Multi-scale Pyramid Pooling for Deep Convolutional Representation

Donggeun Yoo, Sunggyun Park, Joon-Young Lee, and In So Kweon

KAIST
Daejeon, 305-701, Korea.
dgyoo@rcv.kaist.ac.kr, sunggyun@kaist.ac.kr, jylee@rcv.kaist.ac.kr, iskweon@kaist.ac.kr

Abstract

Compared to image representation based on low-level local descriptors, deep neural activations of Convolutional Neural Networks (CNNs) are richer in mid-level representation, but poorer in geometric invariance properties. In this paper, we present a straightforward framework for better image representation by combining the two approaches. To take advantages of both representations, we extract a fair amount of multi-scale dense local activations from a pre-trained CNN. We then aggregate the activations by Fisher kernel framework, which has been modified with a simple scale-wise normalization essential to make it suitable for CNN activations. Our representation demonstrates new state-of-the-art performances on three public datasets: 80.78% (Acc.) on MIT Indoor 67, 83.20% (mAP) on PASCAL VOC 2007 and 91.28% (Acc.) on Oxford 102 Flowers. The results suggest that our proposal can be used as a primary image representation for better performances in wide visual recognition tasks.

1. Introduction

Image representation is one of the most important factors that affect performance on visual recognition tasks. Barbu *et al*. [3] introduced an interesting experiment that a simple classifier along with human brain-scan data substantially outperforms the state-of-the-art methods in recognizing action from video clips.

With a success of local descriptors [22], many researches have devoted in studying global image representation based on a Bag-of-Word (BOW) model [32] that aggregates abundant local statistics captured by hand-designed local descriptors. The BOW representation is further improved with VLAD [15] and Fisher kernel [27, 26] by adding higher order statistics. One major benefit of these global representations based on local descriptors is their invariance property to scale changes, location changes, occlusions and background clutters.

In recent computer vision researches, drastic advances of visual recognition are achieved by deep convolutional neural networks (CNNs) [5], which jointly learn the whole feature hierarchies starting from image pixels to the final class posterior with stacked non-linear processing layers. A deep representation is quite efficient since its intermediate templates are reused. However, the deep CNN is non-linear and have millions of parameters to be estimated. It requires strong computing power for the optimization and large training data to be generalized well. The recent presence of large scale ImageNet [6] database and the raise of parallel computing contribute to the breakthrough in visual recognition. Krizhevsky *et al*. [20] achieved an impressive result using a CNN in large-scale image classification.

Instead of training a CNN for a specific task, intermediate activations extracted from a CNN pre-trained on independent large data have been successfully applied as a generic image representation. Combining the CNN activations with a classifier has shown impressive performance in wide visual recognition tasks such as object classification [29, 8, 25, 13, 4], object detection [11, 13], scene classification [29, 12, 40], fine-grained classification [29, 38], attribute recognition [39], image retrieval [2], and domain transfer [8].

To utilize CNN activations as a generic image representation, a straightforward way is to extract the responses from the first or second fully connected layer of a pre-trained CNN by feeding an image and to represent the image with the responses [8, 2, 13, 11]. However, this representation is vulnerable to geometric variations. There are techniques to address the problem. A common practice is exploiting multiple jitterred images (random crops and flips) for data augmentation. Though the data augmentation has been used to prevent over-fitting [20], recent researches show that *average pooling* in a test stage, augmenting data and averaging the multiple activation vectors, also helps achieve better geometric invariance while improving the performance by +2.92% in [4] and +3.3% in [29] on PASCAL VOC 2007.

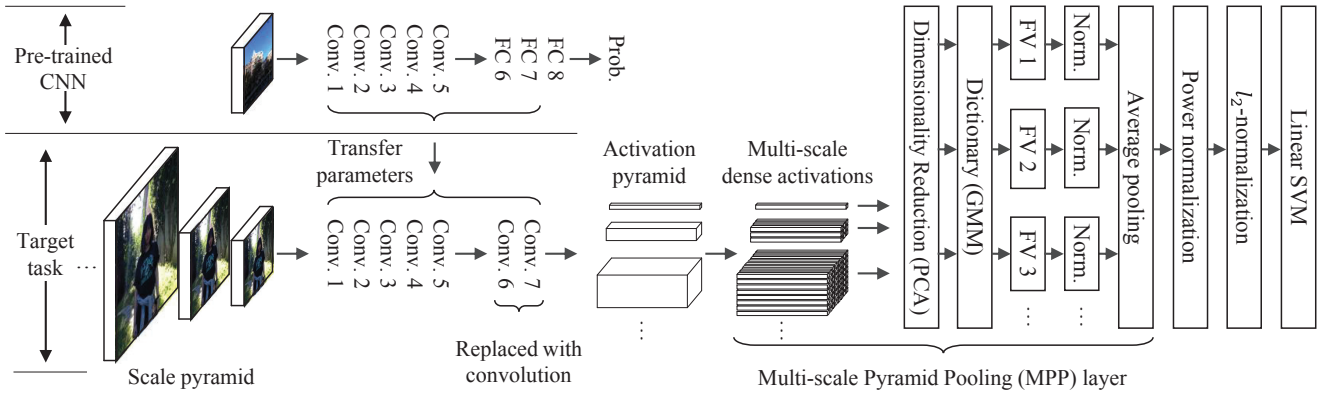

Figure 1. A pipeline of the proposed method. Given a pre-trained CNN, following [33], we replace the first two fully connected layers with the two equivalent convolutional layers. It enables us to efficiently obtain large amount of multi-scale dense activations. The activations are followed by the Multi-scale Pyramid Pooling (MPP) layer we suggest. The consequent image representation is combined with the linear SVM for the target classification task.

A different experiment for enhancing the geometric invariance on CNN activations was also presented. Gong *et al*. [12] proposed a method to exploit multi-scale CNN activations in order to achieve geometric invariance characteristic while improving recognition accuracy. They extracted dense local patches at three different scales and fed each local patch into a pre-trained CNN. The CNN activations are aggregated at finer scales via VLAD encoding which was introduced in [15], and then the encoded activations are concatenated as a single vector to obtain the final representation.

