Multi-scale Orderless Pooling of Deep Convolutional Activation Features

Yunchao Gong\textsuperscript{1}, Liwei Wang\textsuperscript{2}, Ruiqi Guo\textsuperscript{2}, and Svetlana Lazebnik\textsuperscript{2}

\textsuperscript{1}University of North Carolina at Chapel Hill
\textsuperscript{2}University of Illinois at Urbana-Champaign

Abstract. Deep convolutional neural networks (CNN) have shown their promise as a universal representation for recognition. However, global CNN activations at present lack geometric invariance, which limits their robustness for tasks such as classification and matching of highly variable scenes. To improve the invariance of CNN activations without degrading their discriminative power, this paper presents a simple but effective scheme called \textit{multi-scale orderless pooling} (or MOP-CNN for short). This approach works by extracting CNN activations for local patches at multiple scales, followed by orderless VLAD pooling of these activations at each scale level and concatenating the result. This feature representation decisively outperforms global CNN activations and achieves state-of-the-art performance for scene classification on such challenging benchmarks as SUN397, MIT Indoor Scenes, and ILSVRC2012, as well as for instance-level retrieval on the Holidays dataset.

1 Introduction

Recently, deep convolutional neural networks (CNN) \cite{1} have achieved great success for image classification \cite{2}. While some researchers are further improving the architectures and learning algorithms for these networks \cite{3,4,5,6,7}, others are considering how to use CNN features as a universal representation for recognition tasks. A number of recent works \cite{8,9,10,11} show that CNN features trained on sufficiently large and diverse datasets such as ImageNet \cite{12} can be successfully transferred to other visual recognition tasks, e.g., scene classification and object localization, with a only limited amount of task-specific training data. Our work also relies on reusing CNN activations as off-the-shelf features for whole-image tasks like scene classification and retrieval. But, instead of simply using the CNN activation vector over the global image, we ask whether we can get improved performance by combining activations extracted at multiple local image windows. Inspired by previous work on spatial and feature space pooling of local descriptors \cite{13,14,15}, we propose a novel and simple pooling scheme that significantly outperforms global CNN activations for image classification and retrieval, even without any fine-tuning on task-specific data.

Image representation has been a driving motivation for research in computer vision for many years. For much of the last decade, bag-of-features (BoF) methods \cite{14,16,17,18,19} were considered to be the state of the art. They represent
an image as an orderless collection of local features, without considering any global spatial information. This orderless nature endows the BoF representation with the robustness to variation in spatial layout. Especially when built on top of locally invariant features like SIFT [20], BoF can be, to some extent, invariant to image scaling, translation, and so on. However, it does not encode any spatial information of the images, motivating the incorporation of loose spatial information in the BoF vectors through spatial pyramid matching (SPM) [13].

Deep CNN, as exemplified by the system of Krizhevsky et al. [2], is a completely different architecture. Raw image pixels are first sent through five convolutional layers, each of which filters the feature maps and then max-pools the output within local neighborhoods. At this point, the representation *strongly preserves global spatial information*. For example, as shown by Zeiler and Fergus [21], the activations from the fifth max-pooling layer can still be reconstructed to form an image that looks similar to the original one. Though max-pooling within each feature map helps to improve invariance to small-scale deformations [22], invariance to larger-scale, more global deformations might be undermined by strongly embedded spatial information. After the filtering and max-pooling layers follow several fully connected layers, finally producing an activation of 4096 dimensions. While it becomes more difficult to reason about the invariance properties of the output of the fully connected layers, we will present an empirical analysis in Section 2 indicating that the final CNN representation is still fairly sensitive to global translation, rotation, and scaling. Even if one does not care about this lack of invariance for its own sake, we show that it directly translates into a loss of accuracy for classification tasks.

Intuitively, bags of features and deep CNN activations lie towards opposite ends of the “orderless” to “globally ordered” spectrum for visual representations. The SPM work [13] realized that BoF has insufficient spatial information for many recognition tasks, and successfully added such information. Inspired by this, we observe that CNN activations preserve too much spatial information, and study the question of whether can we add more orderless information on top of CNN activations to improve their recognition performance. We present a simple but effective framework for doing this, which we refer to as *multi-scale orderless pooling* (MOP-CNN). The idea is summarized in Figure 1, and details will be discussed in detail in Section 3. Briefly, we begin by extracting deep activation features from local patches at multiple scales. Our coarsest scale is the whole image, so global spatial information is still preserved, and our finer scales allow us to capture more local, fine-grained details of the image. Then we aggregate local patch responses at the finer scales via VLAD encoding [15]. The orderless nature of VLAD helps to build a more invariant representation. Finally, we concatenate the original global deep activations with the VLAD features for the finer scales to form our new image representation.

