CNN-Based Deep Spatial Pyramid Match Kernel for Classification of Varying Size Images

Shikha Gupta[1(✉)], Manjush Mangal[1], Akshay Mathew[1], Dileep Aroor Dinesh[1(✉)], Arnav Bhavsar[1], and Veena Thenkanidiyoor[2]

[1] School of Computing and EE, Indian Institute of Technology, Mandi, H.P., India
{shikha_g,manjush_mangal,akshay_mathew}@students.iitmandi.ac.in
{addileep,arnav}@iitmandi.ac.in
[2] Department of CSE, National Institute of Technology Goa, Ponda, India
veenat@nitgoa.ac.in

Abstract. This paper addresses the issues of handling varying size images in convolutional neural networks (CNNs). When images of different size are given as input to a CNN then it results in varying size set of activation maps at its convolution layer. We propose to explore two approaches to address varying size set of activation maps for the classification task. In the first approach, we explore deep spatial pyramid match kernel (DSPMK) to compute a matching score between two varying size sets of activation maps. We also propose to explore different pooling and normalization techniques for computing DSPMK. In the second approach, we propose to use spatial pyramid pooling (SPP) layer in CNN architectures to remove fixed-length constraint and to allow the original varying size image as input to train and fine-tune the CNN for different datasets. Experimental results show that proposed DSPMK-based SVM and SPP-layer based CNN frameworks achieve state-of-the-art results for scene image classification and fine-grained bird species classification tasks.

Keywords: Convolutional neural network
Deep spatial pyramid match kernel · Image classification
Varying size set of activation map · Spatial pyramid pooling layer
Support vector machine

1 Introduction

Nowadays CNNs have been popular for their relevance to wide extent of applications, such as image segmentation [22], object classification [4,12,31], scene image classification [18,41], fine-grained classification [2,9,43] and so on. Fine-grained recognition has recently become popular [2,44], because it is applicable in a variety of challenging domain such as bird species recognition [35] or flower species recognition [27]. An important issue in fine-grained recognition is

M. De Marsico et al. (Eds.): ICPRAM 2018, LNCS 11351, pp. 44–64, 2019.
https://doi.org/10.1007/978-3-030-05499-1_3

inter-class similarity *i.e*, images of birds with different species can be ambiguous due to uncontrolled natural settings. On the other hand, generic scene image recognition is challenging task because scene images are composed with spatially correlated layout of different objects and concepts [19]. Successful recognition methods need to extract powerful visual representations to deal with high intra-class and low inter-class variability [38], complex semantic structure [29], varying size of same semantic concept across dataset, and so on. For addressing such issues many CNNs like, AlexNet [23], GoogLeNet [32] and VGGNet-16 [31] have already been trained on datasets like Places [45] and ImageNet [7] for image recognition tasks. These deep networks can be altered and prepare to train for other datasets and applications with less modifications. In all similar scenarios, features acquired from pre-trained, altered or fine-tuned CNNs are used to build standard classifier like fully connected neural network or support vector machine (SVM).

One major drawback of these frameworks is that the CNNs require the input images to be of fixed dimensions. For instance, GoogLeNet accepts images of resolution "224×224". Although the standard datasets like SUN397 [38] or MIT-67 [29] consist of variable resolution images which are much bigger than "224×224". Similarly, in case of CUB-200-2011 [35] bird dataset images are varying in size. Also, as we demonstrate, it is useful to consider bird region of interest (ROI) which focuses on the subject, and discards most of the background. In such cases, too the size of ROI can vary with the shape and size of the birds. The traditional methods to use these CNNs is to reshape the random-sized images to a same size. This leads to dissipation of information of the images before giving as input to the CNN for extracting the feature. The capability of classifier to give better results gets affected due to such usage, which is evident from the work published in [18]. To avoid any such prior information loss, different approaches are explored to feed varying resolution images as input to CNN. The works in [18], eliminates the necessity of fixed resolution image by including a spatial pyramid pooling (SPP) layer in CNN and titled the new architecture as SPP-Net. The works in [11], follows the similar technique by evaluating the feature maps of conv layer into a super vector using one of the encoding like Fisher vector (FV) [41] or vector of locally aggregated descriptor (VLAD) [20] by building the Gaussian mixture model (GMM).

As conv layers are the necessary part of convolutional neural network and responsible for producing discriminative activation maps. Generated activation maps are of varying resolution according to original image size and contain more spatial layout information compared to the activation of the fc layers, as fc layer integrates the spatial content present in the conv layer features. Inspired by the same fact, in our previously published work [16], we focused on passing the images in their actual size as input to the convolutional neural network and then acquire varying size sets of deep activation maps from the last conv layer as output.

In literature study, mainly two approaches are proposed to handle varying size pattern classification using support vector machines. In the first, a varying size set of activation maps is transformed into a fixed dimension pattern as

in [11], and further a kernel function for fixed dimension pattern is used to build the support vector machine classifier. In the second, a suitable kernel function is directly designed for varying size set of activation maps. The kernel designed for varying size set of features is called dynamic kernels [8]. The dynamic kernels in [8,15,17,24] shows promising results for classification of varying resolution images and speech signals. We adopt the second approach and propose to design deep spatial pyramid match kernel (DSPMK) as dynamic kernel.

In this work, we extended the previous work of [16] and propose to explore different pooling and normalization techniques for computing DSPMK which is discussed in Sect. 3.2. Inspired from [18], we propose to consider the CNN architecture for fine-tuning by passing original images as input and added spatial pyramid pooling (SPP) layer to the network for handling the same. SPP-layer maps varying size activation maps to fixed size for passing to fully-connected layer. SPP-layer allows for end-to-end fine-tuning and training of the network with variable size images. This is discussed in Sect. 3.1. The key contribution of this work are:

- Deep spatial pyramid match kernel with different pooling and normalization technique to find the similarity score between a pair of varying size set of deep activation maps.
- Introducing SPP-layer [18] in between last convolutional layer and first fc-layer, so that varying size deep activation maps of images can be converted into fixed length representation.
- End-to-end fine-tuning of the network for different dataset with SPP-layer to handle the images in their original size.
- We demonstrate the effectiveness of our approach and its variants, with state-of-the-art results, over two different applications of scene image classification and fine-grained bird image classification.

The rest of the paper is structured as follows: A review of related approaches for image classification using CNN-based features is presented in Sect. 2. Section 3.1, gives the detail about CNN architecture with SPP-layer. In Sect. 3.2, we discuss the DSPMK for varying size set of deep activation maps with different pooling and normalization technique. The experimental studies using the proposed approach on scene image classification and fine-grained bird classification tasks is presented in Sect. 4. In Sect. 5 conclusion is presented.