In this paper, we introduce a *multi-scale pyramid pooling* to improve the discriminative power of CNN activations robust to geometric variations. A pipeline of the proposed method is illustrated in Figure 1. Similar to [12], we also utilize multi-scale CNN activations, but present a different pooling method that shows better performance in our experiments. We extract abundant amount of multi-scale local activations from a CNN, and aggregate them using the state-of-the-art Fisher kernel [27, 26] with a simple but important scale-wise normalization, so called *multi-scale pyramid pooling*. Our proposal demonstrates substantial improvements on both scene and object classification tasks compared to the previous representations including a single activation, the average pooling [29, 4], and the VLAD of activations [12]. Also, we demonstrate object confidence maps which is useful for object detection/localization though only category-level labels without specific object bounding boxes are used in training.

According to our empirical observations, replacing a VLAD kernel with a Fisher kernel does not present significant impact, however it shows meaningful performance improvements when our pooling mechanism that takes an average pooling after scale-wise normalization is applied. It implies that the performance improvement of our representation does not come just from the superiority of Fisher kernel but from the careful consideration of neural activation's property dependent on scales.

2. Multi-scale Pyramid Pooling

In this section, we first review the Fisher kernel framework and then introduce a *multi-scale pyramid pooling* which adds a Fisher kernel based pooling layer on top of a pre-trained CNN.

2.1. Fisher Kernel Review

The Fisher kernel framework on a visual vocabulary is proposed by Perronnin *et al*. in [26]. It extends the conventional Bag-of-Words model to a probabilistic generative model. It models the distribution of low-level descriptors using a Gaussian Mixture Model (GMM) and represents an image by considering the gradient with respect to the model parameters. Although the number of local descriptors varies across images, the consequent Fisher vector has a fixed-length, therefore it is possible to use discriminative classifiers such as a linear SVM.

Let $\mathbf{x}$ denote a d-dimensional local descriptor and $\mathbf{G}_\lambda = \{\mathbf{g}_k, k = 1...K\}$ denote a pre-trained GMM with K Gaussians where $\lambda = \{\omega_k, \mu_k, \sigma_k, k = 1...K\}$. For each visual word $\mathbf{g}_k$, two gradient vectors, $\mathcal{G}_{\mu_k} \in \Re^d$ and $\mathcal{G}_{\sigma_k} \in \Re^d$, are computed by aggregating the gradients of the local descriptors extracted from an image with respect to the mean and the standard deviation of the k^{th} Gaussian. Then, the final image representation, *Fisher vector*, is obtained by concatenating all the gradient vectors. Accordingly, the Fisher kernel framework represents an image with a $2Kd$-dimensional Fisher vector $\mathcal{G} \in \Re^{2Kd}$.

Intuitively, a Fisher vector includes the information about directions of model parameters to best fit the local de-

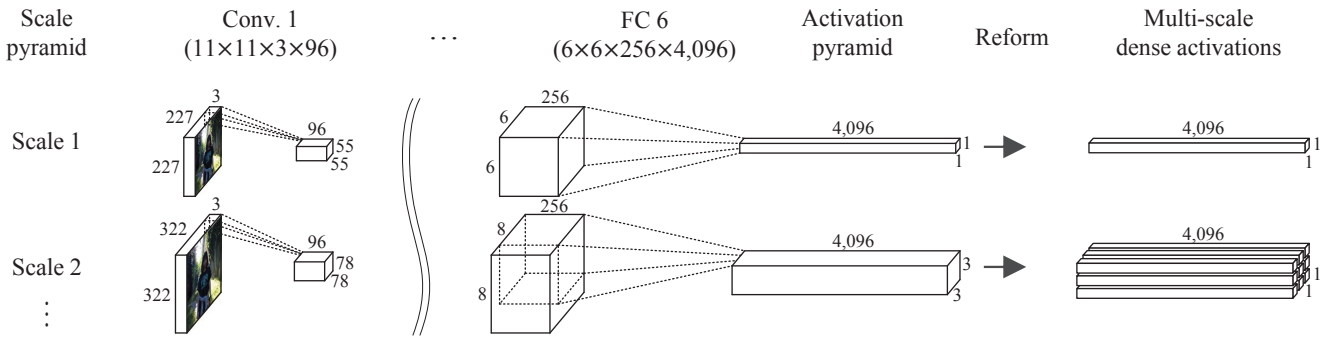

Figure 2. Obtaining multi-scale local activations densely from a pre-trained CNN. In this figure, the target layer is the first fully connected layer (FC6). Because FC6 can be equally implemented by a convolutional layer containing 4,096 filters of 6×6×256 size, we can obtain an activation map where spatial ordering of local descriptors is conserved. A single pre-trained CNN is shared for all scales. Please refer to [33].

scriptors of an image to the GMM. The fisher kernel framework is further improved in [27] by the additional two-stage normalizations: power-normalization with the factor of 0.5 followed by ℓ_2-normalization. Refer to [27] for the theoretical proofs and details.

2.2. Dense CNN Activations

To obtain multi-scale activations from a CNN without modification, previous approach cropped local patches and fed the patches into a network after resizing the patches to the fixed size of CNN input. However, when we extract multi-scale local activations densely, the approach is quite inefficient since many redundant operations are performed in convolutional layers for overlapped regions.

To extract dense CNN activations without redundant operations, we replace the fully connected layers of an existing CNN with equivalent multiple convolution filters along spatial axises, as Tompson *et al*. did in [33]. When an image larger than the fixed size is fed, the modified network outputs multiple activation vectors where each vector is CNN activations from the corresponding local patch. The procedure is illustrated in Fig. 2. With this method, thousands of dense local activations (4,410 per image) from multiple scale levels are extracted in a reasonable extraction time (0.46 seconds per image on a server with a CPU of 2.6GHz Intel Xeon and a GPU of GTX TITAN Black).

2.3. Multi-scale Pyramid Pooling (MPP)

For representing an image, we first generate a scale pyramid for the input image where the minimum scale image has a fixed size of a CNN and each scale image has two times larger resolution than the previous scale image. We feed all the scaled images into a pre-trained CNN and extract dense CNN activation vectors. Then, all the activation vectors are merged into a single vector by our multi-scale pyramid pooling.

