Scene Classification in Remote Sensing Images using Dynamic Kernels

Rajeshreddy Datla[*†], Vishnu Chalavadi[†] and Krishna Mohan C[†]
[*]*Advanced data processing research institute (ADRIN), Dept. of space, Secunderabad, India*
[†]*Department of Computer Science and Engineering*, *Indian Institute of Technology Hyderabad, Hyderabad, India*
Email: *rajesh@adrin.res.in,†cs16m18p000001@iith.ac.in, †ckm@cse.iith.ac.in

2021 International Joint Conference on Neural Networks (IJCNN) | 978-1-6654-3900-8/21/$31.00 ©2021 IEEE | DOI: 10.1109/IJCNN52387.2021.9533648

***Abstract*—Classification of scenes across multi-sensor remote sensing images with different spatial, spectral, temporal resolutions involves identification of variable length spatial patterns of objects in a scene. So, it necessitates the use of local representations from different regions of a scene in order to comprehend the scene formation. In this paper, we propose a dynamic kernel based representation to handle the patterns of variable lengths in the scenes of remote sensing images. These kernels help to assimilate spatial variability captured using convolutional features in a Gaussian mixture model. The statistics of GMM facilitate the dynamic kernels in preserving the local spatial similarities while handling the changes in spatial content globally within the same scene. The efficacy of the proposed method using two variants of the dynamic kernels is demonstrated on three benchmark scene classification datasets, namely, UCM Land Use (21 classes), Aerial image dataset (30 classes), and NWPU-RESISC45 (45 classes). Our experiments show that the mean interval kernel is better discriminative as it makes use of first and second-order statistics of GMM.**

***Index Terms*—Remote sensing images, scene classification, Gaussian mixture model, MAP adaptation, dynamic kernel.**

I. Introduction

With the available remote sensing technology, abundant volumes of high resolution remote sensing images in large-scale are available for Earth observation. Due to macroscopic coverage of satellites, the spatial content of larger regions on the ground is captured at finer-level in the form of high resolution images. These images depict the objects along with their spatial arrangement on the ground and their quick analysis provides useful insights in the decision making process. Scene classification is one of the high-level tasks in remote sensing imagery analysis that distinctly provides class labels to the scenes that are partitioned from the large images. These class labels are determined based on the local and global semantics of the spatial content present in a scene. However, the acquisition of images from multi-sensors with different spatial and spectral resolutions produces visual discrepancies in the spatial content. Also, the spatial pattern of objects is not uniform and differs across the samples of a specific scene. Following are the factors that manifest different spatial patterns of objects in a remote sensing scene: (i) *Number of objects*: The number of intended objects varies across the samples of a scene. For example, buildings are less in sparse residential scenes compare to dense residential scenes. (ii) *Scale*: The same object with different sizes may present in a scene. (iii) *Scene background*: Most of the samples in a specific scene have non-uniform background. Some samples of freeway scene have with and without trees, with and without buildings. Also, the scenes of *basketball court* encompass different surroundings, e.g., parking area or residential area or trees. (iv) *Proximity*: The distance between the objects vary due to their arbitrary distribution in a scene. (v) *Spatial relation*: The irregular patterns of intended objects in a scene possess non-homogeneous spatial relations. The arrangement of buildings can be observed from the scenes of sparse, medium, and dense residential classes. (vi) *Visual similarity*: Objects in the scenes of different classes look similar, though their functionality differ. For instance, the building structure in *church* and *palace* scenes look visually similar. (vii) *Arbitrary spatial arrangement of objects*: In general, objects are depicted at the center of nature scene due to the awareness of the photographer with the objects. Whereas, the locations of objects in remote sensing scenes are relatively arbitrary due to the macroscopic view from larger distance. Further, the complexity increases with the non-uniform imaging conditions, such as view-angle, illumination, etc.

The convolutional neural network (CNN) features are able to comprehensively describe the local semantics of the scenes, due to their model transferability and generalization ability. Also, various approaches based on CNN architectures have been developed to improve the performance of scene classification in remote sensing images [1]–[5]. Motivated by this, we propose a representation learning approach by leveraging dynamic kernels to handle the variability in spatial patterns of the objects. First, we train a single Gaussian mixture model (GMM) to capture the significant local features by employing convolutional features. Then, we measure the similarity between any two scenes by calculating the distance between the features in the scenes and means of GMM. Kernel methods achieve a better separability across different scene classes by projecting distances to higher dimensions [6]. However, most of these methods are suitable to handle fixed length patterns that restricts comparison between the two scenes which contain variable number of local features. So, we exploit dynamic kernels to transform the variable length patterns of objects in the scenes into fixed-length patterns or to choose the best combination of local features.

The use of base kernel in dynamic kernels helps in measuring the similarity between two scenes by calculating the closeness of their local features. In the probability based

978-0-7381-3366-9/21/$31.00 ©2021 IEEE

kernels, the kernel computation is based on the posterior probability of each local feature corresponding to the GMM. In matching based kernels, the kernel computation includes only those local features that are similar to GMM means. This ensures the retention of key local structures constituting spatial patterns during the kernel computation. These significant spatial patterns uniquely represent some of the scenes, such as row houses in *dense residential*, circular road connectivity in *roundabout* scenes. Therefore, dynamic kernels become obvious choice for representing the similarity between the scenes of remote sensing images.

The contributions of this paper are summarized as:

- The local structure describing the spatial patterns of objects is preserved by learning the statistics of GMM.
- Dynamic kernels are exploited to capture global variations by preserving the local structures while handling variable spatial patterns of objects in the scenes.
- Efficacy of the proposed approach is demonstrated on three varieties of scene classification datasets: UCM Land Use (21 classes), Aerial image dataset (30 classes), and NWPU-RESISC45(45 classes).

The paper is organized as follows. Section II discusses the related research works on scene classification in remote sensing images. Section III describes the proposed approach to classify the scene of remote sensing images using dynamic kernels. In Section IV, we discuss the experimental results and their analysis along with the comparison of state-of-the-art approaches. We conclude the paper in Section V.

II. Related work

In this section, we discuss the existing works on scene classification in remote sensing images. We also briefly summarize the approaches based on dynamic kernels to handle the patterns of variable length.

