remote sensing

MDPI

Article

Dense Connectivity Based Two-Stream Deep Feature Fusion Framework for Aerial Scene Classification

Yunlong Yu * and Fuxian Liu

Institute of Air Defense and Anti-Missile, Air Force Engineering University, Xi'an 710051, China; liuxqh@126.com

* Correspondence: yuyunlong123@gmail.com; Tel.: +86-170-8626-5379

Received: 5 June 2018; Accepted: 19 July 2018; Published: 23 July 2018

Abstract: Aerial scene classification is an active and challenging problem in high-resolution remote sensing imagery understanding. Deep learning models, especially convolutional neural networks (CNNs), have achieved prominent performance in this field. The extraction of deep features from the layers of a CNN model is widely used in these CNN-based methods. Although the CNN-based approaches have obtained great success, there is still plenty of room to further increase the classification accuracy. As a matter of fact, the fusion with other features has great potential for leading to the better performance of aerial scene classification. Therefore, we propose two effective architectures based on the idea of feature-level fusion. The first architecture, i.e., texture coded two-stream deep architecture, uses the raw RGB network stream and the mapped local binary patterns (LBP) coded network stream to extract two different sets of features and fuses them using a novel deep feature fusion model. In the second architecture, i.e., saliency coded two-stream deep architecture, we employ the saliency coded network stream as the second stream and fuse it with the raw RGB network stream using the same feature fusion model. For sake of validation and comparison, our proposed architectures are evaluated via comprehensive experiments with three publicly available remote sensing scene datasets. The classification accuracies of saliency coded two-stream architecture with our feature fusion model achieve 97.79%, 98.90%, 94.09%, 95.99%, 85.02%, and 87.01% on the UC-Merced dataset (50% and 80% training samples), the Aerial Image Dataset (AID) (20% and 50% training samples), and the NWPU-RESISC45 dataset (10% and 20% training samples), respectively, overwhelming state-of-the-art methods.

Keywords: convolutional neural network (CNN); mapping local binary patterns (LBP) codes; saliency detection; aerial scene classification; two-stream deep feature fusion model; remote sensing

1. Introduction

Recently, with the rapid development of remote sensing technology and the growing deployment of remote sensing instruments, we have the opportunities to obtain a huge number of high-resolution remote sensing images [1–4]. This phenomenon generates a large demand for automatic and precise identification and classification of land-use and land-cover (LULC) scenes from these remote sensing images [5–10]. Aerial scene classification, which aims to automatically label an aerial scene image according to a set of semantic categories, has a wide range of applications, from civil to military purposes [11]. At the same time, aerial scene classification has drawn much attention from the remote sensing community.

Generally, effective feature extraction is the key for accurate aerial scene classification. Besides traditional methods using hand-crafted features [12–15], deep convolutional neural network (CNN)-based approaches yield amazing performance in recent years [16–19]. The frequently-used CNN models include CaffeNet [20], AlexNet [21], VGG Net [22], GoogLeNet [23], and ResNet [24]. So far, CNNs have

demonstrated outstanding ability to extract high-level discriminative features on aerial scene classification task. One can refer to [25–27] for more details.

Despite the impressive results achieved with CNN-based methods, learning effective features from aerial scene images still poses various challenges. First, unlike natural images, aerial scene images have high intra-class variations. Specifically, objects in the same scene category may appear at different sizes and different orientations. In addition, due to the different imaging conditions, e.g., the altitude of the imaging devices and the solar elevation angles, the appearance of the same scene may also be different. Second, the scene images belonging to different classes may share similar objects and structural variations, which results in small inter-class dissimilarity. Generally speaking, a good representation of aerial scene images is the key to success in this area. In other words, what features we use and how to use these features are becoming more and more important in the field of aerial scene classification.

In this paper, we propose two effective architectures to further improve the classification performance for aerial scene images, in which we employ a pre-trained CNN (GoogLeNet) as a feature extractor. The first architecture, i.e., texture coded two-stream deep architecture, uses raw RGB network stream and mapped local binary patterns (LBP) coded network stream to extract two different sets of features and fuses these two sets of features by using a novel deep feature fusion model. The second architecture, i.e., saliency coded two-stream deep architecture, investigates to what extent saliency coded network stream complement the raw RGB network stream and combines them using the same deep feature fusion model as the first architecture. In the experiments, we evaluate our architectures and compare them with several recent methodologies on three publicly available remote sensing scene datasets. Experimental results demonstrate that our proposed methods outperform the baseline methods and achieve state-of-the-art results on these three utilized datasets.

The remainder of this paper is arranged as follows. Section 2 represents the related works including aerial scene classification methods and feature fusion methods. Section 3 provides details about our proposed frameworks for aerial scene classification. The experimental results and analysis are presented in Section 4, followed by a discussion in Section 5. Finally, Section 6 summarizes this paper.

2. Related Works

In this section, we briefly review the methods for aerial scene classification and feature fusion.

2.1. Methods for Aerial Scene Classification

Generally speaking, the methods in the field of aerial scene classification can be categorized three sorts of methods, i.e., methods based on low-level scene features, methods based on mid-level scene features and methods based on high-level scene features, as shown in Figure 1. In what follows, we review these methods in detail.

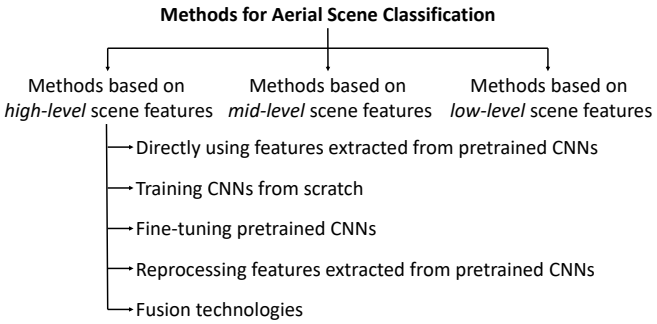


Figure 1. A categorization of the methods for aerial scene classification.

Methods based on low-level scene features: In these methods, the way of characterizing input image patches is to apply low-level scene features such as texture feature, structure feature,

and spectral feature, to name a few. Four low-level methods are commonly utilized, including LBP [28], Scale Invariant Feature Transform (SIFT) [29], Color Histogram (CH) [30] and GIST [31]. However, one type of low-level features may work badly in some cases; therefore, researchers investigate a combination of different types of low-level visual features to improve the results. Luo et al. [32] combined the radiometric features, the Gray Level Co-Occurrence Matrix (GLCM) features and the Gaussian wavelet features to produce a multiple-feature representation.

Methods based on mid-level scene features: Mid-level methods are more suitable for representing the complex image textures and structures with more compact feature vectors. Bag of Visual Words (BoVW) [33] is one of the most popular encoding approaches. Because of its good performance, this approach and its modifications have been widely used for obtaining mid-level representation of aerial scene images [34–38].

Furthermore, other approaches, e.g., Spatial Pyramid Matching (SPM) [39], Locality-constrained Linear Coding (LLC) [40], probabilistic Latent Semantic Analysis (pLSA) [41], Latent Dirichlet Allocation (LDA) [42], Improved Fisher Kernel (IFK) [43], and Vector of Locally Aggregated Descriptors (VLAD) [44] have also been proposed to produce mid-level scene features for aerial scene images.

Methods based on high-level scene features: Compared with the low- and mid-level scene features based approaches, deep learning based methods have the ability to obtain more significant and informative features, which results in better performance [45,46]. Aryal et al. [47] proposed a novel deep learning data analytics approach, i.e., restricted Boltzmann machine network, to learn urban objects from high resolution image. Moreover, Dutta et al. [48] utilized deep recurrent Elman neural network for accurately estimating continental fire incidence from climate data.

CNN is one of the most widely used deep learning algorithms, which is skilled in image processing. Therefore, we only focus on the CNN model in our manuscript. The deep features extracted by CNN are called high-level features. The methods based on high-level features can be divided into the following five groups:

(1) The approaches in the first group use the pre-trained CNNs to extract features from the input aerial scene images. In this group, all CNN models are pre-trained on the ImageNet 2012 dataset [49]. Penatti et al. [45] demonstrated the generalization ability of CNN models, i.e., OverFeat [50] and CaffeNet [20], in the scenario of aerial scene classification. Recently, two new large-scale aerial scene image datasets, i.e., Aerial Image Dataset (AID) [51] and Northwestern Polytechnical University-REmote Sensing Image Scene Classification 45 (NWPU-RESISC45) [52], have been constructed, and pre-trained CNN models achieved higher accuracies over the low- and mid-level features based methods.

(2) The methods in the second group train the CNN models from scratch, in which the filter weights are all randomly initialized and then trained for the target remote sensing scene datasets. In [53,54], the authors investigated the performance of the trained-from-scratch CNN models in the field of aerial scene classification, and the results showed that the trained-from-scratch CNN models get a drop in classification accuracy compared to the pre-trained CNN models. The authors thought that the relatively low performance of the trained-from-scratch CNN models was mainly due to the limited training data.

(3) The methods in the third group fine-tune the pre-trained CNN models on the target aerial scene datasets and use the fine-tuned architectures to extract features for classification. In [52–54], the authors fine-tuned some popular-used CNN models, and the experimental results pointed that the fine-tuning strategy can help the CNN models get much higher classification accuracies than both the full-training strategy and the "using pre-trained CNN models as feature extractors" strategy. In addition, Liu et al. [55] proposed a novel scene classification method through triplet networks, in which the triplet networks were pre-trained on ImageNet [49] and followed by fine-tuning over the target datasets. Their triplet networks reported a state-of-the-art performance in aerial scene classification tasks.

(4) The methods in the fourth group generate the final image representation by reprocessing the features extracted from the pre-trained CNN models. In [56], the multiscale dense convolutional features extracted from pre-trained CNNs were fed into four main parts, i.e., visual words encoding, correlogram extraction, correlation encoding, and classification, and state-of-the-art results were achieved. Cheng et al. [57] proposed a feature representation method for remote sensing image scene classification, termed bag of convolutional features (BoCF), which encodes the convolutional feature descriptors. Moreover, Yuan et al. [58] encoded the extracted deep feature by using the locality-constrained affine subspace coding (LASC) method, which can obtain more discriminative deep features than directly extracted from CNN models. In addition, Liu et al. [59] concatenated the extracted convolutional features to generate the deeply local descriptors, and subsequently, selected a feature encoding method, i.e., Fisher encoding with Gaussian mixture model (GMM) clustering, to process the deeply local descriptors. In another work, Liu et al. [60] proposed a linear transformation of deep features, in which the discriminative convolution filter (DCF) learning approach was performed on the local patches obtained from raw deep features.

(5) The methods in the fifth group use some fusion technologies to conduct the aerial scene classification. Anwer et al. [61] constructed a texture coded two-stream deep architecture which fuses both raw RGB features and texture coded features. However, its fusion approach is based on conventional concatenation strategy. Moreover, Chaib et al. [62] proposed to use discriminant correlation analysis (DCA) to process the extracted two sets of features, and combine the processed features through conventional fusion strategy, i.e., concatenation and addition. Li et al. [63] proposed to use a PCA/spectral regression kernel discriminant analysis (SRKDA) method to fuse the multiscale convolutional features encoded by a multiscale improved Fisher kernel coding method and the features of fully connected layers. In addition, Ye et al. [64] utilized the features from intermediate layers, and subsequently, created a parallel multi-stage (PMS) architecture formed by three sub-models, i.e., the low CNN, the middle CNN and the high CNN. The study [65] adaptively combined the features from lower layers and fully connected layers, in which the feature fusion operation was performed via a linear combination instead of concatenation. Their classification results all showed significant advantages over those "stand-alone" approaches. At the same time, some score-level fusion approaches [11,66] were proposed for aerial scene classification, which can also achieve impressive performance on the publicly available remote sensing scene datasets.