Of the recent efforts adapting off-the-shelf CNN features for various recognition tasks, our work is perhaps the most similar to that of Oquab et al. [10], who extract activations from image sub-windows for tasks where discriminative information is contained in local patches, such as PASCAL image classification.
Our proposed feature is a concatenation of the feature vectors from three levels: (a) Level 1, corresponding to the 4096-dimensional CNN activation for the entire $256 \times 256$ image; (b) Level 2, formed by extracting activations from $128 \times 128$ patches and VLAD pooling them with a codebook of 100 centers; (c) Level 3, formed in the same way as level 2 but with $64 \times 64$ patches.

and action classification. However, they require training data with patch-level supervision and train additional network layers adapted to the target task. By contrast, we require no local supervision and no training of additional network layers. We simply pool all the local activations into a single feature vector describing the image that can be used either for supervised classification or unsupervised retrieval tasks.

The rest of this paper is organized as follows. To motivate our work, Section 2 presents a small-scale empirical study of the behavior of the deep CNN representation in the presence of geometric deformations. This study suggests that CNN activations extracted at sub-image windows can provide more robust and discriminative information than those computed over the entire images. Next, in Section 3, we present our multi-scale orderless pooling approach and discuss relevant baselines. Section 4 presents experiments results for classification on three image datasets (SUN 397, MIT Indoor, and ILSVRC2012) and retrieval on the Holidays dataset. A sizable boost in performance across these popular benchmarks confirms the promise of our method.

Preliminaries. To extract deep features, we use the Caffe CPU implementation [23] pre-trained on ImageNet. Given an input image or a patch, we resample it to $256 \times 256$ pixels, subtract the mean of the pixel values, and feed the patch through the network. Then we take the 4096-dimensional output of the seventh (fully connected) layer, after the rectified linear unit (ReLU) transformation, so that all the values are non-negative (we have also tested the activations before ReLU and found worse performance). To train classifiers on top of these activations or our pooled representations, we use the linear SVM implementation from
Fig. 2. Illustration of image transformations considered in our invariance study. For scaling by a factor of $\rho$, we take crops around the image center of $(1/\rho)$ times original size. For translation, we take crops of 0.7 times the original size and translate them by up to 40 pixels in either direction horizontally or vertically (the translation amount is relative to the normalized image size of $256 \times 256$). For rotation, we take crops from the middle of the image (so as to avoid corner artifacts) and rotate them from -20 to 20 degrees about the center. The corresponding scaling ratio, translation distance (pixels) and rotation degrees are listed below each instance.

the INRIA JSGD package [24]. For all experiments, we fix the regularization parameter to $10^{-5}$ and the learning rate to 0.2, and train for 100 epochs.

2 Invariance Analysis of Deep Activations

To motivate our research, we first investigate the limitations of the global CNN representation. As part of their paper on visualizing deep features, Zeiler and Fergus [21] analyze the transformation invariance of their model on five individual images by displaying the distance between the feature vectors of the original and transformed images, as well as the change in the probability of the correct label for the transformed version of the image (Figure 5 of [21]). These plots show very different patterns for different images, making it difficult to draw general conclusions. We would like to conduct a more comprehensive analysis with an emphasis on prediction accuracy for entire categories, not just individual images. We randomly select four example classes from the SUN dataset [25]: arrival gate, florist shop, volleyball court, and ice skating. Each class contains around 150 images; we train one-vs-all linear SVMs on 50 random images per class on top of global 4096-dimensional CNN activations. All the remaining images are used for testing; we apply a given transformation to all of them, extract the same features, and compute SVM prediction scores for the four classes. The transformations we consider include translation, scaling, flipping and rotation (see Figure 2 for illustration and detailed explanation of transformation parameters). Figure 3 shows the average sigmoid-transformed SVM prediction scores for each class as well as the classification accuracies on the four-class problem for each type of transformation. We can see that for all transformations except horizontal flipping, as the degree of transformation becomes more extreme, both
Fig. 3. Average sigmoid-transformed SVM prediction scores and classification accuracies for different transformations on four SUN classes. For each group of two figures, the upper figure shows the prediction scores, and the bottom figure shows the classification accuracy.

the average prediction scores and the classification accuracies keep dropping for all classes. As for invariance to horizontal flipping, a possible explanation is that the Caffe implementation adds horizontal flips of all training images to the training set (on the other hand, the Caffe training protocol also involves taking random crops of training images, yet this does not seem sufficient for building invariance to such transformations, as our results indicate).