A. Scene classification in remote sensing images

Existing methods explored the use of various low-level, mid-level, and high-level features for scene classification in remote sensing images [7].

1) Low-level feature-based approaches: The low-level feature-based representations are highly dependent on the design of hand-engineered features which mainly focus on the specific characteristics of images. The most common spatial cues used in their design are color, spatial, texture, shape, and structural information, etc. However, the combination of these spatial cues is often difficult to attain due to the characteristics of remote sensing images. Some works in [7]–[11] exploited the global features such as color histograms and texture descriptors in the scene classification task of remote sensing images. Whereas the use of local features in describing an entire scene requires an additional mechanism to encode its local properties.

2) Mid-level feature-based approaches: The mid-level representations help to describe a scene completely by transforming local features into global features. In [7], [12], bag-of-visual-words (BOVW) with scale-invariant feature transform (SIFT) features [13], locality constrained linear coding (LLC) methods, and combination of BOVW and spatial pyramid matching (SPM) are used in the scene classification of remote sensing images. Some works [14]–[17] explored part detectors to obtain an effective sparselets by employing histogram of oriented gradients (HOG) feature descriptors for scene classification. Various representations using local and global features are fused to obtain effective scene classifiers [18], [19].

3) Convolutional feature-based approaches: Compared to handcrafted feature or mid-level feature based methods, convolutional feature based methods have shown big leap in the performance of scene classification in remote sensing images. This is mainly due to the discriminative capability of convolutional features that provide better transferability and generalization capability. In [1], [20], [21], effectiveness of the pre-trained and fine-tuned versions of AlexNet, VGGNet-16, and GoogLeNet on ImageNet was demonstrated on scene classification in remote sensing images. The ensemble of CNNs is exploited to improve the classification performance over pre-trained CNN models [3], [22]. Further improvements on scene classification are also achieved by stacking, fusing, or integrating various CNN features [23]–[27]. In [28], hybrid deep features are explored for scene representation by fusing scene-based and object-based features from both scene level and region level. In [1], an objective function is augmented besides CNN features to address the issue of intra and inter class variations in scene classification task. A scale-free convolutional neural networks [2] helps to preserve the spatial content, as the input images undergo resize in compliance to the deep architectures during fine-tuning process.

Recently, a key filter bank based CNN (KFBNet) [4] preserves global information for scene classification by capturing the class-specific features from key locations of each scene. Another framework automatically captures the latent ontological structure from the scenes of remote sensing scene images using multi-granularity canonical appearance pooling [5]. Siamese style architecture is used to extract CNN features to discover canonical appearance at each grain level. Gaussian co-variance matrices are derived by computing the second-order statistics over the obtained CNN features. The use of second-order statistics achieve better discrimination capability by adopting suitable normalization factor of the co-variance matrix during the training.

B. Dynamic kernels

In general, the representation of variable length patterns in the applications of speech, music, image, video analysis domains explored the combination of Gaussian mixture model (GMM) and hidden markov model (HMM). Dynamic kernels [29] are one of the most prominent approaches to obtain a fixed length representation from variable length patterns. Lee et al. [30] estimate Gaussian densities to construct a probabilistic sequence kernel (PSK) which produces discriminative features instead of generative features. In order to improve the computational performance of PSK, Bhattacharyya distance-based measure between GMM mixture components is employed which includes both first and second-order GMM statistics [31]. In order to model the features from multiple speakers, a single universal background model (UBM) is trained. The means and covariances are adapted for each speaker from the mean and covariances of UBM resulting in mean interval supervectors. The kernel resulted from supervector is referred to as Gaussian mean interval kernel which is employed in the classification using support vector machine (SVM). Further its computational time is reduced with intermediate matching kernels (IMK) [32]. To select the nearest local feature vectors from each scene, IMK uses the set of virtual feature vectors based on GMM mixtures instead of mean or covariance adaptation. IMK is computationally more efficient than Gaussian mean interval kernel and probabilistic sequence kernel (PSK), as the virtual features obtained from a clip are less than the local features [29]. Also, it was shown that further reduction in computation time is possible by the optimal selection of virtual features.

III. Proposed method

This section explains the proposed approach for scene classification in remote sensing images using dynamic kernels. Fig. 1 shows the block diagram for the proposed method with various stages, such as extraction of convolutional features, Gaussian mixture model training, and classification in kernel space.

A. Extraction of convolutional deep features

In general, fine-tuning over the pre-trained CNN models is performed to retrain on other datasets and they are useful in describing both low and high-level characteristics of a scene. We devise an effective dynamic kernel based representation for scene classification by exploiting the convolutional features from various state-of-the-art CNN architectures. For example the AlexNet produces a feature map of size $13 \times 13 \times 256$ from ”conv5” layer for a 227×227 input image. Similarly, details of the feature maps of various CNN architectures that are used in this work are presented in Table I. Subsequently, the extracted convolutional features are used to train a Gaussian mixture model (GMM) to capture both local and global features implicitly. Then the statistics of GMM are transformed to dynamic kernel space in order to perform scene classification which is described in the following sub-sections.

TABLE I
Details of convolutional features of various CNN architectures used for our scene attribute modeling.

Architecture	**Feature layer**	**Feature map size**
AlexNet [33]	conv5	$13 \times 13 \times 256$
GoogLeNet [34]	inception 4(e)	$14 \times 14 \times 832$
VGGNet-16 [35]	block5_conv3	$14 \times 14 \times 512$
DenseNet-121 [36]	conv5_block16	$7 \times 7 \times 1024$
EfficientNet-B0 [37]	top_conv	$7 \times 7 \times 1280$

B. Gaussian mixture model (GMM) training

Each sample of a scene can be represented as $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_L\}$, where $\mathbf{X}$ is a set of feature vectors & L denotes number of feature vectors extracted from a sample. A separate Gaussian mixture model (GMM) is trained by leveraging the convolutional features for each CNN architecture. The GMM with parameter set $\lambda = \{w_m,\ \boldsymbol{\mu}_m,\ \boldsymbol{\sigma}_m\}$ is represented as

$$p(\mathbf{x}_l|\lambda) = \sum_{m=1}^{M} w_m(\mathbf{x}_l|\boldsymbol{\mu}_m, \boldsymbol{\sigma}_m), \tag{1}$$

where m refers to each GMM component, M denotes number of GMM components. Also, Gaussian mixture weights w_m with mean ($\boldsymbol{\mu}_m$) and covariance ($\boldsymbol{\sigma}_m$) of the GMM component m satisfy the constraint $\sum_{m=1}^{M} w_m = 1$. The standard Expectation-Maximization (EM) method is used to estimate the parameter λ of GMM. Iteratively, the EM algorithm estimates the model parameters such as means, covariances, and coefficients of GMM.