2.2. Feature Fusion

In the section of methods based on high-level scene features, we have introduced some popular feature-level fusion methods for aerial scene classification. Nevertheless, creating a superior feature-level fusion strategy for higher aerial scene classification accuracy is still a challenging task. In this section, we briefly review some feature fusion methods which are mainly used in video action recognition tasks and face recognition tasks. The idea of these methods can provide guidance for designing our new feature-level fusion method for aerial scene classification task.

Chowdhury et al. [67] used one intriguing new architecture termed bilinear CNN (BCNN) [68] to conduct face recognition, in which fusion of two CNN features was achieved by computing the outer product of convolutional-layer outputs. BCNN was also used for video action recognition [69]. Moreover, Park et al. [70] proposed to apply element-wise product to fuse multiple sources of knowledge in deep CNNs for video action recognition. In addition, a neural network based feature fusion method has been proposed in [71] where the spatial and the motion features are fused for video classification. In another work, Bodla et al. [72] also proposed a deep feature fusion network to fuse the features generated by different CNNs for template-based face recognition.

3. Proposed Architecture

Here, we first give the description of the methods of mapping LBP codes and saliency detection. Then, we describe the texture coded two-stream deep architecture and the saliency coded two-stream deep architecture. In the end, we propose a novel two-stream deep feature fusion model.

3.1. Mapping LBP Codes

LBP codes have been widely used in the field of texture recognition. At the same time, LBP is one of the most commonly used methods for the description of texture. In addition to the texture recognition and description, LBP codes have also been used for tasks such as person detection, face recognition, and image retrieval, to name a few.

LBP codes capture the local gray-scale distribution, produced by applying thresholds on the intensity values of the pixels in small neighborhoods by the intensity value of each neighborhood's central pixel as the threshold. The resulting codes are binary numbers ((0) lower than threshold value or (1) higher than the threshold value). The LBP codes which is used for describing the image texture features are calculated by,

$$C_{N,R} = \sum_{i=0}^{N-1} s(g_i - g_c)2^i, \tag{1}$$

where $g_i(i = 1, \cdots, N)$ denotes the N sampling points in the circular region, g_c is the center of the circular region, 2^i is a binomial factor for each sign $s(g_i - g_c)$, and R is the radius of this region. The thresholding function $s(t)$ is defined as follows:

$$s(t) = \begin{cases} 1, & x \geq 0, \\ 0, & x < 0. \end{cases} \tag{2}$$

The sign function describes not only the two states between the center point and its neighbors, but also shorten the difference value range of the joint distribution. In this way, the simple values can represent the joint distribution.

Under circumstances of the neighborhood containing eight other pixels, the binary string through the LBP code computation is treated as an eight-bit number between 0 and 255. The final representation is acquired by computing the histogram of LBP codes from the entire image region. The final representation normalizes for translation and is invariant to monotonic photometric transformations.

The unordered nature of the LBP code values is not suited as the input of CNN models. Levi et al. [73] proposed a LBP code mapping method which uses the Multi Dimensional Scaling (MDS) [74,75] to transform the unordered LBP code values to points in a metric space. The dissimilarity matrix Δ can be defined as:

$$\mathbf{\Delta} := \begin{pmatrix} \delta_{11} & \delta_{12} & \delta_{13} & \dots & \delta_{1n} \\ \delta_{21} & \delta_{22} & \delta_{23} & \dots & \delta_{2n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \delta_{d1} & \delta_{d2} & \delta_{d3} & \dots & \delta_{dn} \end{pmatrix}, \tag{3}$$

$$\delta_{i,j} \approx \| V_i - V_j \| = \| MDS(C_i) - MDS(C_j) \|, \tag{4}$$

where C_i and C_j are the LBP codes, V_i and V_j are the mapping of codes C_i and C_j, respectively. $\delta_{i,j}$ denotes the distance (dissimilarity) between LBP codes C_i and C_j. The code-to-code dissimilarity score is based on an approximation to the Earth Mover's Distance (EMD) [76]. The EMD can account for both the different bit values and their locations.

$$EMD(C_i, C_j) = \| CDF(C_i) - CDF(C_j) \|_1, \tag{5}$$

where CDF is the Cumulative Distribution Function (CDF) of the bit values, $\|\cdot\|_1$ is the L1 norm. The authors of [73] also account for the cyclic nature of LBP codes. The modified, cyclic distance, $\delta^{'}(C_i, C_j)$ is defined as:

$$\delta^{'}(C_i, C_j) = min(\delta(C_i^{'}, C_j^{'}), \delta(rev(C_i^{'}), C_j^{'}), \delta(C_i^{'}, rev(C_j^{'}))), \tag{6}$$

where $C_i^{'}$ and $C_j^{'}$ denote the modified codes through appending a single 0-valued bit as the new least significant bit of each code. The distance δ is computed by Equation (5) and the $rev(\cdot)$ operation can rearrange code values in reverse order.

3.2. Saliency Detection

In general, the actual remote sensing scene image usually contains a large amount of interference information in addition to the distinctive visual region. Cognitive psychology studies have shown that the human visual system adopts a serial computing strategy in the analysis of a complex input scene image. This strategy employs the selective visual attention mechanism [77,78] to select the specific area of the scene, i.e., the distinctive visual region, according to the local characteristics of the image, and then move the region to the foveal area with high resolution through the rapid eye movement scan, which can achieve attention to the region for more detailed observation and analysis. The selective visual attention mechanism can help the brain to filter out the plain region and focus on the distinctive visual region. Zheng et al. [79] proposed a saliency detection model that could emulate the human visual attention. By employing the saliency detection model, we have the opportunity to obtain more significant and informative features.

The saliency value of patch I_m can be obtained through two stages, i.e., the global perspective and the local perspective.

The global perspective is obtained by computing the histogram of word occurrence:

$$I = \{frq(W_k^f)\}, \qquad W_k^f \in \Omega \quad f \in F, \tag{7}$$

$$\Omega = \{W_k^f\} = \{[W_1^{color}, \cdots, W_{N^{color}}^{color}]; [W_1^{texture}, \cdots, W_{N^{texture}}^{texture}]\}, \tag{8}$$

$$F = \{color, texture\}, \tag{9}$$

where k represents the word index, $frq(W_k^f)$ denotes the occurrence frequency of the visual word W_k^f, f represents the feature in the feature set F and Ω is the visual vocabulary.

Then, the novelty factor φ_k^f is calculated through the "repetition suppression principle" [80]:

$$\varphi_k^f = 1/frq(W_k^f). \tag{10}$$

Finally, the local perspective can get the dissimilarities between one local patch and the global image. The saliency value of patch I_m is computed as follows:

$$sal(I_m) = \sum_{f \in F} \sum_{k=1}^{N^f} frq^m(W_k^f) \cdot \varphi_k^f, \tag{11}$$

where $frq^m(W_k^f)$ denotes the frequency of occurrence of the word W_k^f for the local patch I_m.

3.3. Two-Stream Deep Architecture

It is well known that a large proportion of noted CNNs take RGB images in the ImageNet dataset [49] as input. Of course, the mapped LBP images and the processed images through saliency detection can also be fed into the CNN models, which can obtain texture coded features and saliency coded features, respectively.

In general, the two-stream deep architecture contains an RGB based CNN model, an auxiliary CNN model and a feature fusion model. In this work, we investigate approaches to complement the RGB based CNN model using different auxiliary models, i.e., the mapped LBP coded CNN model and the saliency coded CNN model.

3.3.1. Texture Coded Two-Stream Deep Architecture

Anwer et al. [61] proposed a deep model, i.e., TEX-Net-LF, by designing a two-stream architecture in which texture coded mapped images are used as the second stream and fuse it with the RGB image stream. However, the feature fusion strategy of the TEX-Net-LF model is a conventional concatenation strategy. In this part, the mapped LBP coded CNN model is used as the auxiliary model, which complements the raw RGB coded CNN model. Here, we propose an effective texture coded two-stream deep architecture which combines the RGB and mapped LBP coded networks by using a novel deep feature fusion strategy. The raw RGB network stream and the mapped LBP coded network stream can capture the appearance and texture information respectively. In this architecture, we employ pre-trained CNN model as feature extractor to deal with the raw RGB images and the mapped LBP images. Figure 2 shows this texture coded two-stream deep architecture designed to combine the raw RGB network stream and the mapped LBP coded network stream. The raw RGB network stream takes RGB images as input, whereas the mapped LBP coded network stream takes mapped LBP images as input. In our framework, both raw RGB network stream and mapped LBP coded network stream are trained separately on the ImageNet ILSVRC 2012 dataset [49].

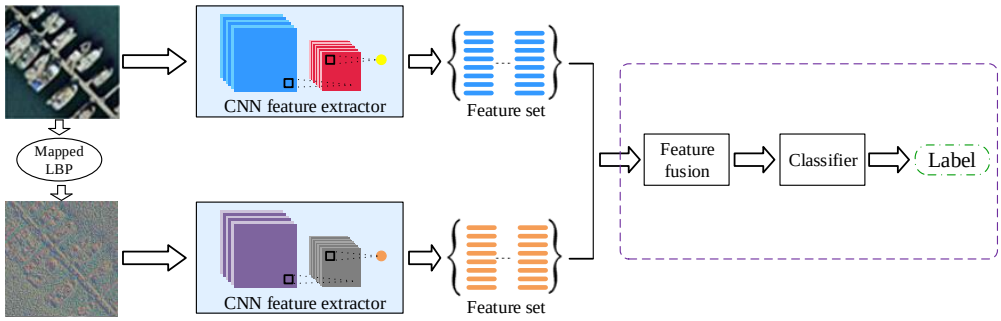


Figure 2. The texture coded two-stream deep architecture. The raw RGB network stream takes RGB images as input. The mapped LBP coded network stream takes mapped LBP images as input. Two differnet sets of features are fused using the two-stream deep feature fusion model.

The GoogLeNet model can get more informative features because of its deeper architecture. Therefore, the GoogLeNet model is applied as feature extractor in our architecture. The schematic view of the GoogLeNet model is depicted in Figure 3. The GoogLeNet model [23], a 22-layer CNN architecture, contains more than 50 convolutional layers distributed in the inception modules. In this work, a pre-trained GoogLeNet model is employed as a feature extractor to extract features from the last pooling layer, which results in a vector of 1024 dimensions for an input image.

3.3.2. Saliency Coded Two-Stream Deep Architecture

Other than texture coded two-stream deep architecture, we also propose a saliency coded two-stream deep architecture using both RGB images and the processed images through saliency detection. Conspicuous information can be obtained through the saliency coded network stream, which is beneficial for accurate classification. We also use pre-trained GoogLeNet as a feature extractor by extracting features from the last pooling layer. Figure 4 shows the saliency coded two-stream deep architecture. In this framework, we use the ImageNet dataset [49] to train a raw RGB network stream and saliency coded network stream separately. Once separately trained, these two streams are combined using a new fusion strategy that is proposed in the next section.