If we consider each activation vector as a local descriptor, it is straightforward to aggregate all the local activations into a Fisher vector as explained in Sec. 2.1. However, CNN activations have different scale properties compared to SIFT-like local descriptors, as will be explained in Sec. 3. To adopt the Fisher kernel suitable to CNN activation characteristics, we introduce adding a *multi-scale pyramid pooling layer* on top of the modified CNN as follows.

Given a scale pyramid S containing N scaled image and local activation vectors $\mathbf{x}_s$ extracted from each scale $s \in S$, we first apply PCA to reduce the dimension of activation vectors and obtain $\mathbf{x}'_s$. Then, we aggregate the local activation vectors $\mathbf{x}'_s$ of each scale s to each Fisher vector $\mathcal{G}^s$. After Fisher encoding, we have N Fisher vectors and they are merged into one global vector by average pooling after ℓ_2-normalization as

$$\mathcal{G}^S = \frac{1}{N}\sum_{s\in S}\frac{\mathcal{G}^s}{\|\mathcal{G}^s\|_2} \quad \text{s.t.} \quad \mathcal{G}^s = \frac{1}{|\mathbf{x}'_s|}\sum_{x\in\mathbf{x}'_s}\nabla_\lambda \log \mathbf{G}_\lambda(x), \tag{1}$$

where $|\cdot|$ denotes the cardinality of a set. We use an average pooling since it is a natural pooling scheme for Fisher kernel rather than vector concatenation. Following the Improved Fisher Kernel framework [27], we finally apply power normalization to tackle burstiness [19, 14] and ℓ_2-normalization to the Fisher vector $\mathcal{G}^S$. The overall pipeline of MPP is illustrated in Figure 1.

3. Analysis of Multi-scale CNN Activations

We compare scale characteristics between traditional local features and CNN activations. It tells us that it is not suitable to directly adopt a Fisher kernel framework to multi-scale local CNN activations for representing an image. To investigate the best way for aggregating the CNN activations into a global representation, we perform empirical studies and conclude that applying scale-wise normalization of Fisher vectors is very important.

A naive way to obtain a Fisher vector $\mathcal{G}'^S$ given multi-scale local activations $X = \{x \in \mathbf{x}_s, s \in S\}$ is to aggregate them as,

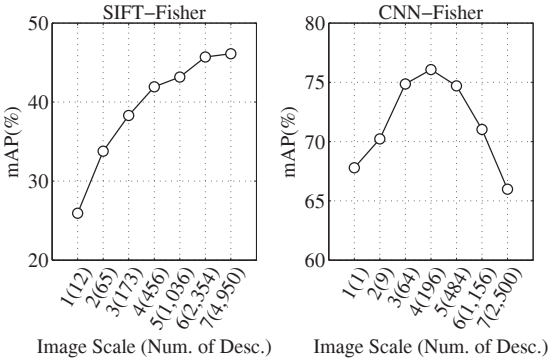

Figure 3. Classification performance of SIFT-Fisher and CNN-Fisher according to image scale on PASCAL VOC 2007. The tick labels of the horizontal axis denote image scales and their average number of local descriptors.

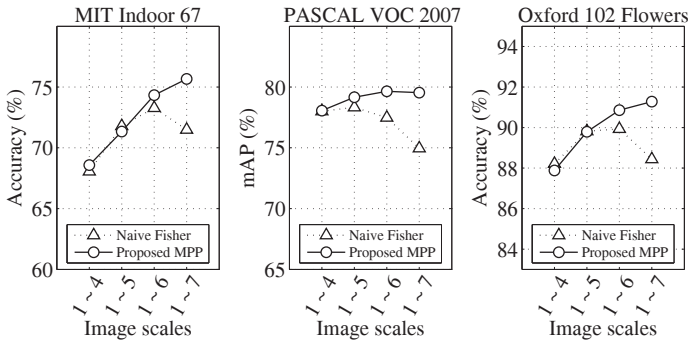

Figure 4. Classification performance of our *multi-scale pyramid pooling* in Eq. (1) and the naive Fisher pooling in Eq. (2). The tick labels of the horizontal axis scale levels in a scale pyramid.

$$\mathcal{G}'^{S} = \frac{1}{|X|} \sum_{s \in S} \sum_{x \in \mathbf{x}_s} \nabla_\lambda \log \mathbf{G}_\lambda(x). \tag{2}$$

Here, every multi-scale local activation vector is pooled to one Fisher vector with an equal weight of $1/|X|$.

To better combine a Fisher kernel with mid-level neural activations, the property of CNN activations according to patch scale should be taken in consideration. In the traditional use of Fisher kernel on visual classification tasks, the hand-designed local descriptors such as SIFT [22] have been often densely computed in multi-scale. This local descriptor encodes low-level gradient information within an local region and captures detailed textures or shapes within a small region rather than the global structure within a larger region. In contrast, a mid-level neural activation extracted from a higher layer of CNNs (e.g. FC6 or FC7 of [20]) represents higher level structure information which is closer to class posteriors. As shown in the CNN visualization proposed by Zeiler and Fergus in [37], image regions strongly activated by a certain CNN filter of the fifth layer usually capture a category-level entire object.

To figure out the different scale properties between the Fisher vector of traditional SIFT (SIFT-Fisher) and that of neural activation from FC7 (CNN-Fisher), we conduct an empirical analysis with scale-wise classification scores on PASCAL VOC 2007 [9]. For the analysis, we first diversify dataset into seven different scale levels from the smallest scale of 227×227 resolution to the biggest scale of $1,816 \times 1,816$ resolution and extract both dense SIFT descriptors and local activation vectors in the seventh layer (FC7) of our CNN. Then, we follow the standard framework to encode Fisher vectors and to train an independent linear SVM for each scale, respectively.

In Fig. 3, we show the results of classification performances using SIFT-Fisher and CNN-Fisher according to scale. The figure demonstrates clear contrast between SIFT-Fisher and CNN-Fisher. CNN-Fisher performs worst at the largest image scale since local activations come from small image regions in an original image, while SIFT-Fisher performs best at the same scale since SIFT properly captures low-level contents within such small regions. If we aggregate the CNN activations of all scales into one Fisher vector by Eq. (2), the poorly performing 2,500 activations will have dominant influence with the large weight of 2,500/4,410 in the image representation.