From the analysis of Figure 3, it is clear that testing on shifted or scaled crops of an image can significantly affect prediction scores. Figure 4 shows the predictions for the 1000-category ILSVRC benchmark for different image windows. From (a)-(f), we can see that even for sub-windows that are small translations of each other, the label predictions can be drastically different. For example, in (f), the red rectangle is correctly labeled “alp,” while the overlapping rectangle
Fig. 4. Classification of local patches inside of the image. The ground truth labels are listed below each image. Labels predicted by the whole image representations are listed in the bottom right corner.

Fig. 5. Highest-response windows (in red) for (a) basilica, (b) control tower, (c) boardwalk, and (d) tower. For each test image resampled to 256 × 256, we search over windows with widths 224, 192, 160, and 128 and a stride of 16 pixels and display the window that gives the highest prediction score for the ground truth category. The detected windows contain similar structures: in (a), (b) and (d), the top parts of towers have been selected; in (c), the windows are all centered on the narrow walkway.

is incorrectly labeled “garfish.” But, while picking the wrong window can give a bad prediction, picking the “right” one can give a good prediction: in (d), the whole image is wrongly labeled, but one of its sub-windows can get the correct label – “schooner.” This immediately suggests a sliding window protocol at test
Multi-scale Orderless Pooling of Deep Convolutional Activation Features

Our representation has three scale levels. The first level corresponds to the 4096-dimensional CNN activation for the global $256 \times 256$ image. For the second and third levels, we extract activations for $128 \times 128$ and $64 \times 64$ patches, respectively, with a stride of 32 pixels. Since the second and third levels contain multiple patches, we need to pool their activations somehow to obtain a single feature vector of reasonable dimensionality. For this, we adopt the vector of locally aggregated descriptors (VLAD) framework [27], which is a simplified version of Fisher Vectors (FV) [14]. Let $p_j$ denote the activation for the $j$th patch at the second or third level. To make computation more efficient, before pooling, we use PCA to reduce these activations to 500 dimensions. We also learn a separate $k$-means codebook for each level with $k = 100$ centers. We denote each center as $c_i$, $i = 1, \ldots, k$. The VLAD descriptor is constructed by assigning each patch to its nearest cluster center and aggregating the residuals of the patches minus the center:

$$x = \left[ \sum_{\text{NN}(p_j)=c_1} p_j - c_1, \sum_{\text{NN}(p_j)=c_2} p_j - c_2, \ldots, \sum_{\text{NN}(p_j)=c_k} p_j - c_k \right],$$

where $\text{NN}(p_j) = c_i$ denotes all patches $p_j$ whose nearest codeword is $c_i$. Following [27], we additionally perform power- and L2-normalization of the pooled vector. The resulting VLAD vector still has quite high dimensionality: given
500-dimensional activations (after PCA) and 100 $k$-means centers, we end up with 50,000 dimensions. This is too high for many large-scale applications, so we further perform PCA on the global vectors and reduce them to 4096 dimensions. Note that applying PCA during the two stages (local patch activation and global feature vector) of this pooling pipeline is a standard practice in previous works [28,29]. Finally, given the original 4096-dimensional feature vector from level one and the two 4096-dimensional pooled PCA-reduced vectors from levels two and three, we rescale them to unit norm and concatenate them to form our final image representation.

Baseline methods. To validate our proposed representation, we need to demonstrate that a simpler patch sampling and pooling scheme cannot achieve the same performance. As simpler alternatives to VLAD pooling, we consider average pooling, which simply involves computing the mean of the 4096-dimensional activations at each scale level, and maximum pooling, which involves computing their element-wise maximum. We did not consider standard BoF pooling because it has been demonstrated to be less accurate than VLAD [27]; to get competitive performance, we would need a codebook size much larger than 100, which would make the quantization step prohibitively expensive. Moreover, we need to examine alternative strategies with regards to pooling across scale levels. The multiscale strategy corresponds to taking the union of all the patches from an image, regardless of scale, and pooling them together. The concatenation strategy refers to pooling patches from three levels separately and then concatenating the result. In addition, we separately examine the performance of individual scale levels as well as concatenations of just pairs of them. In particular, level1 is simply the 4096-dimensional global descriptor of the entire image, which was suggested in [8] as a generic image descriptor. These baselines and their relationship to our full MOP-CNN scheme are summarized in Table 1.