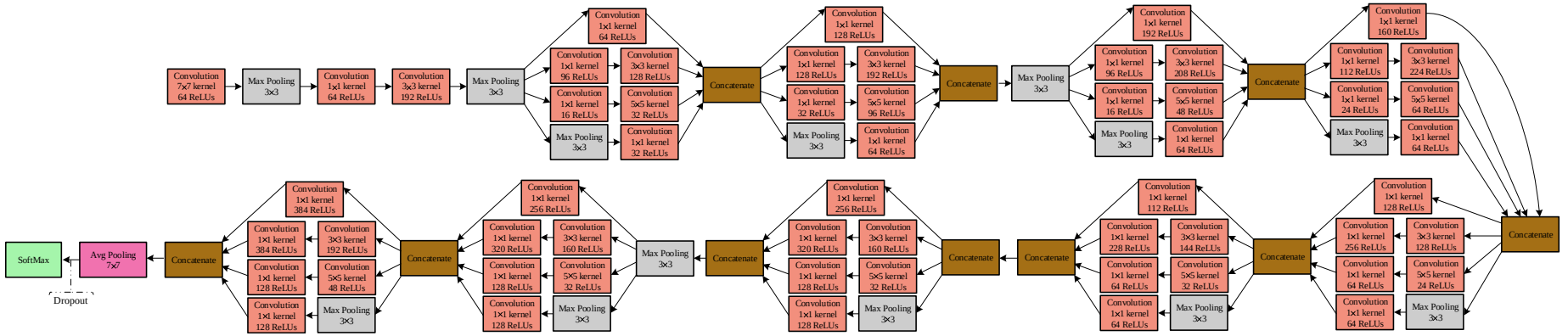


Figure 3. The GoogLeNet architecture used in this paper. In the inception module, the convolutions with different sizes can make the network process the input features at different scales.

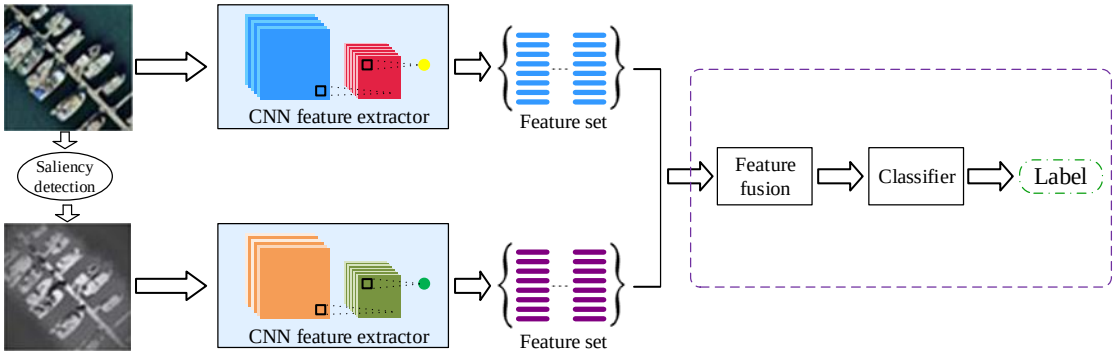


Figure 4. The saliency coded two-stream deep architecture. The raw RGB network stream takes RGB images as input. The saliency coded network stream takes the processed images through saliency detection as input. Two different sets of features are fused using the two-stream deep feature fusion model.

3.4. Two-Stream Deep Feature Fusion Model

Up to now, feature fusion has become a robust and effective way to boost the performance of aerial scene classification. The fused features contain more rich information, which can describe the aerial scene image well. Our proposed two-stream deep feature fusion model is aimed to combine different types of features into a single feature vector with more discriminant information than the input raw features.

Concatenation and addition operations are the most commonly used strategies to fuse the two different types of extracted features. Concatenation operation, also named as serial feature fusion strategy, simply concatenates two different types of features into one single feature vector. It assumes that f_1 and f_2 are the two different types of extracted features with a, b vector dimension, respectively. Then, the fused feature vector is f_{fusion} with $(a+b)$ vector dimension. Addition operation, also termed parallel feature fusion strategy, requires that the two different types of extracted features should have the same dimensionality. The fused feature vector is represented as follows:

$$f_{fusion} = f_1 + if_2, \tag{12}$$

where i is the imaginary unit.

Two-stream deep architecture has the ability to extract two different types of features from the input scene image. Nevertheless, the existing feature fusion approaches have low utilization level of the two sets of extracted features. Hence, a novel two-stream deep feature fusion model is proposed, which is described in Figure 5.

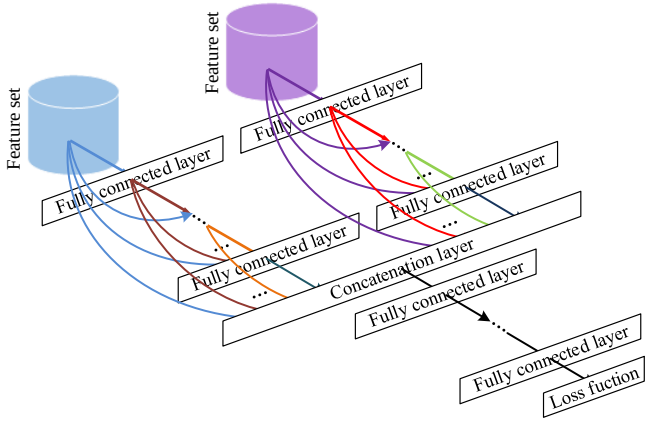


Figure 5. The overall architecture of the two-stream deep feature fusion model.

The idea of this new feature-level fusion strategy is derived from the dense connectivity in a Dense Convolutional Network (DenseNet) [81]. The dense connectivity means the direct connections from any layer to all subsequent layers. Therefore, the current layer takes the feature maps of all preceding layers as its input. The output of the l^{th} layer in Dense Block of DenseNet is defined as follows:

$$y_l = F_l([y_0, y_1, y_2, \ldots, y_{l-2}, y_{l-1}]), \tag{13}$$

where y_l represents the output feature maps of the l^{th} layer, y_0 is a single image, $F_l(\cdot)$ is a composite function which includes three operations: batch normalization (BN) [82], rectified linear units (ReLU) [83], and convolution, $[y_0, y_1, y_2, \ldots, y_{l-2}, y_{l-1}]$ refers to the concatenation of the feature maps produced in layers $0, \ldots, l-1$.

We summarize this idea as basic module of our proposed feature-level fusion strategy in Figure 6. The input of the basic module is the feature vector extracted by the pre-trained GoogLeNet model. The basic module is comprised of a number of fully-connected layers and concatenation layers. The current fully-connected layer takes the output feature vectors of all preceding layers as its input, whose connection type is quite similar to the dense block in DenseNet. The concatenation layer is applied to concatenate the input of the fully-connected layer, which is depicted by arrows gathering in Figure 6. In this basic module, every fully-connected layer is followed by a nonlinear activation function. The output of the l^{th} layer in basic module is defined as follows:

$$O_l = f_l([O_0, O_1, O_2, \ldots, O_{l-2}, O_{l-1}]), \tag{14}$$

where O_l represents the output feature vector of the l^{th} layer, O_0 is the extracted feature vector, $f_l(\cdot)$ is a nonlinear activation function, and $[O_0, O_1, O_2, \ldots, O_{l-2}, O_{l-1}]$ is the concatenation of the output feature vectors of all preceding fully-connected layers.

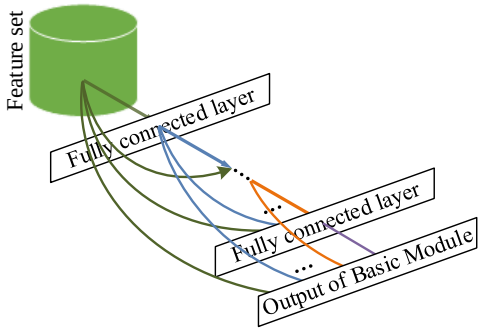


Figure 6. The architecture of the basic module in the two-stream deep feature fusion model.

As shown in Figure 5, the two different feature sets extracted by two streams are fed into two basic modules, respectively. The outputs of the two basic modules are then concatenated together, followed by a series of fully-connected layers. Note that nonlinear activation function is connected to each fully-connected layer.

Our neural network based feature fusion approach can be regarded as a collection of nonlinear high-dimensional projections due to the fully-connected layers. Meanwhile, we borrow the idea of dense block from DenseNet into our basic module, which could encourage feature re-use and strengthen feature propagation. There is one case in which the earlier layer may collapse information to produce short-term features, and the lost information can be regained by using the architecture that has the direct connection to its preceding layer. Such feature fusion approach helps to extract more representative features since we fuse the two different types of features, and this can lead to a better classification performance. Figure 7 shows the detailed network architecture of our proposed feature fusion approach. Note that the ReLU nonlinear activation functions [83] are not presented for brevity.

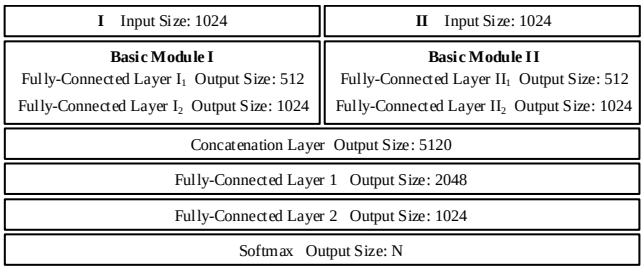


Figure 7. The detailed construction of the two-stream deep feature fusion model.

4. Experimental Design and Results

Here, we evaluate our proposed architectures on a variety of public datasets for aerial scene classification. In the rest of this section, we first give an introduction of the utilized datasets, and then introduce the experimental setup. Finally, we show the experimental results with analysis.

4.1. Description of the Utilized Datasets

4.1.1. UC-Merced Dataset

The first dataset is the well-known UC-Merced dataset [33], which consists of 2100 overhead scene images with 256 × 256 pixels and a spatial resolution of 30 cm per pixel in the RGB color space selected from the United States Geological Survey (USGS) National Map. There are twenty US regions in this USGS National Map: Boston, Birmingham, Buffalo, Columbus, Dallas, Houston, Harrisburg, Jacksonville, Las Vegas, Los Angeles, Miami, Napa, New York, Reno, San Diego, Santa Barbara, Seattle, Tampa, Tucson, and Ventura. These 2100 aerial scene images are equally divided into 21 land use scene classes. Figure 8 shows some example images from the UC-Merced dataset, with one example image per category. In this dataset, some significantly overlapping categories, e.g., medium residential, sparse residential and dense residential which only differ in the density of the structures, make this dataset challenging for classification. More information about this dataset can be obtained at http://vision.ucmerced.edu/datasets.

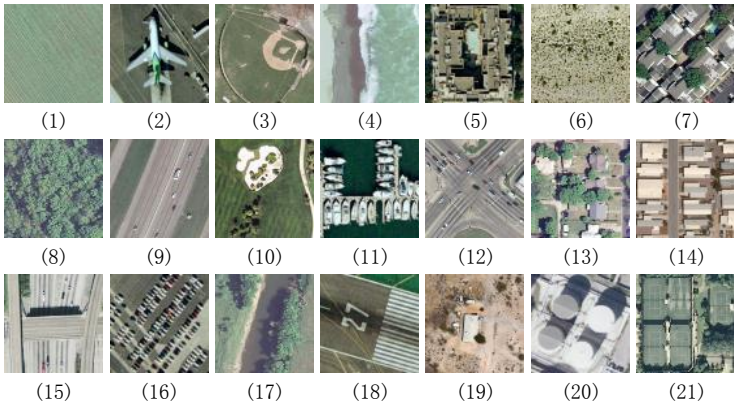


Figure 8. Class representatives of the UC-Merced dataset: (1) Agricultural; (2) Airplane; (3) Baseball diamond; (4) Beach; (5) Buildings; (6) Chaparral; (7) Dense residential; (8) Forest; (9) Freeway; (10) Golf course; (11) Harbor; (12) Intersection; (13) Medium residential; (14) Mobile home park; (15) Overpass; (16) Parking lot; (17) River; (18) Runway; (19) Sparse residential; (20) Storage tanks; (21) Tennis court.