2 Literature Review

In this section, we revisit the state-of-the-art techniques for fine-grained image classification and scene image classification tasks. Traditional method of image classification includes generating the local feature vector of images using local descriptors like, scale invariant feature transform (SIFT) [25] and histogram of oriented gradient (HOG) [6]. Further, GMM-based or SVM-based classifier can be built using the standard function such as Gaussian kernel, where the feature vectors are encoded into a fixed length representation. Generally bag of

Visual Words (BoVW) [5,36,37], sparse coding [40], and Fisher vector (FV) [26] encoding is used for fixed-dimensional representations of an image. These fixed length vector representations does not incorporate spatial information of concepts present in the image.

As an alternative, SVM-based classifiers can be learn with matching based dynamic kernels which are designed with consideration of spatial information. Spatial pyramid match kernel [24], class independent GMM-based intermediate matching kernel [8] and segment-level pyramid match kernel [15] are few of the matching based kernels for matching different size images and speech signals. With the development of deep CNNs, conventional features and related methods are being replaced by leaned features from datasets with linear kernel (LK) based SVM classifier.

The eye-popping performance of various deep CNN architectures on ImageNet large scale visual recognition challenge (ILSVRC) [23,31,32] has motivated the research community to adapt CNNs to other challenging domain and datasets like fine-grained classification. Initially, fc-layer features from convolutional neural network were directly in use to build SVM-based classifier using LK for any task of vision and perform batter than traditional methods [9,46]. Few researchers also encoded learned features into a novel representation e.g, in [26] authors have transformed the features from fc layer to bag of semantics representation. This bag of semantic representation is then summaries in semantic Fisher vector representation. In case of fine-grained bird classification, state-of-the-art approaches are based on deep CNNs [2,9,39,44]. These approach consider part based and bounding box annotation for generating the final representation. Moreover, all these approaches are based on giving fixed size input to the network because of rigid nature of fc-layer as it is based on fixed number of fully connected neurons and expects a fixed length representation of input, whereas the convolution process is not constrained with fixed length representation. So we can say, the necessity of fixed resolution image as input to convolutional networks is an mandatory demand of the fc-layer.

The impact of reshaping the images to a fixed size results in loss of information [18]. On the other side, convolutional layers of CNNs accept any arbitrary sized input image which results in random sized deep activation maps according to the input. Deep activations maps contain the strongest response of filters on the previous layer output and conserve the spatial information of the concepts present in input image. From the work in [11,18], we can observe the similar idea. The approaches in these papers considered spatial pyramid based approach and scaled space of input images to incorporate the concept information of images into the activation maps at different scales. The work in [42], focuses on scale characteristics of images over feature activations. They consider images at different scales to input to the CNN and obtain seven layer pyramid of dense activation maps. Further they have used Fisher framework for encoding the activation maps to aggregate into a fixed length representation. The work proposed in [18], considers the SPP approach to expel the essentialness of same size image as input to convolutional networks. Here, the CNN is fed with images of original

size. However, in [42] the CNN is fed with differently scaled images. The work in [11], also follow the similar way and fed the original sized image to convolutional network. However, the approach for converting into fixed size is different. Here, a GMM using fixed size vector representation obtained from spatial pyramid pooling, is built to generate the Fisher vectors [11]. Finally, all the Fishers vectors are concatenated to form a fixed dimensional representation.

In our work, we focuses on integrating the power convolutional-based varying size set of deep activation maps with dynamic kernel to obtain a matching value between a pair of images of different size. We used DSPMK as dynamic kernel rather of building GMM based dictionary on varying size conv features. In this way, our proposed approach is computationally less expensive. Further we modify the deep CNN architecture by adding the spatial pyramid pooling for end-to-end fine-tuning or training. In the next Section, we discuss CNN architecture with SPP-layer and the proposed DSPMK for the varying size set of deep activation maps.

3 Approaches for Handling Variable Size Images for Classification

In this section, we discuss the approaches to handle the variable size images in CNN for classification on different domain datasets like scene image classification dataset and fine-grained bird classification dataset.

- In Sect. 3.1, we introduce SPP-layer inspired by [18] in between last convolutional layer and first dense layer so that variable size set of convolutional activation maps of images can be converted into fixed length representation for end-to-end training of the network.
- In Sect. 3.2, we present DSPMK proposed in [16] to compute the similarity score between two sample images represented as varying size set of activation maps.

3.1 CNN Architecture with Spatial Pyramid Pooling (SPP) Layer

As mentioned earlier, traditional CNN architecture like AlexNet [23], GoogLeNet [32] or VGGNet-16 [31] are pre-trained on the dataset with images of fixed resolution (e.g 227×227). Conversion of image from original size to fixed size results in loss of information in the beginning of network. CNN architecture is mainly the combination of convolutional (conv) layer and fully-connected (fc) layer. The fc-layer demands fixed size input and conv-layers are free from such restrictions. In this work, we modify the CNN architecture such that images are allowed in its original size for input and gives varying size set of activation maps as output from last convolutional layer.

To handle this further, we propose to use spatial pyramid pooling (SPP) layer inspired by [18] which map varying size set of deep activation maps onto a fixed length representation for end-to-end training. Using SPP-layer, information

aggregation happens at later stage in the network which improve the training process. We have considered VGG-19 architecture [30,31] and added SPP-layer between last convolutional layer and first fc-layer. The SPP-layer sum-pools the varying length convolutional layer activation maps at three different levels to convert them into fixed length vector. In the first level complete convolutional activation maps are considered and sum or max-pool is applied to obtain a fixed length vector. In the second level, activation maps are spatially divided into 4 blocks and sum or max-pooling is applied in respective block for converting the variable size deep activation maps to fixed length vector. In the third level, activation maps are divided into 16 blocks and so on. In this scenario we are fixing the number of spatially divided blocks instead of block size. The fixed length vectors obtained in each level are concatenated to form a fixed length supervector. This fixed length supervector is further passed onto fully-connected layer for end-to-end training using back propagation. The block diagram of proposed CNN architecture with SPP-layer is shown in Fig. 1.

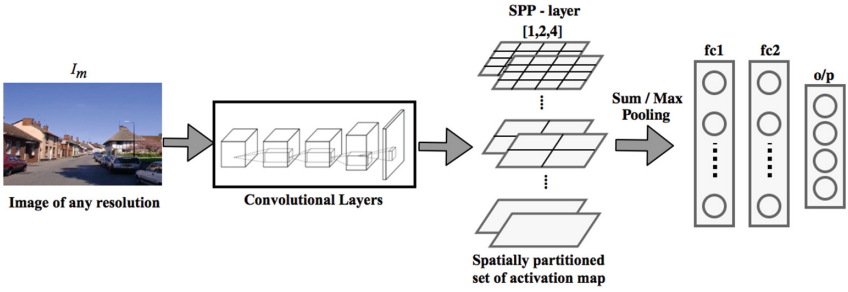

Fig. 1. Block diagram of CNN architecture with SPP-layer.