### Table 1. A summary of baselines and their relationship to the MOP-CNN method.

| pooling method / scale | multiscale | concatenation |
|------------------------|------------|---------------|
| Average pooling        | Avg (multiscale) | Avg (concatenation) |
| Max pooling            | Max (multiscale) | Max (concatenation) |
| VLAD pooling           | VLAD (multiscale) | MOP-CNN |

4 Experimental Evaluation

4.1 Datasets

We test our approach on four well-known benchmark datasets:

SUN397 [25] is the largest dataset to date for scene recognition. It contains 397 scene categories and each has at least 100 images. The evaluation protocol involves training and testing on ten different splits and reporting the average classification accuracy. The splits are fixed and publicly available from [25]; each has 50 training and 50 test images.
MIT Indoor [30] contains 67 categories. While outdoor scenes, which comprise more than half of SUN (220 out of 397), can often be characterized by global scene statistics, indoor scenes tend to be much more variable in terms of composition and better characterized by the objects they contain. This makes the MIT Indoor dataset an interesting test case for our representation, which is designed to focus more on appearance of sub-image windows and have more invariance to global transformations. The standard training/test split for the Indoor dataset consists of 80 training and 20 test images per class.

ILSVRC2012 [12] is the most prominent benchmark for comparing large-scale image classification methods and is the dataset on which the Caffe representation we use [23] is pre-trained. It contains 1000 classes corresponding to leaf nodes in ImageNet. Each class has more than 1000 unique training images, and there is a separate validation set with 50,000 images. Classification accuracy on the validation set is used to evaluate different methods. One difference between this dataset and the previous two is that most of its categories focus on objects, not scenes, and the objects tend to be highly salient and centered in images.

Holidays [31] is a standard benchmark for image retrieval. It contains 1491 images corresponding to 500 image instances. Each instance has 2-3 images describing the same object or location. A set of 500 images are used as queries, and the rest are used as the database. Mean average precision (mAP) is the evaluation metric.

### 4.2 Image Classification Results

Table 2 reports our results on the SUN397 dataset. From the results for baseline pooling methods in (a), we can see that VLAD works better than average and max pooling and that pooling scale levels separately works better than pooling them together (which is not altogether surprising, since the latter strategy raises the feature dimensionality by a factor of three). From (b), we can see that concatenating all three scale levels gives a significant improvement over any subset. For reference, Part (c) of Table 2 gives published state-of-the-art results from the literature. Xiao et al. [25], who have collected the SUN dataset, have also published a baseline accuracy of 38% using a combination of standard features like GIST, color histograms, and BoF. This baseline is slightly exceeded by the level1 method, i.e., global 4096-dimensional Caffe activations pre-trained on ImageNet. The Caffe accuracy of 39.57% is also comparable to the 40.94% with an analogous setup for DeCAF [8]. However, these numbers are still worse than the 47.2% achieved by high-dimensional Fisher Vectors [32] – to our knowledge, the state of the art on this dataset to date. With our MOP-CNN pooling scheme, we are able to achieve 51.98% accuracy with feature dimensionality that is an order of magnitude lower than that of [32]. Figure 6 shows six classes on which MOP-CNN gives the biggest improvement over level1, and six on which it has

---

1 DeCAF is an earlier implementation from the same research group and Caffe is its “little brother.” The two implementations are similar, but Caffe is faster, includes support for both CPU and GPU, and is easier to modify.
Table 2. Scene recognition on SUN397. (a) Alternative pooling baselines (see Section 3 and Table 1); (b) Different combinations of scale levels – in particular, “level1” corresponds to the global CNN representation and “level1 + level2 + level3” corresponds to the proposed MOP-CNN method. (c) Published numbers for state-of-the-art methods.