During Expectation-step, the membership probabilities of GMM mixtures are computed for the given features. The Maximization-step re-estimates and maximizes the parameters by the use of membership probabilities [38]. Hypothetically, after GMM training, the attributes describing the spatial pattern of objects is captured in each Gaussian component. Also, the variance of every Gaussian component is responsible for variations in the spatial patterns of different objects in a scene. These features can be specific to a particular scene or may be present across the scenes. Moreover, the large number of attributes captured in the GMM helps in comparing the scenes across spatial patterns of various objects, which subsequently alleviates the intra-class variability.

The trained GMM contains the attributes of all the scenes. To represent a particular scene, maximum a posteriori (MAP) adaptation is performed to highlight the contribution of the features of a sample from the scenes. The probabilistic alignment of each feature vector is calculated from a sample scene for each mixture of the GMM, as the first step in the MAP adaptation process using

$$p(m|\mathbf{x}_l) = \frac{w_m p(\mathbf{x}_l|m)}{\sum_{m=1}^{M} w_m p(\mathbf{x}_l|m)}, \tag{2}$$

where $\mathbf{x}_l$ represents the feature vector of a sample scene and $p(\mathbf{x}_l|m)$ denotes likelihood of the feature $\mathbf{x}_l$ arriving from the mixture m. The probabilistic alignment is used to compute different dynamic kernels in order to obtain an efficient representation of fixed length patterns from variable length patterns as discussed in subsequent sections.

C. Dynamic kernels for variable spatial patterns

Here, we describe the different types of dynamic kernels, namely, supervector, mean interval, and intermediate matching kernels.

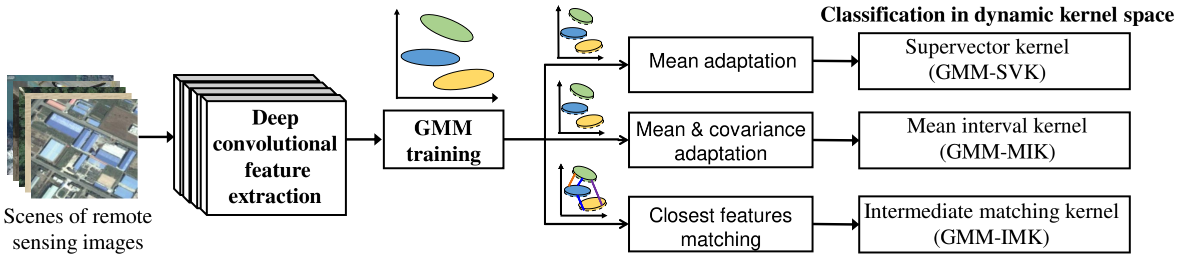
Scenes of remote sensing images
Deep convolutional feature extraction
GMM training
Mean adaptation
Mean & covariance adaptation
Closest features matching
Classification in dynamic kernel space
Supervector kernel (GMM-SVK)
Mean interval kernel (GMM-MIK)
Intermediate matching kernel (GMM-IMK)

Fig. 1. Block diagram of the proposed method for scene classification in remote sensing images using dynamic kernels.

1) Super vector kernel (GMM-SVK): This is a probability based kernel used to compare two scenes by considering the probabilistic distributions of their local feature vectors. This would require the maximum a posteriori of means and covariances of the GMM for each scene obtained from

$$\boldsymbol{\mu}_m(\mathbf{X}) = \alpha \mathbf{F}_m(\mathbf{X}) + (1-\alpha)\boldsymbol{\mu}_m, \tag{3a}$$

and

$$\boldsymbol{\sigma}_m(\mathbf{X}) = \alpha \mathbf{S}_m(\mathbf{X}) + (1-\alpha)\boldsymbol{\sigma}_m. \tag{3b}$$

For a given scene $\mathbf{X}$, Baum-Welch statistics of first-order ($\mathbf{F}_m(\mathbf{X})$) and second-order ($\mathbf{S}_m(\mathbf{X})$) are derived using

$$\mathbf{F}_m(\mathbf{X}) = \frac{1}{n_m(\mathbf{X})} \sum_{l=1}^{L} p(m|\mathbf{x}_l)\mathbf{x}_l, \tag{4a}$$

and

$$\mathbf{S}_m(\mathbf{X}) = diag\left(\sum_{l=1}^{L} p(m|\mathbf{x}_l)\mathbf{x}_l\mathbf{x}_l^T\right). \tag{4b}$$

For a given scene, the posterior probability of a GMM mixture is used in order to adapt the mean and covariance of that particular GMM mixture. An increase in posterior probability indicates the close correlation of the attributes describing spatial patterns captured in the Gaussian component to the attributes describing the spatial patterns in the scene. This shows that the adaptation of means and covariances of a particular mixture are influenced by Baum-Welch statistics of first order ($\mathbf{F}_m(\mathbf{X})$) and second-order ($\mathbf{S}_m(\mathbf{X})$) in comparison to the original GMM mean ($\boldsymbol{\mu}_m$) and covariance ($\boldsymbol{\sigma}_m$). The GMM vector $\boldsymbol{\psi}_m(\mathbf{X})$ for a scene $\mathbf{X}$ is obtained by incorporating the adapted means from (3a) as

$$\boldsymbol{\psi}_m(\mathbf{X}) = \left[\sqrt{w_m}\boldsymbol{\sigma}_m^{-\frac{1}{2}}\boldsymbol{\mu}_m(\mathbf{X})\right]^T. \tag{5}$$