3.2 Deep Spatial Pyramid Match Kernel

In this section, we present deep spatial pyramid match kernel (DSPMK) proposed in [16] for matching varying size set of deep activation maps obtained from convolutional neural network. The entire process of classification using DSPMK-based SVM is demonstrated in block diagram of Fig. 2. As shown in diagram, I_m and I_n are two images given to the convolutional layer of network as input such that we get set of deep activation maps. Different image give variant size activation maps as output i.e, activation maps in set corresponding to image I_m is different from image I_n. From these different size activation maps, we propose to compute similarity sore using DSPMK. DSPMK-based SVM classifier is learn by association of feature maps of training images with the class label. This is in contrast to [11], In [11] varying size activation maps are transformed into fixed size using Fisher framework and are encoded to fixed length super vector like Fisher vector and then LK-based SVM is used for building the classifier. Main

features of the proposed approach is that, DSPMK computes the similarity score on different size actual images at different spatial levels ranging from 0, 1 to L using varying size set of deep activation maps.

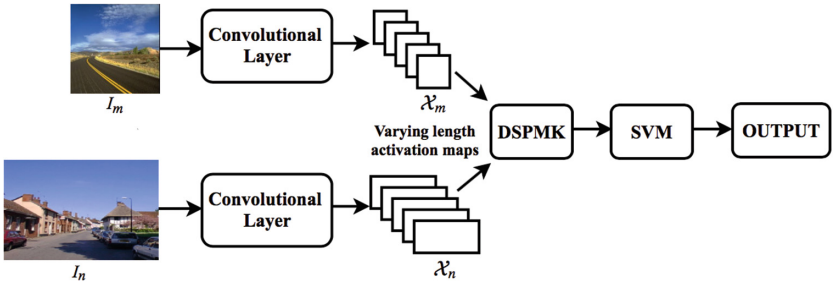

Fig. 2. Block diagram of DSPMK as proposed in [16].

Consider dataset of images as $\mathcal{D} = \{I_1, I_2, \ldots, I_m, \ldots, I_N\}$ and 'f' be the number of kernels or filters in last conv layer of pre-defined deep CNN architecture. Let the mapping $\mathcal{F}$, takes input, actual image and project it to set of deep activation maps using conv layers of CNN. Mapping $\mathcal{F}$ is given as, $\mathcal{X}_m = \mathcal{F}(I_m)$. Size of activation maps obtain from last conv layer in a set corresponding to a image is same but vary from image to image as images are fed in its original resolution to the CNN architecture.

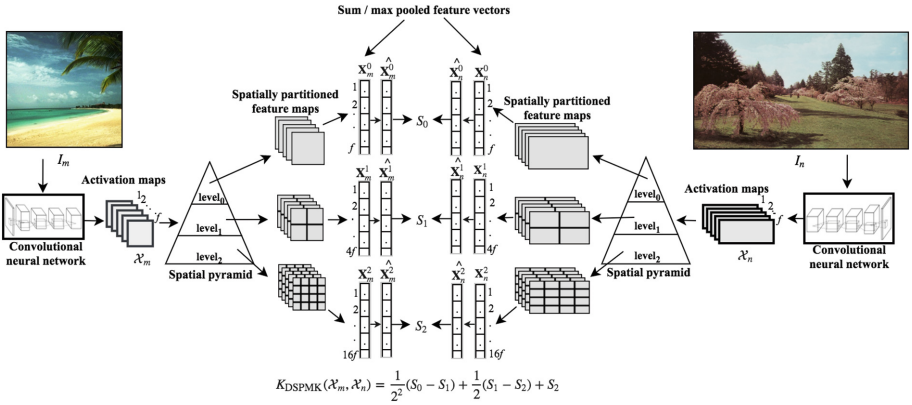

Fig. 3. Illustration of computing similarity score using DSPMK between two different resolution images I_m and I_n, similar to the Fig. 2 of paper [16]. Here, $\mathcal{X}_m$ and $\mathcal{X}_n$ are set of deep activation map computed using conv layer of pre-trained CNN, size of $\mathcal{X}_m$ depends on size of I_m, similarly size of $\mathcal{X}_n$ depends on size of I_n. The matching score at each level l (*i.e*, S_0, S_1 and S_2) is computed using Eq. (3).

Firstly, images to pre-trained CNN is fed in its actual size. For image I_n, we have a set $\mathcal{X}_n = \{\mathbf{x}_{n1}, \mathbf{x}_{n2}, \mathbf{x}_{n3}, \ldots, \mathbf{x}_{nf}\}$ consisting of 'f' feature maps from mapping $\mathcal{F}$, where $\mathbf{x}_{ni} \in \mathbb{R}^{p_n \times q_n}$ and $p_n \times q_n$ is the size of each feature map obtained from last conv layer which varies accordant to the input image resolution. This conclude to varying size deep activation map as shown in Fig. 3 for images I_m and I_n.

Secondly, deep activation maps are spatially divided into sub-blocks to form spatial pyramid. At level-0, activation maps are considered as it is without spatial division. At level-1, every deep activation map is divided into 4 spatial divisions related to 4 quadrant, as depicted in Fig. 3. Consider $L+1$ the total number of levels in the pyramid start from 0, 1 till L. In At any level-l, a deep activation map $\mathbf{x}_{ni}$ is spatially split into 2^{2l} blocks. At any level-l, activation values of cells in every spatial block of all the f deep activation maps are sum or max-pooled and concatenated to form a vector $\mathbf{X}_n^l$ of size $f2^{2l} \times 1$. This scenario is expatiated in Fig. 3 by considering three different levels, $l = 0$, 1 and 2 and same is also described in Algorithm 1 for $L+1$ pyramid levels.

In our proposed framework, we considered three spatial pyramid levels. At level-0, ($i.e$, $l = 0$) the complete activation maps corresponding to input image is sum or max-pooled, in total there are f activation maps in output of conv layer which results is $f \times 1$ size vector representation. At level-1 ($i.e$, $l = 1$), the same activation maps are considered again and divided into four equal spatial blocks. Each block correspond to single activation maps are again sum or max pooled, which results in 1×4 size vector. Same procedure is repeated for f activation maps which results into a vector of $4f \times 1$ size. Similarly, at level-2 ($i.e.$ $l = 2$), again the same activation maps are divided into sixteen equal spatial regions resulting into a vector of $16f \times 1$ dimensional vector. Corresponding to image I_n, after concatenating all the sum or max pooled activation values are results in a single vector called $\mathbf{X}_n^l$.

The $\mathbf{X}_m^l$ can now be seen as representation of image I_m at level-l of pyramid. At this stage, we propose to compute deep spatial pyramid match kernel (DSPMK) to match two images rather than deriving Fisher vector (FV) representation as in [11]. Our proposed approach avoids building GMM to obtain FV and hence reduces the computation complexity as compared to [11]. The process of computing DSPMK is motivated from spatial pyramid match kernel (SPMK) [24]. SPMK involves the histogram intersection function that match the frequency based image representation or normalized vector representation of two images at every levels of pyramid [24]. However, $\mathbf{X}_m^l$ is not in the normalized vector representation of image I_m. We propose to normalized $\mathbf{X}_m^l$ using ℓ_1 and ℓ_2 to obtain normalized vector representation.