4.1.2. AID Dataset

The AID dataset [51] is acquired from Google Earth using different remote imaging sensors. This dataset contains 10,000 images associated with 30 classes, as shown in Figure 9. For each image, it has 600 × 600 pixels and the spatial resolution ranging from 8 m to half a meter. The number of

images per class varies from 220 to 420; see Table 1 for details. Note that images in this dataset are obtained from different countries, i.e., China, England, France, Germany, etc., and at different seasons. Therefore, the AID has high intra-class variations. More information about this dataset can be obtained at http://www.lmars.whu.edu.cn/xia/AID-project.html.

Figure 9. Class representatives of the Aerial Image Dataset (AID) dataset: (1) Airport; (2) Bare land; (3) Baseball field; (4) Beach; (5) Bridge; (6) Center; (7) Church; (8) Commercial; (9) Dense residential; (10) Desert; (11) Farmland; (12) Forest; (13) Industrial; (14) Meadow; (15) Medium residential; (16) Mountain; (17) Park; (18) Parking; (19) Playground; (20) Pond; (21) Port; (22) Railway station; (23) Resort; (24) River; (25) School; (26) Sparse residential; (27) Square; (28) Stadium; (29) Storage tanks; (30) Viaduct.

Table 1. Scene classes and the number of images per class in the Aerial Image Dataset (AID) dataset.

Name	♯images	Name	♯images	Name	♯images
airport	360	farmland	370	port	380
bare land	310	forest	250	railway station	260
baseball field	220	industrial	390	resort	290
beach	400	meadow	280	river	410
bridge	360	medium residential	290	school	300
center	260	mountain	340	sparse residential	300
church	240	park	350	square	330
commercial	350	parking	390	stadium	290
dense residential	410	playground	370	storage tanks	360
desert	300	pond	420	viaduct	420

4.1.3. NWPU-RESISC45 Dataset

The NWPU-RESISC45 dataset [52] is also a new large-scale dataset which keeps 31,500 aerial images spread over 45 scene classes. Each category contains 700 images with a size of 256 × 256 pixels in the RGB color space and the spatial resolution varying from about 30 m to 0.2 m per pixel. Figure 10 gives some example images from this dataset. The scene images in this dataset have four remarkable characteristics including large scale, rich image variations, high within class diversity and high between class similarity. More information about this dataset can be obtained at http://www.escience.cn/people/JunweiHan/NWPU-RESISC45.html.

Figure 10. Class representatives of the NWPU-RESISC45 dataset: (1) Airplane; (2) Airport; (3) Baseball diamond; (4) Basketball court; (5) Beach; (6) Bridge; (7) Chaparral; (8) Church; (9) Circular farmland; (10) Cloud; (11) Commercial area; (12) Dense residential; (13) Desert; (14) Forest; (15) Freeway; (16) Golf course; (17) Ground track field; (18) Harbor; (19) Industrial area; (20) Intersection; (21) Island; (22) Lake; (23) Meadow; (24) Medium residential; (25) Mobile home park; (26) Mountain; (27) Overpass; (28) Palace; (29) Parking lot; (30) Railway; (31) Railway station; (32) Rectangular farmland; (33) River; (34) Roundabout; (35) Runway; (36) Sea ice; (37) Ship; (38) Snowberg; (39) Sparse residential; (40) Stadium; (41) Storage tank; (42) Tennis court; (43) Terrace; (44) Thermal power station; (45) Wetland.

4.2. Experimental Setup

During the course of our experiments, we use a workstation equipped with a 7th Generation Intel Core i9-7900X processor with 10 M of Cache and up to 4.3 GHz (10 cores and 20 threads), 64 GB DDR4 memory, a graphics processing unit (GPU) NVIDIA GeForce GTX1080Ti with a 11 GB memory, and a SAMSUNG 960 PRO NVME M.2 SSD with 1 TB of capacity.

To evaluate our proposed architectures comprehensively, we adopt two different training–testing ratios for each dataset. For UC-Merced dataset, the ratios of training–testing are 50% vs. 50% and 80% vs. 20%, respectively. For the AID dataset, the ratios are 20% vs. 80% and 50% vs. 50%, respectively. For NWPU-RESISC45 dataset, the ratios are 10% vs. 90% and 20% vs. 80%, respectively.

We employ two widely-used metrics for quantitative evaluation, i.e., overall accuracy and confusion matrix. The overall accuracy is calculated as the number of correct predictions in the testing set divided by the total number of samples in the testing set, which reveals the classification performance on the whole dataset. In the confusion matrix, each row represents the instances in an actual type, while each column represents the instances in a predicted type, which can represent a more detailed analysis than the overall accuracy. Note that, in order to achieve a stable performance and reduce the effect of randomness, we repeat the experiment ten times for each training–testing ratio and we take the mean and standard deviation over the ten times as the final performance. In each repetition, the training samples are picked randomly according to the ratio. Note also that we do not make comparisons with the results from those fine-tuned networks. We believe that this fine-tune operation can improve the performance. However, it is not within the scope of our paper. TensorFlow [84] is chosen as the deep learning framework to implement the proposed method.

4.3. Experimental Results and Analysis

4.3.1. Comparison with the Baseline Methods

In order to comprehensively evaluate our proposed method, we present the baseline comparisons on three publicly available remote sensing scene datasets. In these baseline comparisons, we adopt three different fusion strategies, i.e., concatenation, addition and our proposed fusion approach, to combine the different types of features. With regard to the first two fusion strategies, we use the extreme learning machine (ELM) as a classifier. Here, we employ an ELM classifier rather than a linear support vector machine (SVM) [85] classifier because ELM classifier could achieve better performance, which has been validated by [86]. Tables 2–4 present these baseline comparisons on the UC-Merced dataset (Table 2), AID dataset (Table 3) and NWPU-RESISC45 dataset (Table 4). The best values are marked in bold. TEX-TS-Net and SAL-TS-Net denote texture coded two-stream architecture and saliency coded two-stream architecture, respectively. Based on these results in Tables 2–4, our observations are listed as follows.

Table 2. Overall accuracies and standard deviations (%) of different feature fusion methods on the UC-Merced dataset.

Method	Training Ratios (%)	
	50%	80%
TEX-TS-Net (concatenation)	94.31 ± 0.31	95.71 ± 0.66
SAL-TS-Net (concatenation)	94.46 ± 0.60	96.17 ± 0.90
TEX-TS-Net (addition)	95.25 ± 0.54	96.62 ± 0.63
SAL-TS-Net (addition)	95.41 ± 0.58	97.12 ± 0.96
TEX-TS-Net (our feature fusion model)	97.55 ± 0.46	98.40 ± 0.76
SAL-TS-Net (our feature fusion model)	**97.79 ± 0.56**	**98.90 ± 0.95**

* TEX-TS-Net and SAL-TS-Net denote texture coded two-stream architecture and saliency coded two-stream architecture, respectively.

Table 3. Overall accuracies and standard deviations (%) of different feature fusion methods on the AID dataset.

Method	Training Ratios (%)	
	20%	50%
TEX-TS-Net (concatenation)	88.46 ± 0.23	90.15 ± 0.18
SAL-TS-Net (concatenation)	89.15 ± 0.45	91.25 ± 0.59
TEX-TS-Net (addition)	88.56 ± 0.25	90.29 ± 0.19
SAL-TS-Net (addition)	89.21 ± 0.39	91.31 ± 0.49
TEX-TS-Net (our feature fusion model)	93.31 ± 0.11	95.17 ± 0.21
SAL-TS-Net (our feature fusion model)	**94.09 ± 0.34**	**95.99 ± 0.35**

Table 4. Overall accuracies and standard deviations (%) of different feature fusion methods on the NWPU-RESISC45 dataset.

Method	Training Ratios (%)	
	10%	20%
TEX-TS-Net (concatenation)	79.58 ± 0.33	81.13 ± 0.21
SAL-TS-Net (concatenation)	79.69 ± 0.47	81.46 ± 0.22
TEX-TS-Net (addition)	79.63 ± 0.30	81.22 ± 0.27
SAL-TS-Net (addition)	79.75 ± 0.41	81.52 ± 0.28
TEX-TS-Net (our feature fusion model)	84.77 ± 0.24	86.36 ± 0.19
SAL-TS-Net (our feature fusion model)	**85.02 ± 0.25**	**87.01 ± 0.19**

(1) For the concatenation fusion strategy, the classification accuracies of saliency coded two-stream architecture achieve 94.46%, 96.17%, 89.15%, 91.25%, 79.69%, and 81.46% on the UC-Merced dataset (50% and 80% training samples), the AID dataset (20% and 50% training samples), and the NWPU-RESISC45 dataset (10% and 20% training samples), respectively, which are higher than the results of texture coded two-stream architecture, i.e., 94.31%, 95.71%, 88.46%, 90.15%, 79.58%, and 81.13%. At the same time, the saliency coded two-stream architecture also performs better than texture coded two-stream architecture with regard to the other two fusion strategies. The reason is mainly that the method of saliency detection has the ability of focusing more attention on the image regions that are most informative and dominate the class. Therefore, the features extracted from the saliency coded network stream are more informative and significant than that from the mapped LBP coded network stream.

(2) For the texture coded two-stream architecture, the classification accuracies of using our proposed feature fusion strategy can rank 97.55%, 98.40%, 93.31%, 95.17%, 84.77%, and 86.36% on the aforementioned datasets, respectively, which are higher than 95.25%, 96.62%, 88.56%, 90.29%, 79.63%, and 81.22%, obtained by applying the addition fusion strategy. The method with the concatenation fusion strategy has the lowest classification accuracies, i.e., 94.31%, 95.71%, 88.46%, 90.15%, 79.58%, and 81.13%. Meanwhile, we can get the same comparison results in the saliency coded two-stream architecture, in which the concatenation fusion strategy takes the third place, the addition fusion strategy takes the second place, and our proposed fusion model has the highest accuracies. It can be seen from the comparison results that our proposed deep feature fusion model can enhance the representational ability of the extracted features and improve the classification performance.

(3) The saliency coded two-stream deep architecture with our proposed feature fusion model outperforms all the other baseline approaches over these three utilized datasets, having the highest accuracies.

(4) The overall accuracies on the UC-Merced dataset are almost saturated. This is because this dataset is not very rich in terms of image variations. The AID dataset and the NWPU-RESISC45 dataset have higher intra-class variations and smaller inter-class dissimilarity; therefore, the results on them are still more challenging.

4.3.2. Confusion Analysis

In addition, we also report the confusion matrix and detail classification accuracy for each scene label. For the limited article space, we only present the confusion matrixes for the UC-Merced dataset (80% training samples), the AID dataset (50% training samples), and the NWPU-RESISC45 dataset (20% training samples) using saliency coded two-stream architecture with our proposed fusion model and texture coded two-stream architecture with our proposed fusion model, which are the best and the second best methods on these three utilized datasets. Figures 11–13 present these confusion matrixes. Based on these confusion matrixes, our observations are listed as follows.