An ($Mf \times 1$)-dimensional GMM supervector is obtained for all the scenes by concatenating the GMM vectors as $\mathbf{s}_{GSV}(\mathbf{X}) = [\boldsymbol{\psi}_1(\mathbf{X})^T, \boldsymbol{\psi}_2(\mathbf{X})^T, \cdots, \boldsymbol{\psi}_M(\mathbf{X})^T]^T$. The supervector kernel between two scenes $\mathbf{X}_i$ and $\mathbf{X}_j$ is then given by

$$K_{GSV}(\mathbf{X}_i, \mathbf{X}_j) = \mathbf{s}_{GSV}(\mathbf{X}_i)^T\mathbf{s}_{GSV}(\mathbf{X}_j). \tag{6}$$

The mean adaptation requires $M \times (L_i + L_j)$ computations in the construction of supervector kernel (SVK). Also, supervector and kernel construction requires $M \times (f_l^2 + 1)$ computations for local feature vectors of f_l dimension. So, the computation time of supervector kernel is $O(ML + Mf_l^2 + f_s^2)$.

2) Mean interval kernel (GMM-MIK): The supervector includes only the GMM statistics of first-order, but not the second-order. So, a mean interval vector is obtained for every component m of GMM by including second-order statistics along with the deviation from the adapted means as

$$\boldsymbol{\psi}_m(\mathbf{X}) = \left(\frac{\boldsymbol{\sigma}_m(\mathbf{X}) - \boldsymbol{\sigma}_m}{2}\right)^{-\frac{1}{2}} (\boldsymbol{\mu}_m(\mathbf{X}) - \boldsymbol{\mu}_m). \tag{7}$$

The adapted parameters and GMM components vary according to the statistical dissimilarity of mean and covariance. So, the covariance and mean statistical dissimilarities of the mean interval vector are indicated in first and second terms of (7). The GMM mean interval (GMI) supervector is constructed by using mean interval vectors across GMM mixtures as $\mathbf{s}_{GMI}(\mathbf{X}) = [\boldsymbol{\psi}_1(\mathbf{X})^T, \boldsymbol{\psi}_2(\mathbf{X})^T, \cdots, \boldsymbol{\psi}_M(\mathbf{X})^T]^T$. Subsequently, the GMM mean interval kernel between two videos $\mathbf{X}_i$ and $\mathbf{X}_j$ is calculated as

$$K_{GMI}(\mathbf{X}_i, \mathbf{X}_j) = \mathbf{s}_{GMI}(\mathbf{X}_i)^T\mathbf{s}_{GMI}(\mathbf{X}_j). \tag{8}$$

The adaptation of mean and covariance requires $2 \times M \times (L_i + L_j)$ computations for constructing mean interval kernel (MIK). Also, f_s^2 computations are needed to form the supervector and kernel of MIK, where f_l denotes the dimension of local features. Hence, the mean interval kernel takes $O(ML + Mf_l^2 + Mf_l + M^2f_s^2)$ computation time.

3) Intermediate matching kernel (GMM-IMK): In addition to the dynamic kernels described above, there exist a matching kernel for comparing scenes explicitly using their local similarity features [39]. The construction of matching kernel uses the similar local features within the pair of scenes as

$$K_{MK}(\mathbf{X}_i, \mathbf{X}_j) = \sum_{l=1}^{L_i} \max_{l'} k(\mathbf{x}_{il}, \mathbf{x}_{jl'}) + \sum_{l'=1}^{L_j} \max_{l} k(\mathbf{x}_{il}, \mathbf{x}_{jl'}), \tag{9}$$

where $k(.,.)$ is a Gaussian kernel, L_i & L_j are the number of feature vectors in scenes $\mathbf{X}_i$ and $\mathbf{X}_j$, respectively. But the matching kernel is too expensive due to the computation of $O(L^2)$ Gaussian kernels, where L denotes the maximal of L_i & L_j.

In order to decrease the computational time of matching kernels, an intermediate matching kernel (IMK) is explored. The construction of intermediate matching kernels considers a set of fixed virtual features to obtain the closest match for the sets of the feature vectors. A set of virtual feature vectors $\mathbf{V} = \{\mathbf{v}_1, \mathbf{v}_2, \cdots, \mathbf{v}_M\}$, which are closest to the m^{th} virtual feature vector $\mathbf{v}_m$ in videos $\mathbf{X}_i$ & $\mathbf{X}_j$ can be calculated as

$$\mathbf{x}^*_{im} = \underset{\mathbf{x} \in \mathbf{X}_i}{\operatorname{argmin}} \, \mathcal{D}(\mathbf{x}, \mathbf{v}_m) \text{ and } \mathbf{x}^*_{jm} = \underset{\mathbf{x} \in \mathbf{X}_j}{\operatorname{argmin}} \, \mathcal{D}(\mathbf{x}, \mathbf{v}_m). \tag{10}$$

The function $\mathcal{D}(.,.)$ computes the similarity between the feature vectors in $\mathbf{X}_i$ or $\mathbf{X}_j$ and the virtual feature vector in $\mathbf{V}$. This similarity measure will help in determining the spatial patterns from each sample of the scenes, which matches the spatial patterns learnt for that particular GMM component. Every component gives the comparison of two scenes. So, even the small correlations in the spatial patterns across the scenes can be measured with the help of a large number of GMM components. This is useful in resolving the high within-class variability.

For each of the M pairs, a base kernel is computed by determining the closeness of the feature vectors. Subsequently, the sum of all the M base kernels are used to define IMK as

$$K_{IMK}(\mathbf{X}_i, \mathbf{X}_j) = \sum_{m=1}^{M} k(\mathbf{x}_{im}, \mathbf{x}_{jm}). \tag{11}$$

The virtual feature vectors encompass the information related to weights, mean vectors, and covariance matrices of GMM components. The posterior probability of each GMM component that generates a feature vector using (2) is used as a similarity measure. Thus, local feature vectors $\mathbf{x}^*_{ic}$ and $\mathbf{x}^*_{jm}$ for scenes $\mathbf{X}_i$ and $\mathbf{X}_j$ similar to a particular virtual feature vector represented by a component m are selected as

$$\mathbf{x}^*_{im} = \underset{\mathbf{x} \in \mathbf{X}_i}{\operatorname{argmax}} \, p(m|\mathbf{x}) \text{ and } \mathbf{x}^*_{jm} = \underset{\mathbf{x} \in \mathbf{X}_j}{\operatorname{argmax}} \, p(m|\mathbf{x}). \tag{12}$$

The computational time of intermediate matching kernel (IMK) includes: (i) $M \times (L_i + L_j)$ posterior probabilities for a mixture component (ii) $M \times (L_i + L_j)$ comparisons for the selection of most similar feature vectors (iii) M number of base kernel computations. These result in a time complexity of $O(ML)$, where L is the maximal of L_i & L_j. The computation time is reduced, when M is less than L_i & L_j.