One possible strategy for aggregating multi-scale CNN activations is to choose activations of a set of scales relatively performing well. However, the selection of good scales is dependent on dataset and the activations from the large image scale can also contribute to geometric invariance property if we balance the influence of each scale. We empirically examined various combinations of pooling as will be shown in Sec. 4 and we found that scale-wise Fisher vector normalization followed by an simple average pooling is effective to balance the influence.

We perform an experiment to compare our pooling method in Eq. (1) to the naive Fisher pooling in Eq. (2). In the experiment, we apply both of two pooling methods with five different numbers of scales and perform classification on PASCAL VOC 2007. Despite the simplicity of our multi-scale pyramid pooling, it demonstrates superior performances as depicted in Fig. 4. The performance of the naive Fisher kernel pooling in Eq. (2) deteriorates rapidly when finer scale levels are involved. This is because indistinctive neural activations from finer scale levels become dominant in forming a Fisher vector. Our representation, however, exhibits stable performance that the accuracy is constantly increasing and finally being saturated. It verifies that our pooling method aggregates multi-scale CNN activations effectively.

4. Experiments

4.1. Datasets

To evaluate our proposal as a generic image representation, we conduct three different visual recognition tasks with following datasets.

MIT Indoor 67 [28] is used for a scene classification task. The dataset contains 15,620 images with 67 indoor scene classes in total. It is a challenging dataset because many indoor classes are characterized by the objects they contain (e.g. different type of stores) rather than their spatial properties. The performance is measured with top-1 accuracy.

PASCAL VOC 2007 [9] is used for an object classification task. It consists of 9,963 images of 20 object classes in total. The task is quite difficult since the scales of the objects fluctuate and multiple objects of different classes are often contained in the same image. The performance is measured with (11-points interpolated) mean average precision.

Oxford 102 Flowers [24] is used for a fine-grained object classification task, which distinguishes the sub-classes of the same object class. This dataset consists of 8,189 images with 102 flower classes. Each class consists of various numbers of images from 20 to 258. The performance is measured with top-1 accuracy.

4.2. Pre-trained CNNs

We use two CNNs pre-trained on the ILSVRC'12 dataset [6] to extract multi-scale local activations. One is the Caffe reference model [16] composed of five convolutional layers and three fully connected layers. This model performed 19.6% top-5 error when a single center-crop of each validation image are used for evaluation on the ILSVRC'12 dataset. Henceforth, we denote this model by "Alex" since it is nearly the same architecture of Krizhevsky *et al.*'s CNN [20].

The other one is Chatfield *et al.*'s CNN-S model [4] ("CNNS", henceforth). This model, a simplified version of the OverFeat [30], is also composed of five convolutional layers (three in [30]) and three fully connected layers. It shows 15.5% top-5 error on the ILSVRC'12 dataset with the same center-crop. Compared to Alex, it uses 7×7 smaller filters but dense stride of 2 in the first convolutional layer.

Our experiments are conducted mostly with the Alex by default. The CNNS is used only for the PASCAL VOC 2007 dataset to compare our method with [4], which demonstrates excellent performance with the CNNS. Both of the two pre-trained models are available online [35].

4.3. Implementation Details

We use an image pyramid of seven scales by default since the seven scales can cover large enough scale variations and performance in all datasets as shown in Fig. 4.

The overall procedure of our image representation is as follows. Given an image, we make an image pyramid containing seven scaled images. Each image in the pyramid has twice resolution than the previous scale starting from the standard size defined in each CNN (e.g. 227×227 for Alex). We then feed each scale image to the CNN and obtain 4,410 vectors of 4,096 dimensional dense CNN activations from the seventh layer. The dimensionality of each activation vector is reduced to 128 by PCA where a projection is trained with 256,000 activation vectors sampled from training images. A visual vocabulary (GMM of 256 Gaussian distributions) is also trained with the same samples. Consequently, one 65,536 dimensional Fisher vector is computed by Eq. (1), and further power- and ℓ_2-normalization follow. One-versus-rest linear SVMs with a quadratic regularizer and a hinge loss are trained finally.

Our system is mostly implemented using open source libraries including VLFeat [34] for a Fisher kernel framework and MatConvNet [35] for CNNs.

4.4. Results and Analysis

We perform comprehensive experiments to compare various methods on the three recognition tasks. We first show the performance of our method and baseline methods. Then, we compare our result with state-of-the-art methods for each dataset. For simplicity, we use a notation protocol "A(B)" where A denotes a pooling method and B denotes descriptors to be pooled by A. The notations are summarized in Table 1.

We compare our method with several baseline methods. The baseline methods include intermediate CNN activations from a pre-trained CNN with a standard input, an average pooling with multiple jittered images, and modified versions of our method. The comparison results for each dataset are summarized in Table 2(a), 3(a), 4(a). As expected, the most basic representation, Alex-FC7, performs the worst for all datasets. The average pooling in AP10 and AP50 improves the performance +1.39%$\sim$+3%, however the improvement is bounded regardless of the number of data augmentation. The other two baseline methods (MPP w/o SN and CSF) exploit multi-scale CNN activations and they show better results than single-scale representations. Compared to the AP10, the performance gains from multi-scale activations exceed +10%, +1%, and +5% for each dataset. It shows that image representation based on CNN activations can be enriched by utilizing multi-scale local activations.

Even though baseline methods exploiting multi-scale CNN activations show substantial improvements compared

to the single-scale baselines, we can also verify that handling multi-scale activations is important for further improvement. Compared to the naive Fisher kernel pooling (NFK) in Eq. (2), our MPP achieves an extra but significant performance gain of +4.18%, +4.58%, and 2.84% for each dataset. Instead of pooling multi-scale activations as our MPP, concatenating encoded Fisher vectors can be another option as done in Gong *et al.*'s method [12]. The concatenation (CSF) also improves the performance, however the CSF without an additional dimension reduction raises the dimensionality proportional to the number of scales and the MPP still outperforms the CSF for all datasets. The comprehensive test with various pooling strategies so far shows that the proposed image representation can be used as a primary image representation in wide visual recognition tasks.