(1) Over the UC-Merced dataset (see Figure 11), most of the scene categories can achieve the classification accuracy close to or even equal to 1. In the confusion matrix of our feature fusion model based texture coded two-stream architecture, categories with classification accuracy lower than 1 include agriculture (0.95), dense residential (0.95), intersection (0.95), medium residential (0.95), river (0.9) and tennis court (0.95). In the confusion matrix of our feature fusion model based saliency coded two-stream architecture, categories with classification accuracy lower than 1 include buildings (0.95), dense residential (0.9), golf course (0.95) and tennis court (0.95). The scene categories in the confusion matrix of saliency coded two-stream architecture with our proposed fusion model obtain a better performance compared to the confusion matrix of the other method. For example, the agriculture, intersection, medium residential, and river scenes, which are confused in texture coded two-stream architecture with our proposed fusion model, are fully recognized by saliency coded two-stream architecture with our proposed fusion model.

(2) Over the AID dataset (see Figure 12), it should be noted that our methods can get the classification accuracy rate of more than 0.95 under most scene categories, including bare land (a: 0.96, b: 0.96), baseball field (a: 1, b: 0.99), beach (a: 0.98, b: 1), bridge (a: 0.97, b: 0.98), desert (a: 0.97, b: 0.97), farmland (a: 0.96, b: 0.98), forest (a: 0.99, b: 0.99), meadow (a 0.99, b: 0.99), mountain (a: 0.99, b: 0.99), parking (a: 0.99, b: 1), playground (a: 0.98, b: 0.98), pond (a: 0.98, b: 0.99), sparse residential (a: 1, b: 1), port (a: 0.97, b: 0.98), river (a: 0.96, b: 0.99), stadium (a: 0.97, b: 0.99), storage tanks (a: 0.98, b: 0.96), and viaduct (a: 0.99, b: 1). At the same time, our methods can obtain relatively good performance under some scene categories that are difficult to recognize. For instance, from the results obtained by using the multilevel fusion method [66], we can see the scene categories with low accuracy, i.e., school (0.77), square (0.80), resort (0.74), and center (0.80). In Figure 12a, the classification accuracy rates of these four scene categories are improved by our method, i.e., school (0.83), square (0.82), resort (0.81), and center (0.85). In Figure 12b, the results about these four scene classes are listed as follows: school (0.83), square (0.85), resort (0.80), and center (0.89).

(3) Over the NWPU-RESISC45 dataset (see Figure 13), compared with the confusion matrix of BoCF (VGGNet-16) [57], our approaches provide consistent improvement in performance on most scene categories, i.e., commercial area (BoCF (VGGNet-16): 0.68, a: 0.76, b: 0.76), freeway (BoCF (VGGNet-16): 0.69, a: 0.73, b: 0.81), tennis court (BoCF (VGGNet-16): 0.57, a: 0.72, b: 0.74), palace (BoCF (VGGNet-16): 0.46, a: 0.61, b: 0.53), etc.

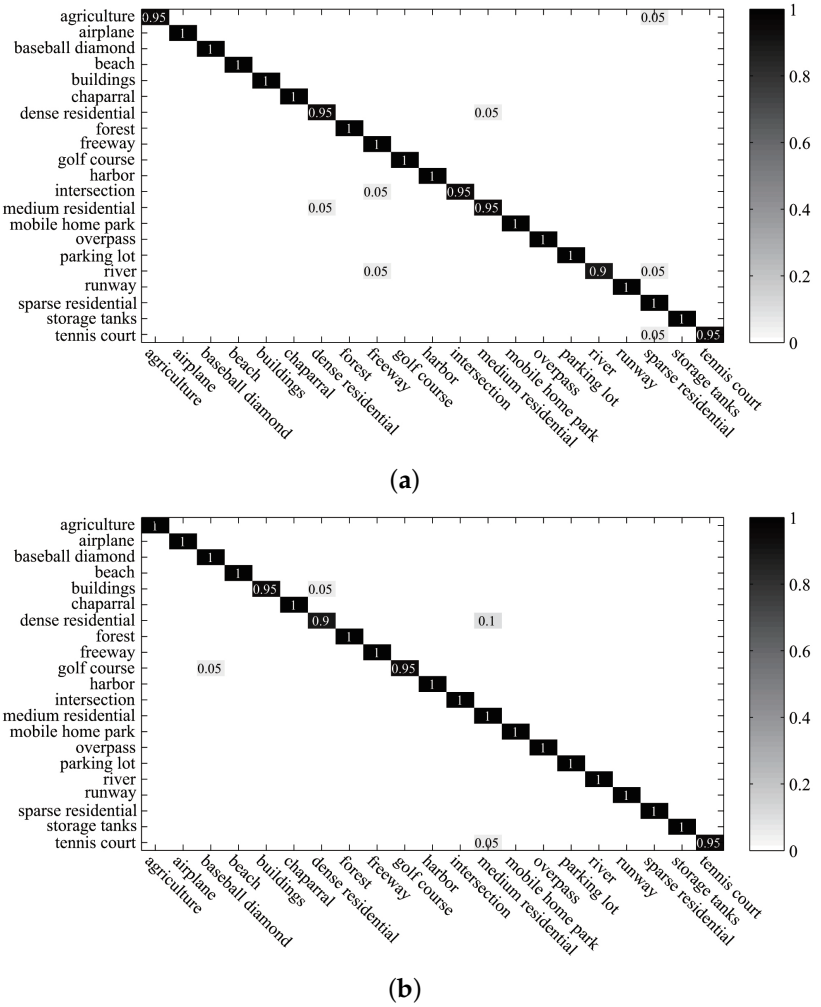
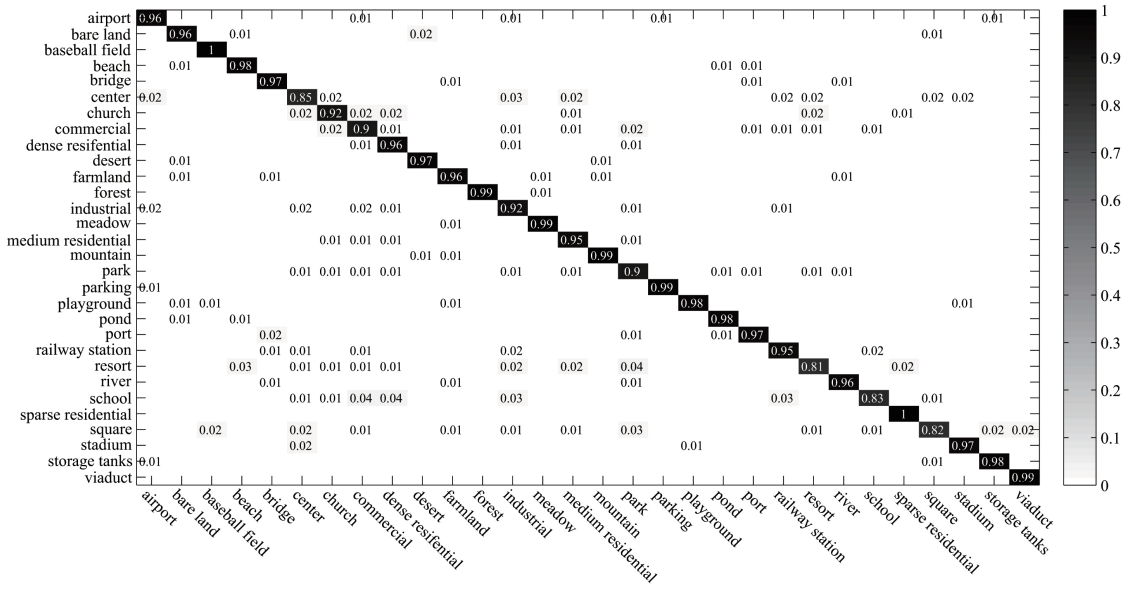
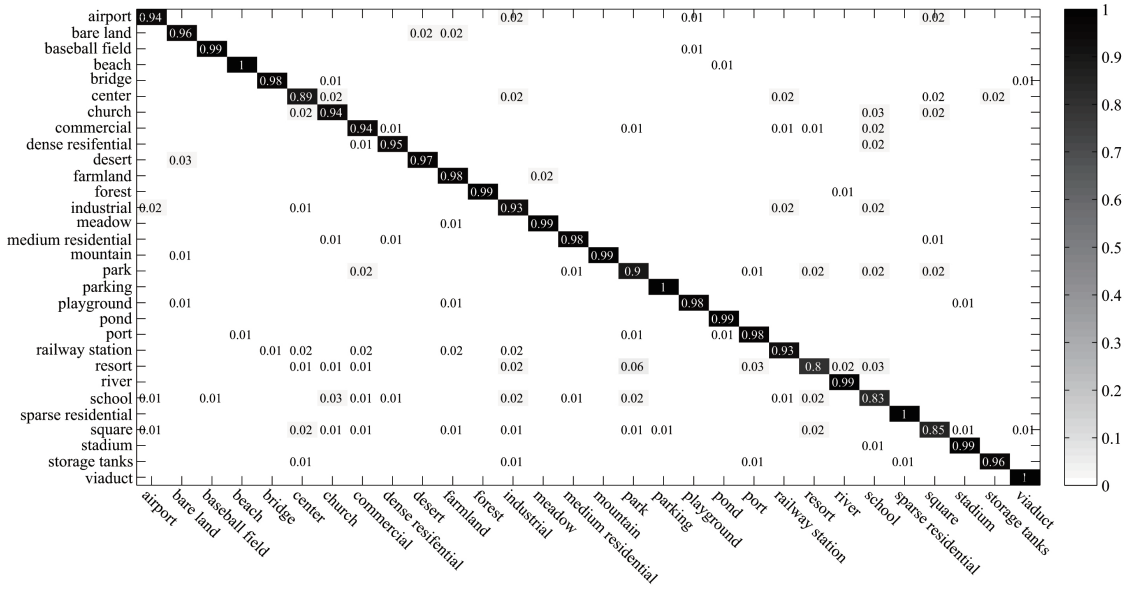


Figure 11. Confusion matrixes of the UC-Merced dataset under the training ratio of 80% using the following methods. (**a**) texture coded two-stream architecture with our proposed fusion model; (**b**) saliency coded two-stream architecture with our proposed fusion model.