IV. Experimental results and analysis

In this section, we discuss the evaluation of the proposed method using different kernels, namely, supervector kernel (GMM-SVK), mean interval kernel (GMM-MIK), and intermediate matching kernel (GMM-IMK) on UC Merced, AID, and NWPU-RESISC45 datasets.

A. Datasets and Experimental Settings

To show the efficacy of the proposed approach, we consider three challenging benchmark scene classification datasets as described in Table II. The feature vectors of 5 CNN architectures (details mentioned in TableI), which define both local and global semantics are extracted from each scene for GMM training. In total 20 GMMs are trained, i.e., convolutional features of 5 CNN models for 4 different mixtures on three datasets, namely, UC Merced, AID, and NWPU-RESISC45.

B. Evaluation of dynamic kernels

Tables III, IV, and V present the performance of various kernels on UC Merced, AID, and NWPU-RESISC45 datasets, respectively by formulating kernel based SVM classifier using LibSVM [42]. The performance of dynamic kernels is better with the convolutional features of EfficientNet-B0 and it is observed that the GMM components beyond 128 do not contribute to the improvement of classification performance. Also, it is observed that mean interval kernels (GMM-MIK) and supervector kernels provide better classification performance than intermediate matching kernels (GMM-IMK). This can be attributed to the ability of GMM statistics (first-order and second-order) in GMM-MIK and GMM-SVK which can effectively capture the contextual information along with the variable length spatial patterns of objects. Though the mean interval kernels is not computationally efficient than intermediate matching kernels, the trade-off can be exercised in opting them based on the use-case.

C. Scene-wise analysis

The classification accuracy for each scene of UC Merced, AID, and NWPU-RESISC45 datasets is presented in the confusion matrices as shown in Fig. 2, Fig. 3, and Fig. 4, respectively.

UC Merced dataset - The best classification accuracy of 99.88% is achieved for 64 components using GMM-MIK. Fig. 2 shows that the scene-wise classification performance on test data (20%) which is close to the overall accuracy of scene classification. Also, it can be observed that our GMM-MIK is able to correctly classify all the samples from 20 scene classes. Only one sample from *building* scene is misclassified as *denseresidetnial.*

AID dataset - The dynamic kernel GMM-MIK achieves better scene classification performance of 96.87% and 99.03%

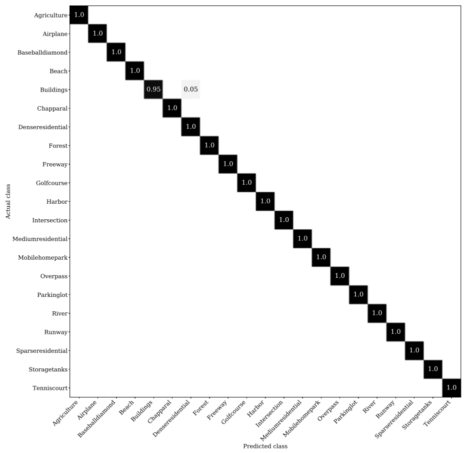

Fig. 2. Confusion matrix on $80\% - 20\%$ training-test ratio of UCM dataset using GMM-MIK for 64 mixtures with EfficientNet-B0 convolutional features.

TABLE II
DETAILS OF THE BENCHMARK DATASETS OF REMOTE SENSING IMAGE SCENE CLASSIFICATION

Dataset	Scene classes	Images per class	Total images	Image sizes	Spatial resolution (m)	Training ratios
UC Merced dataset [40]	21	100	2100	256 × 256	0.3	1 (80%)
AID dataset [41]	30	200 - 400	10000	600 × 600	8 - 0.5	2 (20% & 50%)
NWPU-RESISC45 dataset [7]	45	700	31500	256 × 256	30 - 0.2	2 (10% & 20%)

TABLE III
OVERALL CLASSIFICATION ACCURACY (%) OF GMM-SVK, GMM-MIK, AND GMM-IMK OVER GMM MIXTURES OF $\{2^k\}_{k=4}^{7}$ ON UC MERCED DATASET (80% TRAINING).

CNN Model	**GMM-SVK**				**GMM-MIK**				**GMM-IMK**			
	(Number of GMM mixtures)				(Number of GMM mixtures)				(Number of GMM mixtures)			
	16	**32**	**64**	**128**	**16**	**32**	**64**	**128**	**16**	**32**	**64**	**128**
AlexNet	91.45	92.64	93.83	91.19	92.36	93.80	94.81	92.89	69.22	72.41	74.37	68.56
GoogLeNet	92.91	93.17	94.21	92.88	93.14	94.73	95.42	92.44	72.31	75.17	79.08	76.88
VGG-16	93.18	94.41	95.77	93.90	94.33	95.36	96.12	93.49	72.81	76.70	79.68	77.93
DenseNet-121	94.49	95.62	96.40	93.61	96.44	98.13	99.76	95.66	77.11	80.70	82.58	78.90
EfficientNet-B0	95.62	96.75	96.83	94.84	97.56	98.76	**99.88**	96.89	80.90	83.12	85.54	82.44

TABLE IV
OVERALL CLASSIFICATION ACCURACY (%) OF GMM-SVK, GMM-MIK, AND GMM-IMK OVER GMM MIXTURES OF $\{2^k\}_{k=4}^{7}$ ON AID DATASET.