| method                        | feature dimension | accuracy    |
|-------------------------------|-------------------|-------------|
| (a)                           |                   |             |
| Avg (Multiscale)              | 4,096             | 39.62       |
| Avg (Concatenation)           | 12,288            | 47.50       |
| Max (Multiscale)              | 4,096             | 43.51       |
| Max (Concatenation)           | 12,288            | 48.50       |
| VLAD (Multiscale)             | 4,096             | 47.32       |
| (b)                           |                   |             |
| level1                        | 4,096             | 39.57       |
| level2                        | 4,096             | 45.34       |
| level3                        | 4,096             | 40.21       |
| level1 + level2               | 8,192             | 49.91       |
| level1 + level3               | 8,192             | 49.52       |
| level2 + level3               | 8,192             | 49.66       |
| level1 + level2 + level3 (MOP-CNN) | 12,288         | **51.98**   |
| (c)                           |                   |             |
| Xiao et al. [25]              | –                 | 38.00       |
| DeCAF [8]                     | 4,096             | 40.94       |
| FV (SIFT + Local Color Statistic) [32] | 256,000        | 47.20       |

Fig. 6. SUN classes on which MOP-CNN gives the biggest decrease over the level1 global features (top), and classes on which it gives the biggest increase (bottom).
Table 3. Classification results on MIT Indoor Scenes. (a) Alternative pooling baselines (see Section 3 and Table 1); (b) Different combinations of scale levels; (c) Published numbers for state-of-the-art methods.

Table 3 reports results on the MIT Indoor dataset. Overall, the trends are consistent with those on SUN, in that VLAD pooling outperforms average and max pooling and combining all three levels yields the best performance. There is one interesting difference from Table 2, though: namely, level2 and level3 features work much better than level1 on the Indoor dataset, whereas the difference was much less pronounced on SUN. This is probably because indoor scenes are better described by local patches that have highly distinctive appearance but can vary greatly in terms of location. In fact, several recent methods achieving state-of-the-art results on this dataset are based on the idea of finding such patches [35,34,33]. Our MOP-CNN scheme outperforms all of them – 68.88% vs. 64.03% for the method of Doersch et al. [35].

Table 4 reports results on the ILSVRC2012 dataset. The trends for alternative pooling methods in (a) are the same as before. Interestingly, in (b) we can see that, unlike on SUN and MIT Indoor, level2 and level3 features do not work as well as level1. This is likely because the level1 feature was specifically trained on ILSVRC2012, and this dataset has limited geometric variability. Nevertheless, by combining the three levels, we still get a significant improvement. As for state-of-the-art results on this dataset (part (c) of Table 4), they are given by recent CNN implementations specifically trained on that dataset. In particular, directly running the full pre-trained Caffe network on the global features from the validation set gives 54.34% accuracy, which is higher than our level1 accuracy of 51.46%. The only difference between these two setups, “Caffe (Global)” and “level1,” are the parameters of the last classifier layer – i.e., softmax and SVM, respectively. For Caffe, the softmax layer is jointly trained with all the
previous network layers using multiple random windows cropped from training images, while our SVMs are trained separately using only the global image features. Nevertheless, the accuracy of our final MOP-CNN representation, at 57.93%, is higher than that of the full pre-trained Caffe CNN tested either on the global features (“Global”) or on ten sub-windows (“Center+Corner+Flip”). The highest reported accuracy on ILSVRC2012 is the 59.93% of Krizhevsky et al. [2], but there is no publicly available implementation that fully corresponds to their system (DeCAF [8] and Caffe [23] come a few percent short). However, our MOP-CNN feature is built on top of a CNN with 51% or 54% classification accuracy (depending on how the final classification layer is trained) and is able to improve on its performance, and we expect that it can similarly improve over a more accurate baseline CNN. Figure 7 plots the classification rates for all individual classes for level1 vs. MOP-CNN.

4.3 Image Retrieval Results

As our last experiment, we demonstrate the usefulness of our approach for an unsupervised image retrieval scenario on the Holidays dataset. Table 5 reports the mAP results for nearest neighbor retrieval of feature vectors using the Euclidean distance. On this dataset, level1 is the weakest of all three levels because images of the same instance may be related by large rotations, viewpoint changes, etc., and global CNN activations do not have strong enough invariance to handle these transformations. As before, combining all three levels achieves the best accuracy.
Fig. 7. Per-class classification accuracies on ILSVRC2012 classes for level1 (blue curve) vs. MOP-CNN (red dots). A couple of classes showing especially big increases or decreases have been manually highlighted.