Let $\mathbf{X}_m^l$ and $\mathbf{X}_n^l$ be the representation at level-l corresponding to two images I_m and I_n respectively. The normalized vector representation of $\mathbf{X}_m^l$ and $\mathbf{X}_n^l$ is obtained using ℓ_1 or ℓ_2 normalization as given in Eqs. (1) and (2)

$$\widehat{\mathbf{X}}_m^l = \frac{\mathbf{X}_m^l}{||\mathbf{X}_m^l||_1}, \widehat{\mathbf{X}}_n^l = \frac{\mathbf{X}_n^l}{||\mathbf{X}_n^l||_1} \tag{1}$$

Algorithm 1. Deep spatial pyramid matching kernel $K_{\text{DSPMK}}(\mathcal{X}_m, \mathcal{X}_n)$.

Require:

(i) Activation maps set $\mathcal{X}_m$ and $\mathcal{X}_n$
$\mathcal{X}_m = \{\mathbf{x}_{m1}, ..., \mathbf{x}_{mi}, ..., \mathbf{x}_{mf}\}$; where $\mathbf{x}_{mi} \in \mathbb{R}^{p_m \times q_m}$
$\mathcal{X}_n = \{\mathbf{x}_{n1}, ..., \mathbf{x}_{ni}, ..., \mathbf{x}_{nf}\}$; where $\mathbf{x}_{mi} \in \mathbb{R}^{p_n \times q_n}$

(ii) $L + 1$: number of pyramid levels.

1: **Procedure:**

2: **for** l=0 **to** L **do**

3: Divide each activation map of $\mathcal{X}_m$ into 2^{2l} equal spatial blocks.
$\mathcal{X}_m^l =$
$\{\mathbf{x}^l_{m1(1)}...\mathbf{x}^l_{m1(2^{2l})}, ..., \mathbf{x}^l_{mi(1)}...\mathbf{x}^l_{mi(2^{2l})}, ..., \mathbf{x}^l_{mf(1)}...\mathbf{x}^l_{mf(2^{2l})}\}$

4: Apply sum or max-pooling over each block such that
$x^l_{mi(j)} = \sum_u \sum_v \mathbf{x}^l_{mi(j)}(u, v)$
$\mathbf{X}^l_m =$
$\{x^l_{m1(1)}...x^l_{m1(2^{2l})}, ..., x^l_{mi(1)}...x^l_{mi(2^{2l})}, ..., x^l_{mf(1)}...x^l_{mf(2^{2l})}\}$
$\in \mathbb{R}^{f2^{2l} \times 1}$

5: Normalize the generated feature vector $\mathbf{X}^l_m$ using ℓ_1 or ℓ_2 norm
$\widehat{\mathbf{X}}^l_m =$
$\{\hat{x}^l_{m1(1)}...\hat{x}^l_{m1(2^{2l})}, ..., \hat{x}^l_{mi(1)}...\hat{x}^l_{mi(2^{2l})}, ..., \hat{x}^l_{mf(1)}...\hat{x}^l_{mf(2^{2l})}\}$
$\in \mathbb{R}^{2^{2l \times f} \times 1}$

6: Repeat step 3 to 5 for computing $\widehat{\mathbf{X}}^l_n$ for image I_n

7: Compute intermediate similarity score S_l between $\widehat{\mathbf{X}}^l_m$ and $\widehat{\mathbf{X}}^l_n$ using Equation (3).

8: **end for**

9: Compute final similarity score between $\mathcal{X}_m$ and $\mathcal{X}_n$ using Equation (4).

Ensure:

9: $K_{\text{DSPMK}}(\mathcal{X}_m, \mathcal{X}_n)$

$$\widehat{\mathbf{X}}^l_m = \frac{\mathbf{X}^l_m}{||\mathbf{X}^l_m||_2}, \widehat{\mathbf{X}}^l_n = \frac{\mathbf{X}^l_n}{||\mathbf{X}^l_n||_2} \tag{2}$$

The Histogram intersection (HI) function is used to compute intermediate matching score S_l between $\widehat{\mathbf{X}}^l_m$ and $\widehat{\mathbf{X}}^l_n$ at each level l as,

$$S_l = \sum_{j=1}^{f} \sum_{k=1}^{2^{2l}} min(\hat{x}^l_{mj(k)}, \hat{x}^l_{nj(k)}) \tag{3}$$

Here, the intermediate similarity score S_l found at level-l also includes all the matches found at the finer level $l + 1$. As a result, the number of new matches found at level l is given by $S_l - S_{l+1}$ for $l = 0, \ldots, L - 1$. The DSPMK is computed as a weighted sum of the number of new matches at different levels of the spatial pyramid. The weight associated with level l is set to $\frac{1}{2^{(L-l)}}$, which is inversely proportional to width of spatial regions at that level.

The DSPMK kernel is computed as,

$$K_{\text{DSPMK}}(\mathcal{X}_m, \mathcal{X}_n) = \sum_{l=0}^{L-1} \frac{1}{2^{L-l}}(S_l - S_{l+1}) + S_L \tag{4}$$

The main advantages of proposed approach is that it incorporate any size image without any resizing loss and it combines the convolutional varying size deep activation maps with dynamic kernel named DSPMK based SVM.

4 Experimental Studies

In this section, the efficacy of the proposed framework is studied on scene image and bird species classification task using SVM-based classifier. In experiments, we cover mainly two aspects for handling varying nature of image; one by computing the varying size deep activation maps from last convolutional layer and compute the classification score using DSPMK, and the other by adding the spatial pyramid pooling layer to the network for handling varying nature of image and fine-tuned it with respective dataset for computing the fully trained features.

4.1 Datasets

We tested our proposed approach on two different kinds of datasets one for scene classification which includes datasets such as MIT-8 Scene [28], Vogel-Schiele (VS) [34], MIT-67 [29] and SUN-397 [38], and the other for fine-grained bird species classification with the CUB-200-2011 [35] dataset.

MIT-8-scene: This dataset contain total of 2688 scene images belonging to 8 different semantic classes, like, 'coast', 'mountain', 'forest', 'open-country', 'inside-city', 'highway', 'tall building' and 'street'. We randomly select 100 scene images from each class for training the model and keep remaining images for testing. We consider 5 such sets. The final classification scores computed in this paper correspond to the average classification accuracy for 5 trials.

Vogel-Schiele: This dataset contain total of 700 scene images belonging to 6 different semantic classes, viz., 'forests', 'mountains', 'coasts', 'river', 'open-country', and 'sky-clouds'. We consider 5-fold stratified cross validation and present the result as average classification score of 5-fold.