Training Ratio	**CNN Model**	**GMM-SVK**				**GMM-MIK**				**GMM-IMK**			
		(Number of GMM mixtures)				(Number of GMM mixtures)				(Number of GMM mixtures)			
		16	**32**	**64**	**128**	**16**	**32**	**64**	**128**	**16**	**32**	**64**	**128**
20%	AlexNet	77.21	80.70	83.18	78.43	79.17	83.46	84.82	80.21	63.87	67.36	69.17	65.42
	GoogLeNet	78.85	81.12	84.07	80.01	81.20	84.16	87.76	84.07	65.18	67.43	69.11	65.44
	VGG-16	79.68	82.29	84.91	81.10	85.09	87.03	90.71	88.90	66.18	68.13	72.55	67.74
	DenseNet-121	89.89	91.79	93.56	91.14	92.07	94.50	96.55	93.44	67.22	68.30	73.10	69.14
	EfficientNet-B0	92.34	93.09	95.12	93.45	94.11	95.50	**97.64**	96.77	70.18	75.3	77.25	75.14
50%	AlexNet	80.23	84.66	86.45	83.21	90.23	92.15	94.02	91.21	65.31	69.22	71.34	66.65
	GoogLeNet	81.05	83.12	87.07	84.01	91.20	93.16	96.11	94.87	67.33	69.02	71.56	68.21
	VGG-16	82.68	84.29	88.10	83.35	92.24	94.02	96.71	95.28	68.90	70.70	73.00	69.12
	DenseNet-121	90.21	92.18	93.17	91.62	93.28	95.94	97.82	94.44	70.65	72.81	74.28	71.56
	EfficientNet-B0	93.56	95.22	96.78	94.67	96.72	98.11	**99.03**	97.82	73.29	77.43	80.60	76.37

TABLE V
OVERALL CLASSIFICATION ACCURACY (%) OF GMM-SVK, GMM-MIK, AND GMM-IMK OVER GMM MIXTURES OF $\{2^k\}_{k=5}^{8}$ ON NWPU-RESISC45 DATASET.

Training Ratio	**CNN Model**	**GMM-SVK**				**GMM-MIK**				**GMM-IMK**			
		(Number of GMM mixtures)				(Number of GMM mixtures)				(Number of GMM mixtures)			
		32	**64**	**128**	**256**	**32**	**64**	**128**	**256**	**32**	**64**	**128**	**256**
10%	AlexNet	74.73	76.34	79.27	77.03	76.22	79.17	81.31	78.56	50.75	54.73	60.20	53.56
	GoogLeNet	75.52	77.34	80.37	76.42	79.19	81.54	83.22	80.33	55.28	57.11	61.17	58.09
	VGG-16	76.80	78.34	81.05	78.41	83.04	85.03	87.92	85.11	57.04	59.44	62.66	58.78
	DenseNet-121	79.60	81.64	83.88	81.02	92.81	93.62	95.02	93.44	60.55	64.69	69.29	66.25
	EfficientNet-B0	90.17	91.89	93.56	91.24	93.84	94.72	**95.58**	95.07	64.57	67.03	71.61	66.02
20%	AlexNet	74.73	76.34	79.27	77.03	76.22	79.17	81.31	78.56	50.75	54.73	60.20	53.56
	GoogLeNet	77.00	79.12	82.01	78.11	80.24	82.67	84.91	81.66	57.90	60.29	63.41	61.12
	VGG-16	78.21	80.01	82.76	79.33	84.82	86.10	88.04	85.98	59.73	62.40	65.11	60.39
	DenseNet-121	81.00	82.29	84.52	81.89	93.10	94.17	95.87	93.72	63.41	65.73	70.29	67.60
	EfficientNet-B0	92.34	93.02	94.73	92.00	96.14	97.02	**98.00**	96.56	66.13	69.45	72.48	68.18

with 20% and 50% training data, respectively, for 64 components. Fig. 3 presents the confusion matrix using GMM-MIK on 50% training data. It is evident from the Fig. 3 that the reduction in the confusion of *center*, *church*, *park*, *school* scenes from *square* scene mainly helps in improving the overall classification accuracy.

NWPU-RESISC45 dataset - Fig. 4 provides confusion matrix by considering 20% NWPU-RESISC45 dataset as training data to validate GMM-MIK scene-wise. It is observed that the classification accuracy of all the 45 classes exceeds 90%, and the accuracy exceeds 96% for 43 classes with GMM-MIK. Also, our proposed method greatly reduces the confusion between the scenes of *dense residential* and *medium residential*. However, the confusion between the scenes of*church* and *palace* is the main cause for the overall performance degradation.