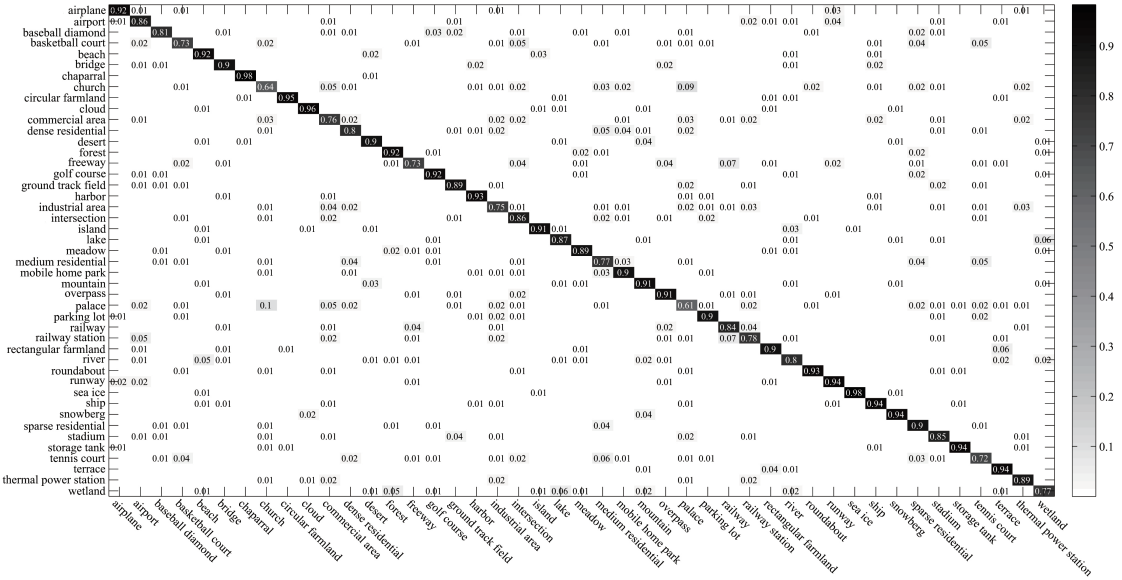


(**a**)

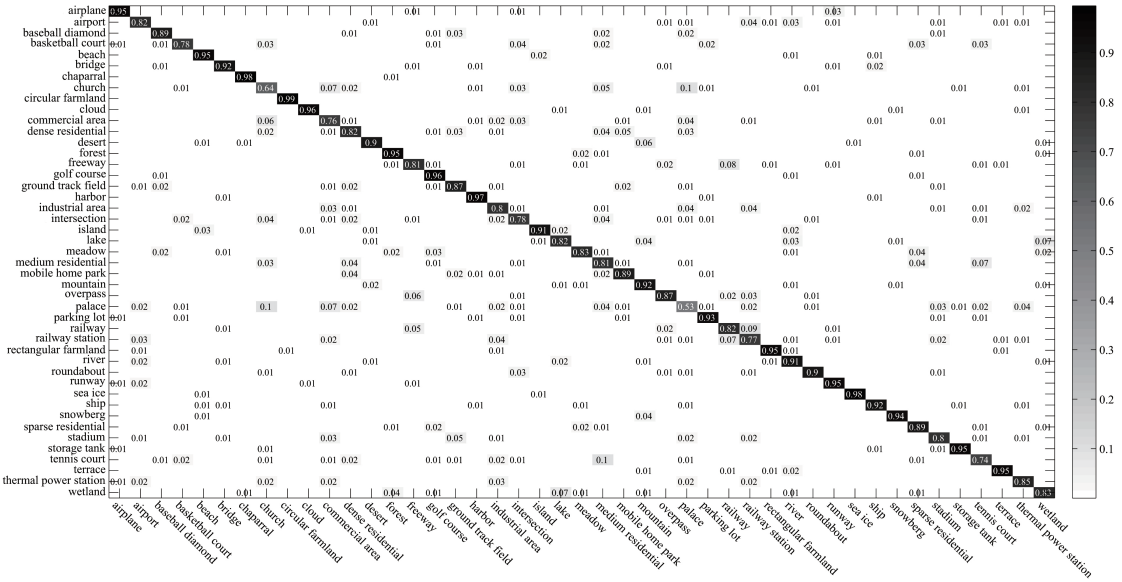


(**b**)

Figure 12. Confusion matrixes of the AID dataset under the training ratio of 50% using the following methods. (**a**) texture coded two-stream architecture with our proposed fusion model; (**b**) saliency coded two-stream architecture with our proposed fusion model.



(**a**)



(**b**)

Figure 13. Confusion matrixes of the NWPU-RESISC45 dataset under the training ratio of 20% using the following methods. (**a**) texture coded two-stream architecture with our proposed fusion model; (**b**) saliency coded two-stream architecture with our proposed fusion model.

5. Discussion

Extensive experiments present that our feature fusion model based saliency coded two-stream architecture, which integrates raw RGB network stream and saliency coded network stream by using a dense connectivity based feature fusion network, is very effective for representing the aerial scene images.

For the sake of comprehensive evaluation, we compare our best numerical results with the most recent results obtained from the state-of-the-art approaches. Tables 5–7 present the comparisons with the state-of-the-art methods on the UC-Merced dataset, the AID dataset, and the NWPU-RESISC45 dataset. The best values are marked in bold. As can be seen in Tables 5–7, our feature fusion model based saliency coded two-stream architecture can achieve remarkable classification results when

compared with the state-of-the-art methods. The detailed comparison and analysis on each dataset are presented as follows.

Table 5. Comparison of classification accuracy with the state-of-the-art methods on the UC-Merced dataset.

Method	Training Ratios (%)	
	50%	80%
SCK [33]	-	72.52
SPCK [34]	-	73.14
BoVW [53]	-	76.81
BoVW + SCK [33]	-	77.80
SIFT + SC [87]	-	81.67 ± 1.23
SSEA [88]	-	82.72 ± 1.18
MCMI [89]	-	88.20
OverFeat [54]	-	90.91 ± 1.19
VLAD [90]	-	92.50
VLAT [90]	-	94.30
MS-CLBP + FV [91]	88.76 ± 0.79	93.00 ± 1.20
GoogLeNet [51]	92.70 ± 0.60	94.31 ± 0.89
CaffeNet [51]	93.98 ± 0.67	95.02 ± 0.81
VGG-VD-16 [51]	94.14 ± 0.69	95.21 ± 1.20
Bidirectional adaptive feature fusion [92]	-	95.48
CNN-ELM [86]	-	95.62
salM^{3}LBP − CLM [93]	94.21 ± 0.75	95.75 ± 0.80
Appearance-based [56]	-	96.05 ± 0.62
TEX-Net-LF [61]	95.89 ± 0.37	96.62 ± 0.49
CaffeNet with DCF [60]	95.26 ± 0.50	96.79 ± 0.66
MDDC [56]	-	96.92 ± 0.57
VGG-VD16 with DCF [60]	95.42 ± 0.71	97.10 ± 0.85
DRB Ensemble [94]	-	97.10
LASC-CNN (single-scale) [58]	-	97.14
Aggregate strategy 1 [59]	95.84	97.28
Aggregate strategy 2 [59]	96.25	97.40
Fusion by addition [62]	-	97.42 ± 1.79
LASC-CNN (multiscale) [58]	-	98.10
SHHTFM [95]	-	98.33 ± 0.98
Ours	**97.79 ± 0.56**	**98.90 ± 0.95**

Table 6. Comparison of classification accuracy with the state-of-the-art methods on the AID dataset.

Method	Training Ratios (%)	
	20%	50%
BoVW [93]	-	78.66 ± 0.52
MS-CLBP+FV [93]	-	86.48 ± 0.27
GoogLeNet [51]	83.44 ± 0.40	86.39 ± 0.55
VGG-VD-16 [51]	86.59 ± 0.29	89.64 ± 0.36
CaffeNet [51]	86.86 ± 0.47	89.53 ± 0.31
DCA with concatenation [62]	-	89.71 ± 0.33
salM^{3}LBP − CLM [93]	86.92 ± 0.35	89.76 ± 0.45
Fusion by concatenation [62]	-	91.86 ± 0.28
Fusion by addition [62]	-	91.87 ± 0.36
TEX-Net-LF [61]	90.87 ± 0.11	92.96 ± 0.18
Bidirectional adaptive feature fusion [92]	-	93.56
Multilevel fusion [66]	-	94.17 ± 0.32
Ours	**94.09 ± 0.34**	**95.99 ± 0.35**

(1) On the UC-Merced dataset (see Table 5), our best architecture outperforms all the other aerial scene classification approaches with an increase in overall accuracy of 1.54%, 0.57% over the second best methods, Aggregate strategy 2 [59] and SHHTFM [95], using 50% and 80% training ratios, respectively.

(2) On the AID dataset (see Table 6), our best architecture performs better than the state-of-the-art approaches and the margins are generally rather large. Our accuracies are higher than the second best results, i.e., TEX-Net-LF [61] and Multilevel fusion [66], by 3.22% and 1.82%, under the training ratios of 20% and 50%, respectively.

(3) On the NWPU-RESISC45 dataset (see Table 7), our best architecture remarkably improves the performance when compared with the state-of-the-art results. Specifically, our method gains a large margin of overall accuracy improvements with 2.37%, 2.69% over the second best method, BoCF (VGGNet-16) [57], using 10% and 20% labeled samples per class as training ratio, respectively.

In this paper, the strengths of our work are listed as follows. Firstly, we apply different auxiliary models, i.e., the mapped LBP coded CNN model and the saliency coded CNN model, to provide complementary information to the conventional RGB based CNN model. This operation can provide more evidence to the classification process. Secondly, the proposed two-stream deep feature fusion model combines different types of features based on dense connectivity, which can achieve a higher utilization level of the extracted features and produce more informative representations for the input scene images. Additionally, our architectures have general applicability, which could be used for solving other tasks, such as traffic-sign recognition and face recognition.

Table 7. Comparison of classification accuracy with the state-of-the-art methods on the NWPU-RESISC45 dataset.

Method	Training Ratios (%)	
	10%	20%
Color histograms [52]	24.84 ± 0.22	27.52 ± 0.14
BoVW [52]	41.72 ± 0.21	44.97 ± 0.28
GoogLeNet [52]	76.19 ± 0.38	78.48 ± 0.26
VGGNet-16 [52]	76.47 ± 0.18	79.79 ± 0.15
AlexNet [52]	76.69 ± 0.21	79.85 ± 0.13
BoCF (AlexNet) [57]	55.22 ± 0.39	59.22 ± 0.18
BoCF (GoogLeNet) [57]	78.92 ± 0.17	80.97 ± 0.17
LASC-CNN (single-scale) [58]	80.69	83.64
LASC-CNN (multiscale) [58]	81.37	84.30
BoCF (VGGNet-16) [57]	82.65 ± 0.31	84.32 ± 0.17
Ours	$\mathbf{85.02 \pm 0.25}$	$\mathbf{87.01 \pm 0.19}$

6. Conclusions

In this paper, we construct two novel two-stream deep architectures based on the idea of feature-level fusion. The first architecture, a texture coded two-stream deep architecture, uses the mapped LBP coded network stream as the auxiliary stream and we fuses it with the raw RGB network stream using a novel deep feature fusion model. The two-stream deep feature fusion model adopts the connection type of dense block in DenseNet. The second architecture, saliency coded two-stream deep architecture, is very similar to the texture coded two-stream deep architecture except that the auxiliary stream is the saliency coded network stream. Finally, we conduct extensive experiments on three publicly available remote sensing scene datasets. The experimental results demonstrate that the two-stream deep architectures with our feature fusion model yield better classification performance

against state-of-the-art methods. Our best architecture, our feature fusion model based saliency coded two-stream architecture, can achieve 97.79%, 98.90%, 94.09%, 95.99%, 85.02%, and 87.01% on the UC-Merced dataset (50% and 80% training samples), the AID dataset (20% and 50% training samples), and the NWPU-RESISC45 dataset (10% and 20% training samples), respectively. Inspired by our current promising results on aerial scene classification task, in the future, we plan to extend our architectures to other applications. Meanwhile, we believe that the current performance could be further improved by exploring better auxiliary features or feature fusion strategies.

Author Contributions: Y.Y. and F.L. conceived and designed the methods; Y.Y. conducted the experiments and analyzed the results; Y.Y. and F.L. wrote this paper.