MIT-67: This is indoor scene dataset. Most of the scene recognition models work well for outdoor scenes but perform poorly in the indoor domain. This dataset contain 15,620 images with 67 scene categories. All images have a minimum resolution of 200 pixels in the smallest axis. It is a challenging dataset, due to the less in-class variability. The standard division [29] for this dataset consist of approximately 80 images of each class for training and 20 images for testing.

SUN397: This database contains 397 categories used in the benchmark of several paper. The number of images varies across categories like indoor, urban and nature but there are at least 100 images per category, and 108,754 images in total. We consider publicly available fixed train and test splits from [38], where

each split has 50 training and 50 testing images per category. We consider the first five split set and the result computed is the average classification accuracy for 5 splits.

Caltech-UCSD Birds CUB-200-2011 dataset consists of 11,788 images of birds belonging to 200 different species. The standard division for this dataset consist of 5994 images for training and 5794 for testing [35] with approximately 30 images per species in the training class, and the rest in test class. Bird data suffers from high intra-class and low inter-class variance. Dataset is available with bird bounding boxes and other annotations. In this work, we evaluate our methods in two scenarios one with the bounding-box which enables one to focus on the bird region rather then background and other without bounding-box information considered at training and test time.

4.2 Experiment Studies for Scene Image Classification Task

In our studies of scene image classification, we have consider different CNN architectures for extracting the features like, AlexNet [23], GoogLeNet [32] and VGGNet-16 [31] which are pre-trained on three different datasets, i.e, ImageNet [7], Places205 and Places365 [45] datasets. Reason behind using the different pre-trained networks (on different datasets) is that all used datasets consist of variety of images. In this context, ImageNet dataset contains mainly object centric images and it shows activations for object-like structures, whereas Places dataset comprise of largely indoor/outdoor scene images. We believe that CNNs trained on Places dataset activate for landscapes, natural structure of scenes with more spatial features, and indoor scene patterns.

In all the convolutional networks, pre-trained weights are kept consistent without fine-tuning. These networks are in use without its fc-layers in our experimental studies so that input images of arbitrary size can be accepted. As discussed in Sect. 3.2, we have passed the original image of arbitrary size as input to deep CNNs and extracted varying size set of deep activation maps from last convolutional layer. The size of set of activation map corresponding to an image depends on the filter size, number of filters f in last convolutional layer and input image size. The number of filters f, in last convolution layer of AlexNet, GoogLeNet and VGGNet-16 are 256, 1024 and 512 respectively. The architecture of these CNNs also differs from each other. So, activation map size will vary from image to image and architecture to architecture.

DSPMK between varying size deep activation map for pair of images is computed as in Fig. 3 using Eq. (1) to (4). We consider $L + 1 = 3$ as the number of levels in spatial pyramid. In computation of DSPMK, we have performed the experiments with both sum and max-pooling techniques. Reason behind using different pooling technique is, max-pooling extracts the most activated feature like edges, corner and texture, whereas, sum-pooling smoothen out the activation map and measures the sum value of existence of a pattern in a given region. Although results with both the pooling technique are comparable as shown in Tables 1 and 2, we observed that max-pooling works a bit better then sum-pooling in our case. It is seen that performance of SVM-based classifier with

Table 1. Comparison of classification accuracy (CA) (in %) with 95% confidence interval for the **SVM-based classifier** using DSPMK computed using sum-pooling on different datasets, similar to study shown in Table 1 of paper [16]. Base features for the proposed approach are extracted from different CNN architecture like, AlexNet, GoogLeNet and VGGNet which are pre-trained deep network on ImageNet, Places-365 and Places-205 dataset respectively.

Different pre-trained deep CNN architectures used to build DSPMK with sum-pooling	MIT-8 scene	Vogel-Schiele	MIT-67	SUN-397
ImageNet-AlexNet [23]	93.52±0.13	79.46±0.23	62.46	45.46±0.12
Places205-AlexNet [46]	93.56±0.12	82.21±0.25	62.24	53.21±0.23
Places365-AlexNet [45]	94.15±0.11	82.90±0.31	66.67	55.43±0.24
ImageNet-GoogLeNet [32]	92.02±0.06	82.30±0.25	71.78	50.32±0.31
Places205-GoogLeNet [46]	92.15±0.18	85.84±0.36	75.97	57.43±0.26
Places365-GoogLeNet [45]	93.70±0.16	85.54±0.21	75.60	59.89±0.21
ImageNet-VGG [31]	93.90±0.07	84.62±0.31	75.78	53.67±0.25
Places205-VGG [46]	94.54±0.03	**86.92±0.26**	**81.87**	61.86±0.24
Places365-VGG [45]	**95.09±0.14**	84.68±0.28	77.76	**62.31±0.25**

DSPMK obtained using deep features from VGGNet-16 is significantly better than that of SVM with DSPMK obtained using deep features from GoogLeNet and AlexNet. Reason being VGGNet-16 has very deep network compare to other architectures and it learns the hierarchical representation of visual data more efficiently. We consider LIBSVM [3] tool to build the DSPMK-based SVM classifier. Specifically, we uses one-against-the-rest approach for multi-class scene image classification. In SVM for building the classifier, we use default value of trade-off parameter $C = 1$. In our further study, we fine-tuned the VGG-16 architecture for respective datasets by adding the spatial pyramid pooling (SPP) layer to the network as shown in Fig. 2. We computed the spatial pyramid pooling features and train the neural network based classifier. We consider the neural network with two hidden layer and one soft-max layer. Dropout is chosen as 0.5 learning rate as 0.01 and 2048 neurons in the hidden layers. We observe that results are comparable with DSPMK-based SVM approach.

Table 3 presents the comparison of scene image classification accuracy of proposed DSPMK-based SVM classifier and the SPP-based neural network classifier with that of state-of-the-art approaches. From Table 3, it is seen that both of our proposed approaches are giving better performance in comparison with

Table 2. Comparison of classification accuracy (CA) (in %) with 95% confidence interval for the **SVM-based classifier** using DSPMK computed using max-pooling on different datasets, similar to study shown in Table 1 of paper [16]. Base features for the proposed approach are extracted from different CNN architecture like, AlexNet, GoogLeNet and VGGNet which are pre-trained deep network on ImageNet, Places-365 and Places-205 dataset respectively.

Different pre-trained deep CNN architectures used to build DSPMK with max-pooling	MIT-8 scene	Vogel-Schiele	MIT-67	SUN-397
ImageNet-AlexNet [23]	94.12±0.11	80.17±0.16	63.67	46.12±0.13
Places205-AlexNet [46]	94.11±0.13	83.18±0.21	63.56	54.01±0.21
Places365-AlexNet [45]	94.65±0.08	83.11±0.20	68.21	56.12±0.23
ImageNet-GoogLeNet [32]	91.12±0.09	83.21±0.27	72.99	52.12±0.28
Places205-GoogLeNet [46]	93.14±0.12	86.91±0.31	76.82	56.12±0.24
Places365-GoogLeNet [45]	92.89±0.13	86.67±0.24	77.22	60.13±0.23
ImageNet-VGG [31]	93.86±0.11	85.21±0.33	75.99	54.91±0.22
Places205-VGG [46]	95.56±0.06	**87.81±0.21**	**82.83**	62.76±0.22
Places365-VGG [45]	**96.21±0.09**	85.66±0.30	78.16	**63.12±0.21**

traditional feature based approaches in [21,25] and also with CNN-based approaches in [11,13,26,42,46].