TABLE VI
CLASSIFICATION ACCURACY (%) OF THE PROPOSED METHOD WITH STATE-OF-THE-ART-METHODS ON THREE BENCHMARK DATASETS

Method	UC Merced	AID		NWPU-RESISC45	
	$Tr = 80\%$	$Tr = 20\%$	$Tr = 50\%$	$Tr = 10\%$	$Tr = 20\%$
Fine-tuned AlexNet + SVM [7]	94.58 ± 0.11	84.23 ± 0.10	93.51 ± 0.10	81.22 ± 0.19	85.16 ± 0.18
Fine-tuned GoogLeNet + SVM [7]	96.82 ± 0.20	87.51 ± 0.11	95.27 ± 0.10	82.57 ± 0.12	86.02 ± 0.18
Fine-tuned VGGNet-16 + SVM [7]	97.14 ± 0.10	89.33 ± 0.23	96.04 ± 0.13	87.15 ± 0.45	90.36 ± 0.18
*Fine-tuned EfficientNet-B0 + SVM [37]	97.94 ± 0.16	90.33 ± 0.17	96.32 ± 0.18	89.15 ± 0.35	91.24 ± 0.16
D-CNN with AlexNet + SVM [1]	96.67 ± 0.10	85.62 ± 0.10	94.47 ± 0.12	85.56 ± 0.20	87.24 ± 0.12
D-CNN with GoogLeNet + SVM [1]	97.07 ± 0.12	88.79 ± 0.10	96.62 ± 0.10	86.89 ± 0.10	90.49 ± 0.15
D-CNN with VGGNet-16 + SVM [1]	98.93 ± 0.10	90.82 ± 0.16	96.89 ± 0.10	89.22 ± 0.50	91.89 ± 0.22
VGG-VD16+MSCP + SVM [21]	98.36 ± 0.58	91.52 ± 0.21	94.42 ± 0.17	85.33 ± 0.17	88.93 ± 0.14
VGG-VD16+MSCP+MRA + SVM [21]	98.40 ± 0.34	92.21 ± 0.17	96.56 ± 0.18	85.33 ± 0.17	88.93 ± 0.14
VGGNet-16+SF-CNN + SVM [2]	99.05 ± 0.27	93.60 ± 0.12	96.66 ± 0.11	89.89 ± 0.16	92.55 ± 0.14
RTN with VGG-D + SVM [43]	98.96	92.44	–	89.90	92.71
MG-CAP with Sqrt-E [5]	99.00 ± 0.10	93.34 ± 0.18	96.12 ± 0.12	90.83 ± 0.12	92.95 ± 0.13
KFBNet with VGGNet-16 + SVM [4]	99.76 ± 0.24	94.27 ± 0.02	97.19 ± 0.07	90.27 ± 0.02	92.54 ± 0.03
Hydra (DenseNet+ResNet) [22]	–	–	–	92.44 ± 0.34	94.51 ± 0.21
KFBNet with DenseNet-121 + SVM [4]	$\mathbf{99.88 \pm 0.12}$	95.50 ± 0.27	97.40 ± 0.10	93.08 ± 0.14	95.11 ± 0.10
Proposed GMM-SVK (EfficientNet-B0)	97.12 ± 0.18	95.12 ± 0.18	96.78 ± 0.12	93.56 ± 0.14	94.73 ± 0.17
Proposed GMM-MIK (EfficientNet-B0)	$\mathbf{99.88 \pm 0.12}$	$\mathbf{97.64 \pm 0.26}$	$\mathbf{99.03 \pm 0.17}$	$\mathbf{95.58 \pm 0.12}$	$\mathbf{98.00 \pm 0.13}$
Proposed GMM-IMK (EfficientNet-B0)	85.54 ± 0.06	77.25 ± 0.15	80.60 ± 0.10	71.61 ± 0.19	72.48 ± 0.12

*Our evaluation.

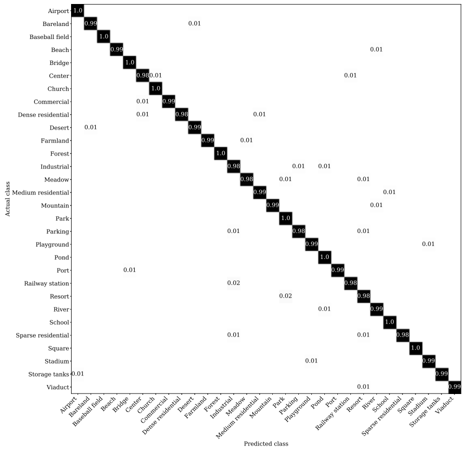

Fig. 3. Confusion matrix on $50\% - 50\%$ training-test ratio of AID dataset using GMM-MIK for 64 mixtures with EfficientNet-B0 convolutional features.

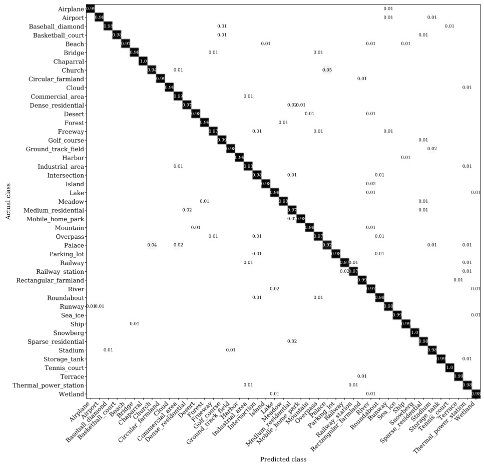

Fig. 4. Confusion matrix on $20\% - 80\%$ training-test ratio of NWPU-RESISC45 dataset using GMM-MIK for 128 mixtures with EfficientNet-B0 convolutional features.

D. Comparison with existing approaches

Table VI provides the comparison of scene classification performance of the proposed approach with existing methods on UC Merced, AID, and NWPU-RESISC45 datasets. Our proposed approach with GMM-MIK outperforms the existing methods by a margin of 2.14% & 1.63% on training ratios of 20% & 50% AID dataset, respectively and a margin of 2.5% & 2.89% on training ratios of 10% & 20% NWPU-RESISC45 dataset, respectively. Also, the average scene classification performance of GMM-MIK on UC Merced dataset is 99.88%. This shows that GMM-MIK is able to capture global variations effectively by preserving the local structures while handling the variable spatial patterns of objects in the scenes. Thus, the use of both first and second-order GMM statistics in capturing the global context of spatial patterns provides valuable information for the scene classification than the local structure of spatial patterns that are captured by CNN models.

V. CONCLUSION

In this paper, we propose an approach based on dynamic kernels to classify scenes of remote sensing images. We exploit various dynamic kernels over the trained Gaussian mixture model to capture the variable spatial patterns of the objects locally while preserving global variations. The mean interval kernel (GMM-MIK) is shown to be effective among the other kernels in handling the variable length spatial patterns of objects in the scenes of remote sensing images. The employment of both first and second-order statistics of Gaussian mixture model in the computation of GMM-MIK provides a valuable information which is useful for the scene classification task. Intermediate matching kernels (GMM-IMK) have better computational time complexity, but they are not very discriminative in comparison to GMM-MIK. The performance of the proposed approach is demonstrated on three varieties of benchmark scene classification datasets. In future, we would experiment these dynamic kernels in the finer categorization of objects in remote sensing images.

References

[1] G. Cheng, C. Yang, X. Yao, L. Guo, and J. Han, "When deep learning meets metric learning: Remote sensing image scene classification via learning discriminative cnns," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 56, no. 5, pp. 2811–2821, 2018.

[2] J. Xie, N. He, L. Fang, and A. Plaza, "Scale-free convolutional neural network for remote sensing scene classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 57, pp. 6916–6928, 2019.

[3] Y. Boualleg, M. Farah, and I. R. Farah, "Remote sensing scene classification using convolutional features and deep forest classifier," *IEEE Geoscience and Remote Sensing Letters*, vol. 16, pp. 1944–1948, 2019.

