processing (ICIP), 2015 IEEE International Conference on*. IEEE, 2015, pp. 2616–2620.
[46] L. Zhang and Y. Zhang, "Airport detection and aircraft recognition based on two-layer saliency model in high spatial resolution remote-sensing images," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 10, no. 4, pp. 1511–1524, 2017.
[47] Y. Yang and S. Newsam, "Bag-of-visual-words and spatial extensions for land-use classification," in *Proceedings of the 18th SIGSPATIAL International Conference on Advances in Geographic Information Systems*. ACM, 2010, pp. 270–279.
[48] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.
[49] L. Huang, C. Chen, W. Li, and Q. Du, "Remote sensing image scene classification using multi-scale completed local binary patterns and fisher vectors," *Remote Sensing*, vol. 8, no. 6, p. 483, 2016.
[50] F. Zhang, B. Du, and L. Zhang, "Scene classification via a gradient boosting random convolutional network framework," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 54, no. 3, pp. 1793–1802, 2016.
[51] G. Cheng, J. Han, L. Guo, Z. Liu, S. Bu, and J. Ren, "Effective and efficient midlevel visual elements-oriented land-use classification using vhr remote sensing images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 53, no. 8, pp. 4238–4249, 2015.
[52] W. Shao, W. Yang, G.-S. Xia, and G. Liu, "A hierarchical scheme of multiple feature fusion for high-resolution satellite scene categorization," in *International Conference on Computer Vision Systems*. Springer, 2013, pp. 324–333.
[53] S. Chen and Y. Tian, "Pyramid of spatial relatons for scene-level land use classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 53, no. 4, pp. 1947–1957, 2015.

Qi Wang (M'15-SM'15) received the B.E. degree in automation and the Ph.D. degree in pattern recognition and intelligent systems from the University of Science and Technology of China, Hefei, China, in 2005 and 2010, respectively. He is currently a Professor with the School of Computer Science, with the Unmanned System Research Institute, and with the Center for OPTical IMagery Analysis and Learning (OPTIMAL), Northwestern Polytechnical University, Xi'an, China. His research interests include computer vision and pattern recognition.



Shaoteng Liu is currently working toward the M.E. degree in the School of Computer Science and the Center for OPTical IMagery Analysis and Learning (OPTIMAL), Northwestern Polytechnical University, Xi'an, China. His current research interests include scene classification of remote sensing image, hyperspectral image classification, and deep learning.

Jocelyn Chanussot (M'04-SM'04-F'12) received the M.Sc. degree in electrical engineering from the Grenoble Institute of Technology (Grenoble INP), Grenoble, France, in 1995, and the Ph.D. degree from Savoie University, Annecy, France, in 1998.

In 1999, he joined the Geography Imagery Perception Laboratory, Delegation Generale de l'Armement, in Arcueil, France (DGA-French National Defense Department). From 1999 to 2005, he was an Assistant Professor with Grenoble INP, where he was an Associate Professor from 2005 to 2007. He has been a Visiting Scholar at Stanford University, Stanford, CA, USA; KTH, Stockholm, Sweden; and NUS, Singapore. Since 2013, he has been an Adjunct Professor with the University of Iceland, Reykjavik, Iceland. From 2014 to 2015, he was a Visiting Professor at the University of California, Los Angeles, CA, USA. He is currently a Professor of signal and image processing with the Grenoble INP. He is conducting his research at the Grenoble Images Speech Signals and Automatics Laboratory (GIPSA-Lab). His research interests include image analysis, multicomponent image processing, nonlinear filtering, and data fusion in remote sensing.

Dr. Chanussot was a member of the Machine Learning for Signal Processing Technical Committee of the IEEE Signal Processing Society during 2006-2008 and the IEEE Geoscience and Remote Sensing Society (GRSS) AdCom during 2009-2010, in charge of the membership development. He is a Fellow of the IEEE and a member of the Institut Universitaire de France (2012-2017). He was a co-recipient of the NORSIG 2006 Best Student Paper Award, the IEEE GRSS 2011 and 2015 Symposium Best Paper Awards, the IEEE GRSS 2012 Transactions Prize Paper Award, and the IEEE GRSS 2013 Highest Impact Paper Award. He served as the Founding President of the IEEE GRSS French Chapter during 2007-2010 which received the 2010 IEEE GRSS Chapter Excellence Award. He was the General Chair of the first IEEE GRSS Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing. He was the Chair during 20092011 and the Co-Chair of the GRS Data Fusion Technical Committee during 2005-2008. He was the Program Chair of the IEEE International Workshop on Machine Learning for Signal Processing in 2009. He was an Associate Editor of the IEEE GEOSCIENCE AND REMOTE SENSING LETTERS during 2005-2007 and for the *Pattern Recognition* during 2006-2008. Since 2007, he has been an Associate Editor of the IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING. From 2011 to 2015, he was the Editor-in-Chief of the IEEE JOURNAL OF SELECTED TOPICS IN APPLIED EARTH OBSERVATIONS AND REMOTE SENSING. He served as a Guest Editor of the PROCEEDINGS OF THE IEEE, in 2013, and the IEEE SIGNAL PROCESSING MAGAZINE in 2014.

Xuelong Li (M'02-SM'07-F'12) is a Full Professor with the Center for OPTical IMagery Analysis and Learning (OPTIMAL), State Key Laboratory of Transient Optics and Photonics, Xi'an Institute of Optics and Precision Mechanics, Chinese Academy of Sciences, Xi'an 710119, Shaanxi, PR China, and the University of Chinese Academy of Sciences, Beijing 100049, PR China.