The works in [25], uses scale invariant feature transform (SIFT) descriptors to represent images as set of local feature vectors, which are further converted into bag-of-visual word (BOVW) representation for classification using linear kernel based SVM classifier. The works in [21] uses the learned bag-of-part (BoP) representation and combine with improved Fisher vector for building linear kernel based SVM classifier. The works in [13] extracted CNN-based features from multiple scale of image at different levels and performs orderless vectors of locally aggregated descriptors (VLAD) pooling [20] at every scale separately. The representations from different level are then concatenated to form a new representation known as multi-scale orderless pooling (MOP) which is used for training linear kernel based SVM classifier. The works in [46] uses more direct approach, where a large scale image dataset (Places dataset) is used for training the AlexNet architecture and extracted fully-connected (fc7) layer feature from the trained network. The basic architecture of their Places-CNN is same as that of the AlexNet [23] trained on ImageNet. The works in [46] trained a Hybrid-CNN, by combining the training data of Places dataset with ImageNet dataset. Here, features from fully-connected (fc7) layer are then used for training linear kernel

Table 3. Comparison of classification accuracy (CA) (in %) with 95% confidence interval of proposed approach with state-of-the-art approaches on MIT-8 scene, Vogel-Schiele, MIT-67 Indoor and SUN-397 dataset, similar to study shown in Sect. 4, Table 2 of paper [16]. (SIFT: Scale invariant feature transform, IFK: Improved Fisher kernel, BoP: Bag of part, MOP: Multi-scale orderless pooling, FV: Fisher vector, DSP: Deep spatial pyramid, MPP: Multi-scale pyramid pooling, DSFL: Discriminative and shareable feature learning and NN: Neural network).

Method	MIT-8-Scene	Vogel Schiele	MIT-67	SUN-397
SIFT+BOVW [25]	79.13±0.13	67.49±0.21	45.86	24.82±0.34
IFK+BoP [21]	85.76±0.12	73.23±0.23	63.18	-
MOP-CNN [13]	89.45±0.11	76.81±0.27	68.88	51.98±0.24
Places-CNN-fc7 [46]	88.30±0.09	76.02±0.31	68.24	54.32±0.14
Hybrid-CNN-fc7 [46]	91.23±0.04	78.56 ±0.21	70.80	53.86±0.21
fc8-FV [26]	88.43±0.08	79.56±0.23	72.86	54.40±0.30
VGGNet-16 + DSP [11]	92.34±0.12	81.34±0.27	76.34	57.27±0.34
MPP(Alex-fc7)+DSFL [42]	93.21±0.14	82.12±0.25	80.78	-
VGG16 + SPP-feature + NN based classifier (Ours)	94.01±0.11	85.89±0.23	80.94	-
VGG16 + DSPMK-based SVM with max-pooling (Ours)	**96.21±0.09**	**87.81±0.21**	**82.83**	**63.12±0.21**

based SVM classifier. The works in [26] obtained the semantic Fisher vector (FV) using standard Gaussian mixture encoding for CNN-based feature. Further linear kernel based SVM classifier is build using semantic FV for classification of scene images. The works in [11] uses the generative model based approach to build a dictionary on top of CNN activation maps. A FV representation for different spatial region of activation map is then obtained from the dictionary. A power and l_2 normalization is applied on the combined FV from different spatial region. A linear kernel based SVM classifier is then used for scene classification. The works in [42] combine the features from fc7 layer of AlexNet (Alex-fc7) and their complementary features named discriminative and shareable feature learning (DSFL). DSFL learns discriminative and shareable filters with a target dataset. The final image representation is used with the linear kernel based SVM classifier for the scene classification task.

In contrast to all the above briefly explained approaches, our proposed approach es use the image of arbitrary size and gives the deep activation map of varying size without any loss of information. The deep spatial pyramid match

kernel can handle the varying size set of deep activation maps and incorporates the local spatial information at the time of computing level wise matching score. Specifically, our proposed approach is very simple and discriminative in nature which outperforms the other CNN-based approaches without combining any complementary features as in [42]. Our first proposed approach, based on SPP-feature with neural network (NN) also shows good quality results (second to only our proposed DSPMK method), as this approach consider original size images for fine-tuning the network. Our second proposed framework, bring out that for scene recognition, good performance is accomplishable by using last conv layer features with DSPMK-based-SVM. Proposed framework is free of fully connected layer, believe on the actual size image, memory efficient, simple and take very less computing time in compare to state-of-the-art techniques.

4.3 Experiment Studies for Fine-Grained Bird Species Classification

The experiments for fine-grained bird species classification cover three main aspects of our approach. First, we compute varying size deep activation map by passing images in its original size without any prior loss of information. Second, we use DSPMK to compute matching score between them. Third, we fine-tune the VGG-19 architecture [30] by adding SPP-layer to it. We fine-tune the network for CUB-200-2011 dataset [35] and compute variable size deep activation map features and SPP-features for further experiments. We show our proposed approach is generic and along with scene image classification it works well for fine-grained bird species classification.

Table 4, shows the results of fine-grained bird species classification with different methods. We have shown the results for testing with bounding box (Bbox) and without bounding box. The bounding box annotation essentially helps us to crop only the prominent bird region of interest (RoI) while discarding the background. Such regions may also be obtained by detection algorithm. The case without Bbox corresponds to complete actual image. Firstly, we passes the image in fixed size i.e, "224 × 224" for both the cases to the CNN architecture and computed fixed length fc7 and pool5 features. We use linear kernel based SVM to compute the classification score. Secondly, we pass the image in its original size without resizing it to"224 × 224" and computed varying size deep activation maps. In this context, we perform experiments using DSPMK-based SVM with different pooling technique for computing the classification score. Next, we fine-tune the VGG-19 architecture by adding SPP-layer between last convolutional layer and first fully-connected layer. We consider the fine-tuned network for further experiments in two ways. In the first approach, we compute the varying size set of deep activation map and use DSPMK-based SVM for computation of classification score. In the second approach, we compute SPP-features from fine-tuned network and train neural network based classifier. In this context, we uses two hidden layer with 2096 neurons in each. We empirically chosen learning rate as 0.001 and dropout as 0.5.

We observe in Table 4 that, if the images are not resized and no Bbox RoI detection is available, original images can be used instead with proposed

Table 4. Comparison of classification accuracy (CA) (in %) for the SVM-based classifier using linear kernel and DSPMK, fine-tuned VGG19 with SPP-layer based neural network on CUB-200-2011 dataset. Proposed approach uses base features extracted from VGG19 [30]. Here NN indicate neural network.