We also apply the spatial pyramid (SP) kernel [21] to our representation. We construct a spatial pyramid into four sub-regions (whole, top, middle, bottom) and it increases the dimensionality of our representation four times. The results are unequable but the differences are marginal for all datasets. This result is not surprising because the rich activations from smaller image scales already cover the global layout. It makes the SP kernel redundant.

In Table 2(b), we compare our result with various state-of-the-art methods on Indoor 67. Similar to ours, Gong *et al*. [12] proposed a pooling method for multi-scale CNN activations. They performed VLAD pooling at each scale and concatenated them. Compared to [12], our representation largely outperforms the method with a gain of +7.07%. The performance gap possibly comes from 1) the large number of scales, 2) the superiority of the Fisher kernel, and 3) the details of pooling strategy. While they use only three scales, we extract seven-scale activations with a quite efficient way (Fig. 2). *Though adding local activations from very finer scales such as 6 or 7 in a naive way may harm the performance, it actually contribute to a better invariance property by the proposed MPP*. In addition, as our experiment of the "CSF" was shown, the MPP is more suitable for aggregating multi-scale activations than the concatenation. It implies that our better performance does not just come from the superior Fisher kernel, but from the better handling of multi-scale neural activations.

The record holder in the Indoor 67 dataset has been Zuo *et al*. [41] who combined the Alex-FC6 and their complementary features so called DSFL. DSFL learns discriminative and shareable filters with a target dataset. When we stack an additional MPP at the Pool5 layer, we (77.76%) already surpass the records with a pre-trained Alex only. We also stack DSFL feature[1] over our representation and the result shows 80.78%. It shows that our representation is also improved by combining complementary features.

[1]Pre-computed DSFL vectors for the MIT Indoor 67 dataset are provided by the authors.

The results on VOC 2007 is summarized in Table 3(b). There are two methods ([25] and [29]) that use the same Alex network. Razavian *et al*. [29] performed target data augmentation and Oquab *et al*. [25] used a multi-layer perceptron (MLP) instead of a linear SVM with ground truth bounding boxes. Our representation outperforms the two methods using the pre-trained Alex without data augmentation or the use of bounding box annotations. The gains are +1.84% and +2.34% respectively.

Most of recent state-of-the-art methods are adopting better CNNs for the source task (i.e. ImageNet classification) or the target task, such as Spatial Pyramid Pooling (SPP) network [13], Multi-label CNN [36] and the CNNS [4]. Our basic MPP(Alex-FC7) demonstrates slightly lower precisions (79.54%) compared to them, however we use the basic Alex CNN without fine-tuning on VOC 2007. When our representation is equipped with the superior CNNS [4], which is not fine-tuned on VOC 2007, our representation (81.40%) reaches nearly stat-of-the-art performance. When we also use the same target data augmentation [4] in SVM training, we achieved a new state-of-the-art score of 83.20% without fine-tuning. This result beats [4] by +3.5% in the same setup, and it is even higher than the previous state-of-the-art score of [4], which uses fine-tuned CNNS.

Table 5 shows per-class performances on VOC 2007. Compared to state-of-the-art methods, our method performs best in 6 classes. It is interesting that the 6 classes include "bottle", "pottedplant", and "tvmonitor", which are the relatively small objects in the VOC 2007 dataset. The results clearly demonstrates the benefit of our MPP that aggregates activations from very finer-scales as well, which are prone to harm the performance if it is handled inappropriately.

Table 4(b) shows the classification performances on 102 Flowers. Our method (91.28%) outperforms the previous state-of-the-art method [18] (90.20%).

4.5. Weakly-Supervised Object Confidence Map

One interesting feature of our method is that we can present object confidence maps for object classification tasks, though we train the SVM classifiers *without bounding box annotation but only with class-level labels*. To recover confidence maps, we trace how much weight is given to each local patch and accumulate all the weights in the spatial domain. To trace the weight of each patch, we compute our final representation per patch using the corresponding single activation vector only and compute the score from the pre-trained SVM classifiers we used for object classification.

Fig. 5 shows several examples of object confidence map on the VOC 2007 test images. In the figures, we can verify our image representation encodes the discriminative image patches well, despite large within-class variations as well as substantial geometric changes. As we discussed in Sec. 4.4,

Method	Description
CNN-FC7	A standard activation vector from FC7 of a CNN with a center-crop of a 256×256 size input image.
AP10(CNN-FC7)	Average pooling of a 5 crops and their flips, given a 256×256 size input image.
AP50(CNN-FC7)	Average pooling of a 25 crops and their flips, given a 256×256 size input image.
NFK(CNN-FC7)	Naive Fisher kernel pooling without scale-wise vector normalization, given a multi-scale image pyramid.
CSF(CNN-FC7)	Concatenation of scale-wise normalized Fisher vectors, given a multi-scale image pyramid.
MPP(CNN-FC7)	The proposed representation, given a multi-scale image pyramid.

Table 1. Summary of our notation protocol. Consequent image representations by the listed methods are finally ℓ_2-normalized.