[4] F. Li, R. Feng, W. Han, and L. Wang, "High-resolution remote sensing image scene classification via key filter bank based on convolutional neural network," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 58, no. 11, pp. 8077–8092, 2020.

[5] S. Wang, Y. Guan, and L. Shao, "Multi-granularity canonical appearance pooling for remote sensing scene classification," *IEEE Transactions on Image Processing*, vol. 29, pp. 5396–5407, 2020.

[6] C. Burges, "A tutorial on support vector machines for pattern recognition," *Data Mining and Knowledge Discovery*, vol. 2, pp. 121–167, 2004.

[7] 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.

[8] G. J. Burghouts and J.-M. Geusebroek, "Performance evaluation of local colour invariants," *Computer Vision and Image Understanding*, vol. 113, no. 1, pp. 48 – 62, 2009. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1077314208001008

[9] J.-M. Geusebroek, R. Boomgaard, A. Smeulders, and H. Geerts, "Color invariance," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, pp. 1338–1350, 2001.

[10] K. van de Sande, T. Gevers, and C. Snoek, "Evaluating color descriptors for object and scene recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 9, pp. 1582–1596, 2010.

[11] J. dos Santos, O. Penatti, and R. Torres, "Evaluating the potential of texture and color descriptors for remote sensing image retrieval and classification." in *VISAPP 2010 - Proceedings of the International Conference on Computer Vision Theory and Applications*, vol. 2, 2010, pp. 203–208.

[12] Y. Yang and S. Newsam, "Comparing sift descriptors and gabor texture features for classification of remote sensed imagery," *2008 15th IEEE International Conference on Image Processing*, pp. 1852–1855, 2008.

[13] G. LoweDavid, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, 2004.

[14] G. Cheng, J. Han, L. Guo, and T. Liu, "Learning coarse-to-fine sparselets for efficient object detection and scene classification," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 1173–1181.

[15] "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.

[16] 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.

[17] G. Cheng, P. Zhou, J. Han, J. Han, and K. Li, "Auto-encoder-based shared mid-level visual dictionary learning for scene classification using very high resolution remote sensing images," *IET Computer Vision*, vol. 9, 2015.

[18] Q. Zhu, Y. Zhong, B. Zhao, G. 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.

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

[20] G. Cheng, Z. Li, X. Yao, L. Guo, and Z. Wei, "Remote sensing image scene classification using bag of convolutional features," *IEEE Geoscience and Remote Sensing Letters*, vol. 14, no. 10, pp. 1735–1739, 2017.

[21] N. He, L. Fang, S. Li, A. Plaza, and J. Plaza, "Remote sensing scene classification using multilayer stacked covariance pooling," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 56, no. 12, pp. 6899–6910, 2018.

[22] R. Minetto, M. P. Segundo, and S. Sarkar, "Hydra: An ensemble of convolutional neural networks for geospatial land classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 57, pp. 6530–6541, 2019.

[23] D. Marmanis, M. Datcu, T. Esch, and U. Stilla, "Deep learning earth observation classification using imagenet pretrained networks," *IEEE Geoscience and Remote Sensing Letters*, vol. 13, pp. 105–109, 2016.

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

[25] W. Zhao and S. Du, "Scene classification using multi-scale deeply described visual words," *International Journal of Remote Sensing*, vol. 37, pp. 4119 – 4131, 2016.

[26] E. Li, J. Xia, P. Du, C. Lin, and A. Samat, "Integrating multilayer features of convolutional neural networks for remote sensing scene classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 55, pp. 5653–5665, 2017.

[27] 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.

[28] C. Sitaula, Y. Xiang, A. Basnet, S. Aryal, and X. Lu, "HDF: Hybrid deep features for scene image representation," *2020 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8, 2020.

[29] A. D. Dileep and C. C. Sekhar, "GMM-based intermediate matching kernel for classification of varying length patterns of long duration speech using support vector machines," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 25, no. 8, pp. 1421–1432, 2014.

[30] K.-A. Lee, C. H. You, H. Li, and T. Kinnunen, "A GMM-based probabilistic sequence kernel for speaker verification," in *INTERSPEECH*, 2007, pp. 294–297.

[31] C. H. You, K. A. Lee, and H. Li, "GMM-SVM kernel with a bhattacharyya-based distance for speaker recognition," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, no. 6, pp. 1300–1312, 2010.

[32] S. Boughorbel, J. P. Tarel, and N. Boujemaa, "The intermediate matching kernel for image local features," *Proceedings. 2005 IEEE International Joint Conference on Neural Networks, 2005.*, vol. 2, pp. 889–894 vol. 2, 2005.

[33] A. Krizhevsky, I. Sutskever, and G. Hinton, "Imagenet classification with deep convolutional neural networks," *Neural Information Processing Systems*, vol. 25, 01 2012.

[34] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1–9, 2015.

[35] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *CoRR*, vol. abs/1409.1556, 2015.

[36] G. Huang, Z. Liu, G. Pleiss, L. Van Der Maaten, and K. Weinberger, "Convolutional networks with dense connectivity," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–1, 2019.

[37] M. Tan and Q. V. Le, "EfficientNet: Rethinking model scaling for convolutional neural networks," *ArXiv*, vol. abs/1905.11946, 2019.

[38] R. Duda, P. Hart, and D. Stork, "Pattern classification (2nd ed.)," 1999.

[39] C. Wallraven, B. Caputo, and A. B. A. Graf, "Recognition with local features: the kernel recipe," *Proceedings Ninth IEEE International Conference on Computer Vision*, pp. 257–264 vol.1, 2003.

[40] Y. Yang and S. Newsam, "Bag-of-visual-words and spatial extensions for land-use classification," in *GIS '10*, 2010.

[41] 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, pp. 3965–3981, 2017.

[42] C.-C. Chang and C. Lin, "Libsvm: A library for support vector machines," *ACM Trans. Intell. Syst. Technol.*, vol. 2, pp. 1–27, 2011.

[43] Z. Chen, S. Wang, X. Hou, and L. Shao, "Recurrent transformer network for remote sensing scene categorisation," in *BMVC*, 2018.