Method	Testing with Bbox	Testing without Bbox
VGG19-fc7+ linear kernel based SVM	79.02	72.94
VGG19-Pool5+ linear kernel based SVM	78.30	69.84
SVM using DSPMK with sum-pooling	82.07	74.36
SVM using DSPMK with max-pooling	**82.12**	80.81
VGG19 + fine-tuning with SPP + fc7	78.41	76.63
VGG19 + fine-tuning with SPP + SVM using DSPMK with sum-pooling	79.11	78.16
VGG19 + fine-tuning with SPP + SVM using DSPMK with max-pooling	80.01	78.89
VGG19 + SPP-feature + NN based classifier	81.24	**81.03**

DSPMK-based SVM approach. In this context, one can notice that classification accuracy will be marginally affected. This is natural as the case with Bbox focuses only on the bird RoI. However, this difference is relatively small for most of variants of the proposed methods using DSPMK and SPP. This indicates that for bird images of the size and scale as in the CUB dataset, the proposed methods are largely invariant to ROI selection, and thus can obviate an ROI detection step. When images are used without bounding box annotation, we observe that there is huge i.e, (approx 10%) improvement in performance from linear kernel based SVM with VGG-19 pool5 features to DSPMK-based SVM with varying size activation maps features from last conv layer. We believe that, our proposed approach compute the matching score between two images more efficiently with consideration of spatial information.

In Table 5, we compare the classification results of proposed approaches with state-of-the-art results. The deformable part descriptor (DPD) in [44], is based on the supervised version of deformable part models (DPD) [10] for training, which then allows for pose normalization by comparing corresponding parts. The work in [1], learns a linear classifier for each pair of parts and classes.

Table 5. Comparison of classification accuracy (CA) (in %) on CUB-200-2011 dataset between different state-of-the-art method with that of the proposed approaches. Some of the state-of-the-art approaches uses part annotations during training and testing. The proposed approaches do not use any part information. (DPD: Deformable part descriptors; POOFs: Part-based One-vs-One Features; NN: Neural Network)

Method	Accuracy	Remark
DPD [44]	50.98	Uses parts info
POOFs [1]	56.78	Uses parts and Bbox info
Part transfer [14]	57.84	Uses parts and Bbox info
$DeCAF_6$ [9]	58.75	Uses Bbox info
DPD + $DeCAF_6$ [9]	64.96	Uses parts and Bbox info
Pose Normalized CNN [2]	75.70	Uses parts info
Parts-RCNN-FT [43]	76.37	Uses parts info
VGG19 + fine-tuning with SPP + fc7 (Ours)	78.41	Uses Bbox info
VGG19 + fine-tuning with SPP + DSPMK (Ours)	79.11	Uses Bbox info
VGG19 + SPP-feature + NN based classifier (Ours)	81.24	Uses Bbox info
DSPMK-based SVM with max-pooling (Ours)	**82.12**	Uses Bbox info

The decision values from many of such classifiers are used as feature representation. This approach also require ground-truth part annotations at training and also at test time. The work in, [14], is based on nonparametric part detection. Here, the basic idea is to use nearest neighbor matching to obtain similar training example from human-annotated part positions. The work in [9] is based on feature extraction from part regions detected using a DPM, which have sufficient depictive power and generalization ability to perform desired task. The work in [2] uses deep CNNs for extracting the features from image patches that are located and normalized by the pose. The work in [43], generate object proposals using Selective Search [33] and uses the part locations to calculate localized features from R-CNNs.

From Table 5, we also infer that our approaches for bird species classification does not require part annotation, and yet improves over very complex state-of-the-art approaches that use part based annotation at the time of training

and testing. In contrast, our approaches are generic and easy to adapt to other datasets as we only require a pre-trained CNN architecture. For fine-tuning the CNN architecture with SPP-layer, we perform experiments without bounding box as well as with bounding box. It is observed that proposed framework perform much improved without any extra annotations.

5 Conclusion

In this work, we propose deep spatial pyramid match kernel (DSPMK) for improving the base features from last conv CNN's layer. DSPMK-based SVM can classify different size images which are represented as the varying size set of deep activation maps. Further, we propose to add spatial pyramid pooling layer in CNN architecture so that, we can fine-tune the pre-trained CNNs for other datasets containing varying size images. Our model has a dynamic kernel which calculates the layer-wise intermediate matching score and strengthens the matching procedure of conv layer features. The training of DSPMK-based SVM classifier take very less time in compare to training of GMM in [11]. In our research, we have considered the last convolutional layer features rather than fc layer features as fc layer limits these features to the fixed size and requires much larger computation time as it contains approximately 90% of the aggregate parameters of CNN. Thus, conv layer features are effectively considered in handling large varying size images in scene image classification datasets like, SUN-397 and MIT-67, as well as for size variations in the fine-grained classification with the CUB dataset. Almost all approaches in fine-grained classification are specialized, but we show that our approach is generic and works well for both the diverse datasets. In terms of performance, our proposed approach accomplishes state-of-the-art results for standard scene classification and bird species classification dataset. In future, for capturing differences of the activations caused by the varying size of concepts in an image, multi-scale deep spatial pyramid match kernel can be investigated.

References

1. Berg, T., Belhumeur, P.N.: Poof: part-based one-vs.-one features for fine-grained categorization, face verification, and attribute estimation. In: 2013 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 955–962. IEEE (2013)
2. Branson, S., Van Horn, G., Belongie, S., Perona, P.: Bird species categorization using pose normalized deep convolutional nets. arXiv preprint arXiv:1406.2952 (2014)
3. Chang, C.C., Lin, C.J.: LIBSVM: A library for support vector machines. ACM Trans. Intell. Syst. Technol. **2**, 27:1–27:27 (2011)
4. Chatfield, K., Simonyan, K., Vedaldi, A., Zisserman, A.: Return of the devil in the details: delving deep into convolutional nets. arXiv preprint arXiv:1405.3531 (2014)