Method	FT	plane	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	motor	person	plant	sheep	sofa	train	tv	mAP
Alex-FC7	No.	85.0	79.7	82.8	80.4	39.7	69.3	82.9	81.7	58.7	57.8	68.5	75.9	83.0	72.5	90.6	51.7	71.1	60.8	85.0	70.5	72.4
AP10(Alex-FC7)	No.	85.7	80.8	83.3	80.7	40.4	71.5	83.8	82.7	60.7	60.5	70.6	79.0	84.5	75.0	91.3	53.4	70.1	62.6	86.5	72.1	73.7
MPP(Alex-FC7)	No.	90.2	86.9	86.6	84.4	54.0	80.0	87.9	86.0	63.4	72.2	75.7	83.1	87.8	83.9	93.0	**64.8**	75.8	69.6	89.9	75.9	79.5
MPP(CNNS-FC7)	No.	90.2	88.6	89.0	84.7	**58.2**	**82.8**	88.1	89.0	**64.9**	77.0	**78.4**	86.9	89.2	86.7	92.8	61.2	81.3	**70.0**	89.8	**79.3**	81.4
Perronnin *et al*. [27] 10'	No.	75.7	64.8	52.8	70.6	30.0	64.1	77.5	55.5	55.6	41.8	56.3	41.7	76.3	64.4	82.7	28.3	39.7	56.6	79.7	51.5	58.3
Razavian *et al*. [29] '14	No.	90.1	84.4	86.5	84.1	48.4	73.4	86.7	85.4	61.3	67.6	69.6	84.0	85.4	80.0	92.0	56.9	76.7	67.3	89.1	74.9	77.2
Oquab *et al*. [25] '14	No.	88.5	81.5	87.9	82.0	47.5	75.5	90.1	87.2	61.6	75.7	67.3	85.5	83.5	80.0	95.6	60.8	76.8	58.0	90.4	77.9	77.7
Wei *et al*. [36] '14	Yes.	95.1	90.1	**92.8**	**89.9**	51.5	80.0	**91.7**	91.6	57.7	**77.8**	70.9	89.3	89.3	85.2	93.0	64.0	**85.7**	62.7	94.4	78.3	81.5
Chatfield*et al*. [4] '14	Yes.	**95.3**	**90.4**	92.5	89.6	54.4	81.9	91.5	**91.9**	64.1	76.3	74.9	**89.7**	**92.2**	**86.9**	**95.2**	60.7	82.9	68.0	**95.5**	74.4	**82.4**

Table 5. Per-class classification performances on PASCAL VOC 2007.

Method	Description	CNN	Acc.
Baseline	Alex-FC7	Yes.	57.91
Baseline	AP10(Alex-FC7)	Yes.	60.90
Baseline	AP50(Alex-FC7)	Yes.	60.37
Baseline	NFK(Alex-FC7)	Yes.	71.49
Baseline	CSF(Alex-FC7)	Yes.	72.24
Ours	MPP(Alex-FC7)	Yes.	75.67
Ours	MPP(Alex-FC7)+SP	Yes.	**75.97**
Ours	MPP(Alex-FC7,Pool5)	Yes.	77.56
Ours	MPP(Alex-FC7)+DSFL[41]	Yes.	**80.78**

(a) baselines and our methods.

Method	Description	CNN	Acc.
Singh *et al*. [31] '12	Part+GIST+DPM+SP	No.	49.40
Juneja *et al*. [17] '13	IFK+Bag-of-Parts	No.	63.18
Doersch *et al*. [7] '13	IFK+MidlevelRepresent.	No.	66.87
Zuo *et al*. [41] '14	DSFL	No.	52.24
Zuo *et al*. [41] '14	DSFL+Alex-FC6	Yes.	**76.23**
Zhou *et al*. [40] '14	Alex-FC7	Yes.	68.24
Zhou *et al*. [40] '14	Alex-FC7	Yes.	70.80
Razavian *et al*. [29] '14	AP(Alex)+PT+TA.	Yes.	69.00
Gong *et al*. [12] '14	VLAD Concat.(Alex-FC7)	Yes.	68.90

(b) state-of-the-art methods on MIT Indoor 67.

Table 2. Classification performances on MIT Indoor 67. (SP: Spatial Pyramid, DPM: Deformable Part-based Model, PT: Power Transform, IFK: Improved Fisher Kernel, DSFL: Discriminative and Shareable Feature Learning.)

Method	Description	FT	BB	CNN	mAP
Baseline	Alex-FC7	No.	No.	Yes.	72.36
Baseline	AP10(Alex-FC7)	No.	No.	Yes.	73.75
Baseline	AP50(Alex-FC7)	No.	No.	Yes.	73.60
Baseline	NFK(Alex-FC7)	No.	No.	Yes.	74.96
Baseline	CSF(Alex-FC7)	No.	No.	Yes.	78.46
Ours	MPP(Alex-FC7)	No.	No.	Yes.	**79.54**
Ours	MPP(Alex-FC7)+SP	No.	No.	Yes.	79.29
Ours	MPP(CNNS-FC7)	No.	No.	Yes.	81.40
Ours	MPP(CNNS-FC7)+TA	No.	No.	Yes.	**83.20**

(a) Baselines and our methods.

Method	Description	FT	BB	CNN	mAP
Perronnin *et al*. [27] 10'	IFK(SIFT+color)	No.	No.	No.	60.3%
He *et al*. [13] '14	SPPNET-FC7	No.	No.	Yes.	80.10%
Wei *et al*. [36] '14	Multi-label CNN	Yes.	No.	Yes.	81.50%
Razavian *et al*. [29] '14	AP(Alex)+PT+TA	No.	No.	Yes.	77.20%
Oquab *et al*. [25] '14	Alex-FC7+MLP	No.	Yes.	Yes.	77.70%
Chatfield *et al*. [4] '14	AP(CNNS-FC7)+TA	No.	No.	Yes.	79.74%
Chatfield *et al*. [4] '14	AP(CNNS-FC7)+TA	Yes.	No.	Yes.	**82.42**%

(b) state-of-the-art methods on PASCAL VOC 2007 classification.

Table 3. Classification performances on PASCAL VOC 2007 classification. "FT" represents fine-tuning of a pre-trained CNN on VOC2007 and "BB" denotes the use of ground truth object bounding boxes in training. (SP: Spatial Pyramid, IFK: Improved Fisher Kernel, SPPNET: Spatial Pyramid Pooling Network, PT: Power Transform, TA: Target data Augmentation in SVM training, MLP: Multilayer Perceptron.)