Funding: This research was funded by the National Natural Science Foundation of China under Grant No. 71771216, and Grant No. 71701209.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Faisal, A.; Kafy, A.; Roy, S. Integration of Remote Sensing and GIS Techniques for Flood Monitoring and Damage Assessment: A Case Study of Naogaon District. *Bangladesh J. Remote Sens. GIS* **2018**, *7*, 2. [CrossRef]
2. Bi, S.; Lin, X.; Wu, Z.; Yang, S. Development technology of principle prototype of high-resolution quantum remote sensing imaging. In *Quantum Sensing and Nano Electronics and Photonics XV*; International Society for Optics and Photonics: Bellingham, WA, USA, 2018, Volume 10540, p. 105400Q.
3. Weng, Q.; Quattrochi, D.; Gamba, P.E. *Urban Remote Sensing*; CRC Press: Boca Raton, FL, USA, 2018.
4. Mukherjee, A.B.; Krishna, A.P.; Patel, N. Application of Remote Sensing Technology, GIS and AHP-TOPSIS Model to Quantify Urban Landscape Vulnerability to Land Use Transformation. In *Information and Communication Technology for Sustainable Development*; Springer: Singapore, 2018; pp. 31–40.
5. Yang, Y.; Newsam, S. Geographic image retrieval using local invariant features. *IEEE Trans. Geosci. Remote Sens.* **2013**, *51*, 818–832. [CrossRef]
6. Zheng, X.; Sun, X.; Fu, K.; Wang, H. Automatic annotation of satellite images via multifeature joint sparse coding with spatial relation constraint. *IEEE Geosci. Remote Sens. Lett.* **2013**, *10*, 652–656. [CrossRef]
7. Hu, J.; Xia, G.S.; Hu, F.; Zhang, L. A comparative study of sampling analysis in the scene classification of optical high-spatial resolution remote sensing imagery. *Remote Sens.* **2015**, *7*, 14988–15013. [CrossRef]
8. Ammour, N.; Bashmal, L.; Bazi, Y.; Al Rahhal, M.; Zuair, M. Asymmetric Adaptation of Deep Features for Cross-Domain Classification in Remote Sensing Imagery. *IEEE Geosci. Remote Sens. Lett.* **2018**, *15*, 597–601. [CrossRef]
9. Alhichri, H.; Othman, E.; Zuair, M.; Ammour, N.; Bazi, Y. Tile-Based Semisupervised Classification of Large-Scale VHR Remote Sensing Images. *J. Sens.* **2018**, *2018*. [CrossRef]
10. Banerjee, B.; Chaudhuri, S. Scene Recognition From Optical Remote Sensing Images Using Mid-Level Deep Feature Mining. *IEEE Geosci. Remote Sens. Lett.* **2018**, *15*. [CrossRef]
11. Minetto, R.; Segundo, M.P.; Sarkar, S. Hydra: An Ensemble of Convolutional Neural Networks for Geospatial Land Classification. *arXiv* **2018**, arXiv:1802.03518.
12. Yang, Y.; Newsam, S. Comparing SIFT descriptors and Gabor texture features for classification of remote sensed imagery. In Proceedings of the 15th IEEE International Conference on Image Processing (ICIP 2008), San Diego, CA, USA, 12–15 October 2008; pp. 1852–1855.
13. Dos Santos, J.A.; Penatti, O.A.B.; da Silva Torres, R. Evaluating the Potential of Texture and Color Descriptors for Remote Sensing Image Retrieval and Classification. In Proceedings of the Fifth International Conference on Computer Vision Theory and Applications, Angers, France, 17–21 May 2010; Volume 2, pp. 203–208.
14. Zhao, L.; Tang, P.; Huo, L. A 2-D wavelet decomposition-based bag-of-visual-words model for land-use scene classification. *Int. J. Remote Sens.* **2014**, *35*, 2296–2310.
15. Chen, S.; Tian, Y. Pyramid of spatial relatons for scene-level land use classification. *IEEE Trans. Geosci. Remote Sens.* **2015**, *53*, 1947–1957. [CrossRef]
16. Hu, F.; Xia, G.S.; Hu, J.; Zhang, L. Transferring deep convolutional neural networks for the scene classification of high-resolution remote sensing imagery. *Remote Sens.* **2015**, *7*, 14680–14707. [CrossRef]

17. Luus, F.P.; Salmon, B.P.; Van den Bergh, F.; Maharaj, B.T.J. Multiview deep learning for land-use classification. *IEEE Geosci. Remote Sens. Lett.* **2015**, *12*, 2448–2452. [CrossRef]
18. Chen, J.; Wang, C.; Ma, Z.; Chen, J.; He, D.; Ackland, S. Remote Sensing Scene Classification Based on Convolutional Neural Networks Pre-Trained Using Attention-Guided Sparse Filters. *Remote Sens.* **2018**, *10*, 290. [CrossRef]
19. Chew, R.F.; Amer, S.; Jones, K.; Unangst, J.; Cajka, J.; Allpress, J.; Bruhn, M. Residential scene classification for gridded population sampling in developing countries using deep convolutional neural networks on satellite imagery. *Int. J. Health Geogr.* **2018**, *17*, 12. [CrossRef] [PubMed]
20. Jia, Y.; Shelhamer, E.; Donahue, J.; Karayev, S.; Long, J.; Girshick, R.; Guadarrama, S.; Darrell, T. Caffe: Convolutional architecture for fast feature embedding. In Proceedings of the 22nd ACM International Conference on Multimedia, Orlando, FL, USA, 3–7 November 2014; pp. 675–678.
21. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. *Imagenet Classification with Deep Convolutional Neural Networks*; Neural Information Processing Systems Foundation, Inc.: La Jolla, CA, USA, 2012; pp. 1097–1105.
22. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**, arXiv:1409.1556.
23. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going deeper with convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'15), Boston, MA, USA, 7–12 June 2015.
24. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'16), Las Vegas Valley, NV, USA, 26 June–1 July 2016; pp. 770–778.
25. Zhu, X.X.; Tuia, D.; Mou, L.; Xia, G.S.; Zhang, L.; Xu, F.; Fraundorfer, F. Deep Learning in Remote Sensing: A Comprehensive Review and List of Resources. *IEEE Geosci. Remote Sens. Mag.* **2017**, *5*, 8–36. [CrossRef]
26. Ball, J.E.; Anderson, D.T.; Chan, C.S. Comprehensive survey of deep learning in remote sensing: Theories, tools, and challenges for the community. *J. Appl. Remote Sens.* **2017**, *11*, 042609. [CrossRef]
27. Li, Y.; Zhang, H.; Xue, X.; Jiang, Y.; Shen, Q. Deep learning for remote sensing image classification: A survey. In *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*; John Wiley & Sons: Hoboken, NJ, USA, 2018; p. e1264.
28. Ojala, T.; Pietikainen, M.; Maenpaa, T. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Trans. Pattern Anal. Mach. Intell.* **2002**, *24*, 971–987. [CrossRef]
29. Lowe, D.G. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vis.* **2004**, *60*, 91–110. [CrossRef]
30. Swain, M.J.; Ballard, D.H. Color indexing. *Int. J. Comput. Vis.* **1991**, *7*, 11–32. [CrossRef]
31. Oliva, A.; Torralba, A. Modeling the shape of the scene: A holistic representation of the spatial envelope. *Int. J. Comput. Vis.* **2001**, *42*, 145–175. [CrossRef]
32. Luo, B.; Jiang, S.; Zhang, L. Indexing of remote sensing images with different resolutions by multiple features. *IEEE J. Sel. Top. Appl. Earth Observ. Remote Sens.* **2013**, *6*, 1899–1912. [CrossRef]
33. Yang, Y.; Newsam, S. 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, San Jose, CA, USA, 2–5 November 2010; pp. 270–279.
34. Yang, Y.; Newsam, S. Spatial pyramid co-occurrence for image classification. In Proceedings of the 2011 IEEE International Conference on Computer Vision, Barcelona, Spain, 6–13 November 2011; pp. 1465–1472.
35. Shao, W.; Yang, W.; Xia, G.S.; Liu, G. A hierarchical scheme of multiple feature fusion for high-resolution satellite scene categorization. In *Computer Vision Systems, Proceedings of the 9th International Conference, ICVS 2013, St. Petersburg, Russia, 16–18 July 2013*; Springer: Berlin/Heidelberg, Germany, 2013; pp. 324–333.
36. Zhao, L.J.; Tang, P.; Huo, L.Z. Land-use scene classification using a concentric circle-structured multiscale bag-of-visual-words model. *IEEE J. Sel. Top. Appl. Earth Observ. Remote Sens.* **2014**, *7*, 4620–4631. [CrossRef]
37. Sridharan, H.; Cheriyadat, A. Bag of lines (BoL) for improved aerial scene representation. *IEEE Geosci. Remote Sens. Lett.* **2015**, *12*, 676–680. [CrossRef]
38. Hu, J.; Jiang, T.; Tong, X.; Xia, G.S.; Zhang, L. A benchmark for scene classification of high spatial resolution remote sensing imagery. In Proceedings of the 2015 IEEE International Geoscience and Remote Sensing Symposium (IGARSS), Milan, Italy, 26–31 July 2015; pp. 5003–5006.