5. Csurka, G., Dance, C., Fan, L., Willamowski, J., Bray, C.: Visual categorization with bags of keypoints. In: Workshop on Statistical Learning in Computer Vision, ECCV, vol. 1, pp. 1–2. Prague (2004)
6. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: 2005 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), vol. 1, pp. 886–893 (2005)
7. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: a large-scale hierarchical image database. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2009, pp. 248–255. IEEE (2009)
8. Dileep, A.D., Chandra Sekhar, C.: GMM-based intermediate matching kernel for classification of varying length patterns of long duration speech using support vector machines. IEEE Trans. Neural Netw. Learn. Syst. **25**(8), 1421–1432 (2014)
9. Donahue, J., et al.: Decaf: a deep convolutional activation feature for generic visual recognition. In: International Conference on Machine Learning, pp. 647–655 (2014)
10. Felzenszwalb, P.F., Girshick, R.B., McAllester, D., Ramanan, D.: Object detection with discriminatively trained part-based models. IEEE Trans. Pattern Anal. Mach. Intell. **32**(9), 1627–1645 (2010)
11. Gao, B.B., Wei, X.S., Wu, J., Lin, W.: Deep spatial pyramid: the devil is once again in the details. CoRR abs/1504.05277 (2015)
12. Girshick, R., Donahue, J., Darrell, T., Malik, J.: Rich feature hierarchies for accurate object detection and semantic segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 580–587 (2014)
13. Gong, Y., Wang, L., Guo, R., Lazebnik, S.: Multi-scale orderless pooling of deep convolutional activation features. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014. LNCS, vol. 8695, pp. 392–407. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-10584-0_26
14. Göring, C., Rodner, E., Freytag, A., Denzler, J.: Nonparametric part transfer for fine-grained recognition. In: CVPR, vol. 1, p. 7 (2014)
15. Gupta, S., Dileep, A.D., Thenkanidiyoor, V.: Segment-level pyramid match kernels for the classification of varying length patterns of speech using svms. In: 2016 24th European Signal Processing Conference (EUSIPCO), pp. 2030–2034. IEEE (2016)
16. Gupta, S., Pradhan, D., Dileep, A.D., Thenkanidiyoor, V.: Deep spatial pyramid match kernel for scene classification. In: ICPRAM, pp. 141–148 (2018)
17. Gupta, S., Thenkanidiyoor, V., Aroor Dinesh, D.: Segment-level probabilistic sequence kernel based support vector machines for classification of varying length patterns of speech. In: Hirose, A., Ozawa, S., Doya, K., Ikeda, K., Lee, M., Liu, D. (eds.) ICONIP 2016. LNCS, vol. 9950, pp. 321–328. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46681-1_39
18. He, K., Zhang, X., Ren, S., Sun, J.: Spatial pyramid pooling in deep convolutional networks for visual recognition. IEEE Trans. Pattern Anal. Mach. Intell. **37**(9), 1904–1916 (2015)
19. Henderson, J.: Introduction to real-world scene perception. Vis. Cogn. **12**(6), 849–851 (2005)
20. Jégou, H., Douze, M., Schmid, C., Pérez, P.: Aggregating local descriptors into a compact image representation. In: 2010 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3304–3311. IEEE (2010)
21. Juneja, M., Vedaldi, A., Jawahar, C., Zisserman, A.: Blocks that shout: distinctive parts for scene classification. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 923–930 (2013)
22. Kang, K., Wang, X.: Fully convolutional neural networks for crowd segmentation. arXiv preprint arXiv:1411.4464 (2014)

23. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Advances in Neural Information Processing Systems, pp. 1097–1105 (2012)
24. Lazebnik, S., Schmid, C., Ponce, J.: Beyond bags of features: spatial pyramid matching for recognizing natural scene categories. In: 2006 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), vol. 2, pp. 2169–2178 (2006)
25. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. Int. J. Comput. Vis. **60**(2), 91–110 (2004)
26. Mandar, D., Chen, S., Gao, D., Rasiwasia, N., Nuno, V.: Scene classification with semantic fisher vectors. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2974–2983 (2015)
27. Nilsback, M.E., Zisserman, A.: Automated flower classification over a large number of classes. In: Sixth Indian Conference on Computer Vision, Graphics & Image Processing, ICVGIP 2008, pp. 722–729. IEEE (2008)
28. Oliva, A., Torralba, A.: Modeling the shape of the scene: a holistic representation of the spatial envelope. Int. J. Comput. Vis. **42**(3), 145–175 (2001)
29. Quattoni, A., Torralba, A.: Recognizing indoor scenes. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2009, pp. 413–420. IEEE (2009)
30. Simon, M., Rodner, E.: Neural activation constellations: Unsupervised part model discovery with convolutional networks. In: International Conference on Computer Vision (ICCV) (2015)
31. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)
32. Szegedy, C., et al.: Going deeper with convolutions. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–9 (2015)
33. Uijlings, J.R., Van De Sande, K.E., Gevers, T., Smeulders, A.W.: Selective search for object recognition. Int. J. Comput. Vis. **104**(2), 154–171 (2013)
34. Vogel, J., Schiele, B.: Natural scene retrieval based on a semantic modeling step. In: Enser, P., Kompatsiaris, Y., O'Connor, N.E., Smeaton, A.F., Smeulders, A.W.M. (eds.) CIVR 2004. LNCS, vol. 3115, pp. 207–215. Springer, Heidelberg (2004). https://doi.org/10.1007/978-3-540-27814-6_27
35. Wah, C., Branson, S., Welinder, P., Perona, P., Belongie, S.: The caltech-ucsd birds-200-2011 dataset (2011)
36. Wang, J., Yang, J., Yu, K., Lv, F., Huang, T., Gong, Y.: Locality-constrained linear coding for image classification. In: 2010 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3360–3367 (2010)
37. Wang, Z., Feng, J., Yan, S., Xi, H.: Linear distance coding for image classification. IEEE Trans. Image Process. **22**(2), 537–548 (2013)
38. Xiao, J., Hays, J., Ehinger, K.A., Oliva, A., Torralba, A.: Sun database: large-scale scene recognition from abbey to zoo. In: 2010 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3485–3492. IEEE (2010)
39. Xiao, T., Xu, Y., Yang, K., Zhang, J., Peng, Y., Zhang, Z.: The application of two-level attention models in deep convolutional neural network for fine-grained image classification. In: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 842–850. IEEE (2015)
40. Yang, J., Yu, K., Gong, Y., Huang, T.: Linear spatial pyramid matching using sparse coding for image classification. In: 2009 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1794–1801 (2009)
41. Yoo, D., Park, S., Lee, J.Y., Kweon, I.S.: Fisher kernel for deep neural activations. arXiv preprint arXiv:1412.1628 (2014)

42. Yoo, D., Park, S., Lee, J.Y., So Kweon, I.: Multi-scale pyramid pooling for deep convolutional representation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp. 71–80 (2015)
43. Zhang, N., Donahue, J., Girshick, R., Darrell, T.: Part-based R-CNNs for fine-grained category detection. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014. LNCS, vol. 8689, pp. 834–849. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-10590-1_54
44. Zhang, N., Farrell, R., Iandola, F., Darrell, T.: Deformable part descriptors for fine-grained recognition and attribute prediction. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 729–736 (2013)
45. Zhou, B., Lapedriza, A., Khosla, A., Oliva, A., Torralba, A.: Places: a 10 million image database for scene recognition. IEEE Trans. Pattern Anal. Mach. Intell. **40**, 1452–1464 (2017)
46. Zhou, B., Lapedriza, A., Xiao, J., Torralba, A., Oliva, A.: Learning deep features for scene recognition using places database. In: Advances in Neural Information Processing Systems, pp. 487–495 (2014)