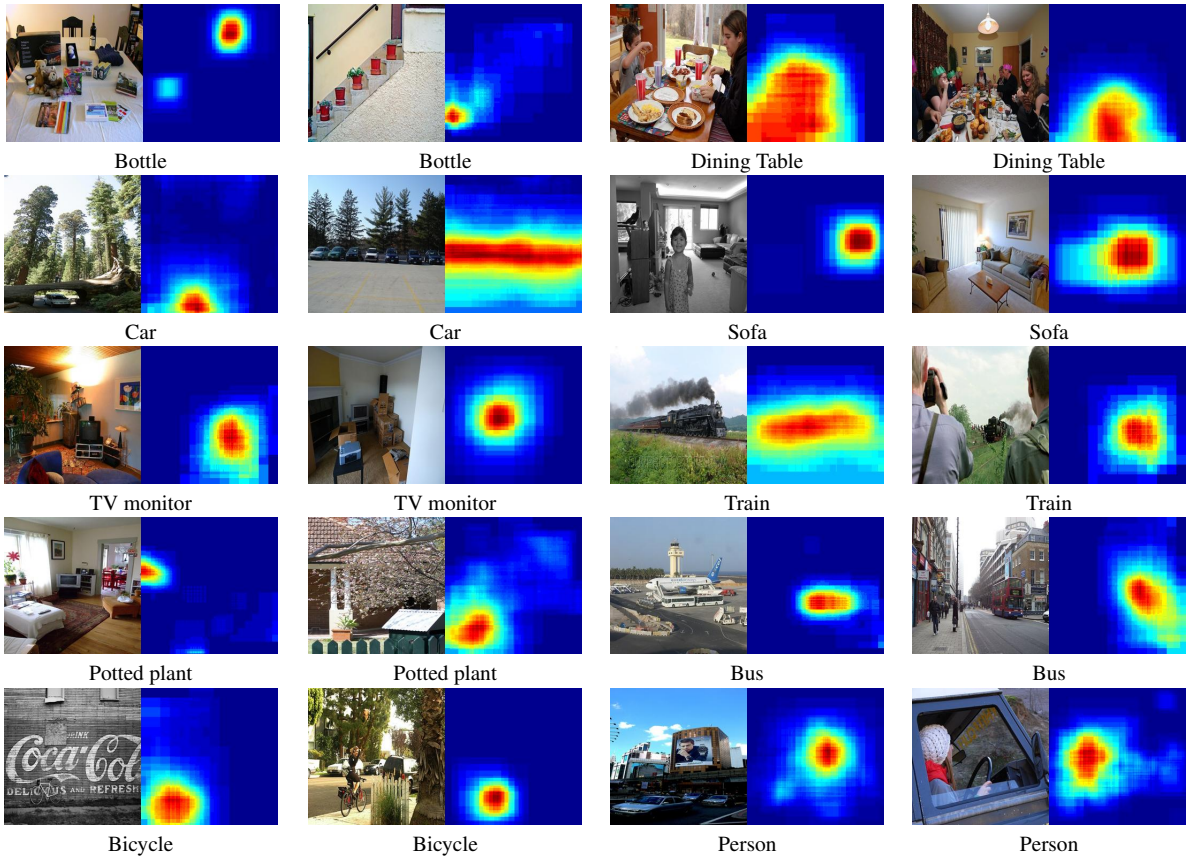

Figure 5. Examples of object confidence maps obtained by our image representation on the PASCAL VOC 2007. All examples are test images, not training images. Note, the object bonding box annotations are not used for training.

Method	Description	Seg.	CNN	Acc.
Baseline	Alex-FC7	No.	Yes.	81.43
Baseline	AP10(Alex-FC7)	No.	Yes.	83.40
Baseline	AP50(Alex-FC7)	No.	Yes.	83.56
Baseline	NFK(Alex-FC7)	No.	Yes.	88.44
Baseline	CSF(Alex-FC7)	No.	Yes.	89.35
Ours	MPP(Alex-FC7)	No.	Yes.	**91.28**
Ours	MPP(Alex-FC7)+SP	No.	Yes.	90.05

(a) baselines and our methods.

Method	Description	Seg.	CNN	Acc.
Nilsback and Zisserman [24] '08	Multple kernel learning	Yes.	No.	77.70
Angelova and Zhu [1] '13	Seg+DenseHoG +LLC+MaxPooling	Yes.	No.	80.70
Koniusz *et al*. [18] '13	Bag-of-words + HOP	No.	No.	**90.2**
Murray and Perronnin [23] '14	GMP of FK(SIFT+color)	No.	No.	81.50
Fernando *et al*. [10] '14	Bag-of-FLH	Yes.	No.	72.70
Razavian *et al*. [29] '14	AP(Alex)+PT+TA	No.	Yes.	86.8

(b) state-of-the-art methods on Oxford 102 Flowers.

Table 4. Classification performances on Oxford 102 Flowers. "Seg." denotes the use of ground truth segmentations in training. (HOP: Higher-order Occurrence Pooling.)

the images containing small-size objects also present the accurate confidence maps. These maps may further be utilized as an considerable cue for object detection/localization and also be useful for analyzing image representation.

5. Discussion

We have proposed the multi-scale pyramid pooling for better use of neural activations from a pre-trained CNN. There are two conclusions we can derive through our study. One is that we should take the scale characteristic of neural activations into consideration for the successful combination of a Fisher kernel and a CNN. The activations become uninformative as a patch size becomes smaller, however they can contribute to better scale invariance when they meet a simple scale-wise normalization. The other is that reasonable object-level confidence maps can be obtained from our image representation even though only class-level labels are given, which can be further applied to object detection or localization tasks. In the comprehensive experiments on three different recognition tasks, the results suggest that our proposal can be a primary image representation in wide visual recognition tasks.

6. Acknowledgement

This work was supported by the Technology Innovation Program (No. 10048320), funded by the Ministry of Trade, Industry & Energy (MI, Korea).

References

[1] A. Angelova and S. Zhu. Efficient object detection and segmentation for fine-grained recognition. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 811–818. IEEE, 2013. 8

[2] A. Babenko, A. Slesarev, A. Chigorin, and V. Lempitsky. Neural codes for image retrieval. In *Proceedings of European Conference on Computer Vision (ECCV)*, 2014. 1

[3] A. Barbu, D. P. Barrett, W. Chen, S. Narayanaswamy, C. Xiong, J. J. Corso, C. D. Fellbaum, C. Hanson, S. J. Hanson, S. Hélie, E. Malaia, B. A. Pearlmutter, J. M. Siskind, T. M. Talavage, and R. B. Wilbur. Seeing is worse than believing: Reading people's minds better than computer-vision methods recognize actions. In *Proceedings of European Conference on Computer Vision (ECCV)*, 2014. 1

[4] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman. Return of the devil in the details: Delving deep into convolutional nets. In *Proceedings of British Machine Vision Conference (BMVC)*, 2014. 1, 2, 5, 6, 7