39. Lazebnik, S.; Schmid, C.; Ponce, J. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06), New York, NY, USA, 17–22 June 2006; Volume 2, pp. 2169–2178.
40. Wang, J.; Yang, J.; Yu, K.; Lv, F.; Huang, T.; Gong, Y. Locality-constrained linear coding for image classification. In Proceedings of the 2010 IEEE Conference on Computer Vision and Pattern Recognition (CVPR'10), San Francisco, CA, USA, 13–18 June 2010; pp. 3360–3367.
41. Bosch, A.; Zisserman, A.; Muñoz, X. Scene classification via pLSA. In *Computer Vision—ECCV 2006: Proceeding of the 9th European Conference on Computer Vision, Graz, Austria, 7–13 May 2006*; Springer: Berlin/Heidelberg, Germany, 2006; pp. 517–530.
42. Blei, D.M.; Ng, A.Y.; Jordan, M.I. Latent dirichlet allocation. *J. Mach. Learn. Res.* **2003**, *3*, 993–1022.
43. Perronnin, F.; Sánchez, J.; Mensink, T. Improving the fisher kernel for large-scale image classification. In *Computer Vision—ECCV 2006, Proceeding of the 11th European Conference on Computer Vision, Heraklion, Crete, Greece, 5–11 September 2010*; Springer: Berlin/Heidelberg, Germany, 2010; pp. 143–156.
44. Jegou, H.; Perronnin, F.; Douze, M.; Sánchez, J.; Perez, P.; Schmid, C. Aggregating local image descriptors into compact codes. *IEEE Trans. Pattern Anal. Mach. Intell.* **2012**, *34*, 1704–1716. [CrossRef] [PubMed]
45. Penatti, O.A.; Nogueira, K.; dos Santos, J.A. Do deep features generalize from everyday objects to remote sensing and aerial scenes domains? In Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Boston, MA, USA, 7 June 2015; pp. 44–51.
46. Zhang, F.; Du, B.; Zhang, L. Scene classification via a gradient boosting random convolutional network framework. *IEEE Trans. Geosci. Remote Sens.* **2016**, *54*, 1793–1802. [CrossRef]
47. Aryal, J.; Dutta, R. Smart city and geospatiality: Hobart deeply learned. In Proceeding of the 2015 31st IEEE International Conference on Data Engineering Workshops (ICDEW), Seoul, Korea, 13–17 April 2015; pp. 108–109.
48. Dutta, R.; Aryal, J.; Das, A.; Kirkpatrick, J.B. Deep cognitive imaging systems enable estimation of continental-scale fire incidence from climate data. *Sci. Rep.* **2013**, *3*, 3188. [CrossRef] [PubMed]
49. Russakovsky, O.; Deng, J.; Su, H.; Krause, J.; Satheesh, S.; Ma, S.; Huang, Z.; Karpathy, A.; Khosla, A.; Bernstein, M.; et al. Imagenet large scale visual recognition challenge. *Int. J. Comput. Vis.* **2015**, *115*, 211–252. [CrossRef]
50. Sermanet, P.; Eigen, D.; Zhang, X.; Mathieu, M.; Fergus, R.; LeCun, Y. Overfeat: Integrated recognition, localization and detection using convolutional networks. *arXiv* **2013**, arXiv:1312.6229.
51. Xia, G.S.; Hu, J.; Hu, F.; Shi, B.; Bai, X.; Zhong, Y.; Zhang, L.; Lu, X. AID: A benchmark data set for performance evaluation of aerial scene classification. *IEEE Trans. Geosci. Remote Sens.* **2017**, *55*, 3965–3981. [CrossRef]
52. Cheng, G.; Han, J.; Lu, X. Remote sensing image scene classification: benchmark and state of the art. *Proc. IEEE* **2017**, *105*, 1865–1883. [CrossRef]
53. Castelluccio, M.; Poggi, G.; Sansone, C.; Verdoliva, L. Land use classification in remote sensing images by convolutional neural networks. *arXiv* **2015**, arXiv:1508.00092.
54. Nogueira, K.; Penatti, O.A.; dos Santos, J.A. Towards better exploiting convolutional neural networks for remote sensing scene classification. *Pattern Recognit.* **2017**, *61*, 539–556. [CrossRef]
55. Liu, Y.; Huang, C. Scene classification via triplet networks. *IEEE J. Sel. Top. Appl. Earth Observ. Remote Sens.* **2018**, *11*, 220–237. [CrossRef]
56. Qi, K.; Yang, C.; Guan, Q.; Wu, H.; Gong, J. A Multiscale Deeply Described Correlatons-Based Model for Land-Use Scene Classification. *Remote Sens.* **2017**, *9*, 917. [CrossRef]
57. Cheng, G.; Li, Z.; Yao, X.; Guo, L.; Wei, Z. Remote sensing image scene classification using bag of convolutional features. *IEEE Geosci. Remote Sens. Lett.* **2017**, *14*, 1735–1739. [CrossRef]
58. Yuan, B.; Li, S.; Li, N. Multiscale deep features learning for land-use scene recognition. *J. Appl. Remote Sens.* **2018**, *12*, 015010. [CrossRef]
59. Liu, N.; Wan, L.; Zhang, Y.; Zhou, T.; Huo, H.; Fang, T. Exploiting Convolutional Neural Networks with Deeply Local Description for Remote Sensing Image Classification. *IEEE Access* **2018**, *6*, 11215–11228. [CrossRef]
60. Liu, N.; Lu, X.; Wan, L.; Huo, H.; Fang, T. Improving the Separability of Deep Features with Discriminative Convolution Filters for RSI Classification. *ISPRS Int. J. Geo-Inf.* **2018**, *7*, 95. [CrossRef]

61. Anwer, R.M.; Khan, F.S.; van de Weijer, J.; Molinier, M.; Laaksonen, J. Binary patterns encoded convolutional neural networks for texture recognition and remote sensing scene classification. *arXiv* **2017**, arXiv:1706.01171.
62. Chaib, S.; Liu, H.; Gu, Y.; Yao, H. Deep feature fusion for VHR remote sensing scene classification. *IEEE Trans. Geosci. Remote Sens.* **2017**, *55*, 4775–4784. [CrossRef]
63. Li, E.; Xia, J.; Du, P.; Lin, C.; Samat, A. Integrating Multilayer Features of Convolutional Neural Networks for Remote Sensing Scene Classification. *IEEE Trans. Geosci. Remote Sens.* **2017**, *55*, 5653–5665. [CrossRef]
64. Ye, L.; Wang, L.; Sun, Y.; Zhao, L.; Wei, Y. Parallel multi-stage features fusion of deep convolutional neural networks for aerial scene classification. *Remote Sens. Lett.* **2018**, *9*, 295–304. [CrossRef]
65. Liu, Y.; Liu, Y.; Ding, L. Scene Classification Based on Two-Stage Deep Feature Fusion. *IEEE Geosci. Remote Sens. Lett.* **2018**, *15*, 183–186. [CrossRef]
66. Yu, Y.; Liu, F. Aerial Scene Classification via Multilevel Fusion Based on Deep Convolutional Neural Networks. *IEEE Geosci. Remote Sens. Lett.* **2018**, *15*, 287–291. [CrossRef]
67. Chowdhury, A.R.; Lin, T.Y.; Maji, S.; Learned-Miller, E. One-to-many face recognition with bilinear cnns. *arXiv* **2015**, arXiv:1506.01342.
68. Lin, T.Y.; RoyChowdhury, A.; Maji, S. Bilinear cnn models for fine-grained visual recognition. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 7–13 December 2015; pp. 1449–1457.
69. Feichtenhofer, C.; Pinz, A.; Zisserman, A. Convolutional two-stream network fusion for video action recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'16), Las Vegas Valley, NV, USA, 26 June–1 July 2016.
70. Park, E.; Han, X.; Berg, T.L.; Berg, A.C. Combining multiple sources of knowledge in deep cnns for action recognition. In Proceedings of the 2016 IEEE Winter Conference on Applications of Computer Vision (WACV), Lake Placid, NY, USA, 7–9 March 2016; pp. 1–8.
71. Wu, Z.; Wang, X.; Jiang, Y.G.; Ye, H.; Xue, X. Modeling spatial-temporal clues in a hybrid deep learning framework for video classification. In Proceedings of the 23rd ACM international conference on Multimedia, Brisbane, Australia, 26–30 October 2015; pp. 461–470.
72. Bodla, N.; Zheng, J.; Xu, H.; Chen, J.C.; Castillo, C.; Chellappa, R. Deep heterogeneous feature fusion for template-based face recognition. In Proceedings of the 2017 IEEE Winter Conference on Applications of Computer Vision (WACV), Santa Rosa, CA, USA, 24–31 March 2017; pp. 586–595.
73. Levi, G.; Hassner, T. Emotion recognition in the wild via convolutional neural networks and mapped binary patterns. In Proceedings of the 2015 ACM on International Conference on Multimodal Interaction, Seattle, WA, USA, 9–13 Novmber 2015; pp. 503–510.
74. Borg, I.; Groenen, P.J. *Modern Multidimensional Scaling: Theory and Applications*; Springer Science & Business Media: Berlin, Germany, 2005.
75. Seber, G.A. *Multivariate Observations*; John Wiley & Sons: Hoboken, NJ, USA, 2009; Volume 252.
76. Rubner, Y.; Tomasi, C.; Guibas, L.J. The earth mover's distance as a metric for image retrieval. *Int. J. Comput. Vis.* **2000**, *40*, 99–121. [CrossRef]
77. Moore, T.; Zirnsak, M. Neural mechanisms of selective visual attention. *Annual Rev. Psychol.* **2017**, *68*, 47–72. [CrossRef] [PubMed]
78. Bauer, A.; Schneider, S.; Waldorf, M.; Braks, K.; Huber, T.J.; Adolph, D.; Vocks, S. Selective visual attention towards oneself and associated state body satisfaction: An eye-tracking study in adolescents with different types of eating disorders. *J. Abnormal Child. Psychol.* **2017**, *45*, 1647–1661. [CrossRef] [PubMed]
79. Zheng, Z.; Zhang, T.; Yan, L. Saliency model for object detection: Searching for novel items in the scene. *Opt. Lett.* **2012**, *37*, 1580–1582. [CrossRef] [PubMed]
80. Ranganath, C.; Rainer, G. Cognitive neuroscience: Neural mechanisms for detecting and remembering novel events. *Nat. Rev. Neurosci.* **2003**, *4*, 193. [CrossRef] [PubMed]
81. Huang, G.; Liu, Z.; Weinberger, K.Q.; van der Maaten, L. Densely connected convolutional networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'17), Honolulu, HI, USA, 21–26 July 2017; Volume 1, p. 3.
82. Ioffe, S.; Szegedy, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv* **2015**, arXiv:1502.03167.

83. Glorot, X.; Bordes, A.; Bengio, Y. Deep sparse rectifier neural networks. In Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics (ICAIS), Klagenfurt, Austria, 6–8 September 2011; pp. 315–323.
84. Abadi, M.; Agarwal, A.; Barham, P.; Brevdo, E.; Chen, Z.; Citro, C.; Corrado, G.S.; Davis, A.; Dean, J.; Devin, M.; et al. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *arXiv* **2016**, arXiv:1603.04467.
85. Steinwart, I.; Christmann, A. *Support Vector Machines*; Springer Science & Business Media: Berlin, Germany, 2008.
86. Weng, Q.; Mao, Z.; Lin, J.; Guo, W. Land-use classification via extreme learning classifier based on deep convolutional features. *IEEE Geosci. Remote Sens. Lett.* **2017**, *14*, 704–708. [CrossRef]
87. Cheriyadat, A.M. Unsupervised feature learning for aerial scene classification. *IEEE Trans. Geosci. Remote Sens.* **2014**, *52*, 439–451. [CrossRef]
88. Zhang, F.; Du, B.; Zhang, L. Saliency-guided unsupervised feature learning for scene classification. *IEEE Trans. Geosci. Remote Sens.* **2015**, *53*, 2175–2184. [CrossRef]
89. Ren, J.; Jiang, X.; Yuan, J. Learning LBP structure by maximizing the conditional mutual information. *Pattern Recognit.* **2015**, *48*, 3180–3190. [CrossRef]
90. Negrel, R.; Picard, D.; Gosselin, P.H. Evaluation of second-order visual features for land-use classification. In Proceedings of the 2014 12th International Workshop onContent-Based Multimedia Indexing (CBMI 2014), Klagenfurt, Austria, 18–20 June 2014; pp. 1–5.
91. Huang, L.; Chen, C.; Li, W.; Du, Q. Remote sensing image scene classification using multi-scale completed local binary patterns and fisher vectors. *Remote Sens.* **2016**, *8*, 483. [CrossRef]
92. Ji, W.; Li, X.; Lu, X. Bidirectional Adaptive Feature Fusion for Remote Sensing Scene Classification. In *Computer Vision, Proccedings of the Second CCF Chinese Conference, CCCV 2017, Tianjin, China, 11–14 October 2017*; Springer: Berlin/Heidelberg, Germany, 2017; pp. 486–497.
93. Bian, X.; Chen, C.; Tian, L.; Du, Q. Fusing local and global features for high-resolution scene classification. *IEEE J. Sel. Top. Appl. Earth Observ. Remote Sens.* **2017**, *10*, 2889–2901. [CrossRef]
94. Gu, X.; Angelov, P.P.; Zhang, C.; Atkinson, P.M. A massively parallel deep rule-based ensemble classifier for remote sensing scenes. *IEEE Geosci. Remote Sens. Lett.* **2018**, *15*, 345–349. [CrossRef]
95. Zhu, Q.; Zhong, Y.; Wu, S.; Zhang, L.; Li, D. Scene Classification Based on the Sparse Homogeneous–Heterogeneous Topic Feature Model. *IEEE Trans. Geosci. Remote Sens.* **2018**, *56*, 2689–2703. [CrossRef]