[5] Y. L. Cun, B. Boser, J. S. Denker, D. Henderson, R. Howard, W. Hubbard, and L. Jackel. Backpropagation applied to handwritten zip code recognition. *Neural Computation*, 1:541–551, 1989. 1

[6] J. Deng, W. Dong, R. Socher, L. J. Li, K. Li, and L. F. Fei. Imagenet: A large-scale hierarchical image database. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009. 1, 5

[7] C. Doersch, A. Gupta, and A. A. Efros. Mid-level visual element discovery as discriminative mode seeking. In *Advances in Neural Information Processing Systems (NIPS)*, 2013. 7

[8] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell. DeCAF: A deep convolutional activation feature for generic visual recognition. In *Computing Research Repository (CoRR)*, 2013. 1

[9] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results. http://www.pascal-network.org/challenges/VOC/voc2007/workshop/index.html. 4, 5

[10] B. Fernando, É. Fromont, and T. Tuytelaars. Mining mid-level features for image classification. *International Journal on Computer Vision (IJCV)*, 108(3):186–203, 2014. 8

[11] R. B. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014. 1

[12] Y. Gong, L. Wang, R. Guo, and S. Lazebnik. Multi-scale orderless pooling of deep convolutional activations features. In *Proceedings of European Conference on Computer Vision (ECCV)*, 2014. 1, 2, 6, 7

[13] K. He, X. Zhang, S. Ren, and J. Sun. Spatial pyramid pooling in deep convolutional networks for visual recognition. In *Proceedings of European Conference on Computer Vision (ECCV)*, 2014. 1, 6, 7

[14] H. Jegou, M. Douze, and C. Schmid. On the burstiness of visual elements. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1169–1176, 2009. 3

[15] H. Jegou, M. Douze, C. Schmid, and P. Pérez. Aggregating local descriptors into a compact image representation. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2010. 1, 2

[16] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. *arXiv preprint arXiv:1408.5093*, 2014. 5

[17] M. Juneja, A. Vedaldi, C. V. Jawahar, and A. Zisserman. Blocks that shout: Distinctive parts for scene classification. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013. 7

[18] P. Koniusz, F. Yan, P.-H. Gosselin, and K. Mikolajczyk. Higher-order occurrence pooling on mid- and low-level features: Visual concept detection, 2013. 6, 8

[19] P. Koniusz, F. Yan, and K. Mikolajczyk. Comparison of mid-level feature coding approaches and pooling strategies in visual concept detection. *Computer Vision and Image Understanding*, 117(5):479–492, 2013. 3

[20] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems (NIPS)*, 2013. 1, 4, 5

[21] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages II: 2169–2178, 2006. 6

[22] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal on Computer Vision (IJCV)*, 60(2):91–110, 2004. 1, 4

[23] N. Murray and F. Perronnin. Generalized max pooling. *Computing Research Repository (CoRR)*, abs/1406.0312, 2014. 8

[24] M.-E. Nilsback and A. Zisserman. Automated flower classification over a large number of classes. In *Proceedings of the Indian Conference on Computer Vision, Graphics and Image Processing*, 2008. 5, 8

[25] M. Oquab, L. Bottou, I. Laptev, and J. Sivic. Learning and transferring mid-level image representations using convolutional neural networks. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014. 1, 6, 7

[26] F. Perronnin and C. R. Dance. Fisher kernels on visual vocabularies for image categorization. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2007. 1, 2

[27] F. Perronnin, J. Sánchez, and T. Mensink. Improving the fisher kernel for large-scale image classification. In *Proceedings of European Conference on Computer Vision (ECCV)*, 2010. 1, 2, 3, 7

[28] A. Quattoni and A.Torralba. Recognizing indoor scenes. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009. 5

[29] A. S. Razavian, H. Azizpour, J. Sullivan, and S. Carlsson. CNN features off-the-shelf: an astounding baseline for recognition. In *Computing Research Repository (CoRR)*, 2014. 1, 2, 6, 7, 8

[30] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun. Overfeat: Integrated recognition, localization and detection using convolutional networks. In *Proceedings of International Conference on Learning Representations (ICLR)*, 2014. 5

[31] S. Singh, A. Gupta, and A. A. Efros. Unsupervised discovery of mid-level discriminative patches. In *Proceedings of European Conference on Computer Vision (ECCV)*, 2012. 7

[32] J. Sivic and A. Zisserman. Video google: A text retrieval approach to object matching in videos. In *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, 2003. 1

[33] J. Tompson, A. Jain, Y. LeCun, and C. Bregler. Joint training of a convolutional network and a graphical model for human pose estimation. *CoRR*, Sept. 17 2014. 2, 3

[34] A. Vedaldi and B. Fulkerson. VLFeat: An open and portable library of computer vision algorithms. http://www.vlfeat.org/, 2008. 5

[35] A. Vedaldi and K. Lenc. MatConvNet: Convolutional neural networks for matlab. http://www.vlfeat.org/matconvnet/, 2014. 5

[36] Y. Wei, W. Xia, J. Huang, B. Ni, J. Dong, Y. Zhao, and S. Yan. Cnn: Single-label to multi-label. In *Computing Research Repository (CoRR)*, 2014. 6, 7

[37] M. D. Zeiler and R. Fergus. Visualizing and understanding convolutional networks. In *Proceedings of European Conference on Computer Vision (ECCV)*, 2014. 4

[38] N. Zhang, J. Donahue, R. B. Girshick, and T. Darrell. Part-based R-CNNs for fine-grained category detection. In *Proceedings of European Conference on Computer Vision (ECCV)*, 2014. 1

[39] N. Zhang, M. Paluri, M. Ranzato, T. Darrell, and L. D. Bourdev. PANDA: Pose aligned networks for deep attribute modeling. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014. 1

[40] B. Zhou, A. Lapedriza, J. Xiao, A. Torralba, and A. Oliva. Learning deep features for scene recognition using places database. In *Advances in Neural Information Processing Systems (NIPS)*, 2014. 1, 7

[41] Z. Zuo, G. Wang, B. Shuai, L. Zhao, Q. Yang, and X. Jiang. Learning discriminative and shareable features for scene classification. In *Proceedings of European Conference on Computer Vision (ECCV)*, 2014. 6, 7