| method                                      | feature dimension | mAP  |
|---------------------------------------------|-------------------|------|
|(a) Avg (Multiscale)                         | 4,096             | 71.32|
|Avg (Concatenation)                         | 12,288            | 75.02|
|Max (Multiscale)                            | 4,096             | 76.23|
|Max (Concatenation)                         | 12,288            | 75.07|
|VLAD (Multiscale)                           | 4,096             | 78.42|
|(b) level1                                  | 4,096             | 70.53|
|level2                                      | 4,096             | 74.02|
|level3                                      | 4,096             | 75.45|
|level1 + level2                             | 8,192             | 75.86|
|level1 + level3                             | 8,192             | 78.92|
|level2 + level3                             | 8,192             | 77.91|
|level1 + level2 + level3 (MOP-CNN)          | 12,288            | 78.82|
|(c) MOP-CNN + PCA + Whitening              | 512               | 78.38|
|MOP-CNN + PCA + Whitening                  | 2048              | **80.18**|
|(d) FV [27]                                 | 8,192             | 62.50|
|FV + PCA [27]                               | 256               | 62.60|

Table 5. Image retrieval results on the Holidays dataset. (a) Alternative pooling baselines (see Section 3 and Table 1); (b) Different combinations of scale levels; (c) Full MOP-CNN descriptor vector compressed by PCA and followed by whitening [36], for two different output dimensionalities; (c) Published state-of-the-art results with a compact global descriptor (see text for discussion).

performance of 78.82%. Using aggressive dimensionality reduction with PCA and whitening as suggested in [36], we can raise the mAP even further to 80.8% with only a 2048-dimensional feature vector. The state of the art performance on this dataset with a compact descriptor is obtained by Jégou et al. [27] by using FV/VLAD and PCA. Our approach still achieves a clear advantage. Note that it is possible to obtain even better results on Holidays with methods based on inverted files with very large vocabularies. In particular, Tolias et al. [37] report 88% but their representation would take more than 4 million dimensions per image if expanded into an explicit feature vector, and is not scalable to large datasets. Even further improvements may be possible by adding techniques such as query expansion and geometric verification, but they are not applicable for
Fig. 8. Image retrieval examples on the Holiday dataset. Red border means ground truth images. We only show three retrieved examples per query because each query only has one to two ground truth images.

generic image representation, which is our main focus. Finally, we show retrieval examples in Figure 8. We can clearly see that MOP-CNN has improved robustness to shifts, scaling, and viewpoint changes over global CNN activations.

5 Discussion

This paper has presented a multi-scale orderless pooling scheme that is built on top of deep activation features of local image patches. On four very challenging datasets, we have achieved a substantial improvement over global CNN activations, in most cases outperforming the state of the art. These results are achieved with the same set of parameters (i.e., patch sizes and sampling, codebook size, PCA dimension, etc.), which clearly shows the good generalization ability of the proposed approach. As a generic low dimensional image representation, it is not restricted to supervised tasks like image classification, but can also be used for unsupervised tasks such as retrieval.

We believe this work opens several promising avenues for future research. It remains interesting to investigate more sophisticated ways to incorporate orderless information in CNN. One possible way is to change the architecture of current deep networks fundamentally to improve their holistic invariance. Also, the feature extraction stage of our current pipeline is somewhat slow, and it is interesting to exploit the convolutional network structure to speed it up.
References

1. LeCun, Y., Boser, B., Denker, J., Henderson, D., Howard, R., Hubbard, W., Jackel, L.: Handwritten digit recognition with a back-propagation network. NIPS (1990)
2. Krizhevsky, A., Sutskever, I., Hinton, G.: Imagenet classification with deep convolutional neural networks. In: Advances in Neural Information Processing Systems 25. (2012) 1106–1114
3. Goodfellow, I., Warde-Farley, D., Mirza, M., Courville, A., Bengio, Y.: Maxout networks. ICML (2013)
4. Le, Q., Ranzato, M., Monga, R., Devin, M., Chen, K., Corrado, G., Dean, J., Ng, A.: Building high-level features using large scale unsupervised learning. In: International Conference in Machine Learning. (2012)
5. Wan, L., Zeiler, M., Zhang, S., Lecun, Y., Fergus, R.: Regularization of neural networks using dropconnect. ICML (2013)
6. Hinton, G.E., Srivastava, N., Krizhevsky, A., Sutskever, I., Salakhutdinov, R.R.: Improving neural networks by preventing co-adaptation of feature detectors. Arxiv (2012)
7. K. Simonyan, A.V., Zisserman, A.: Deep fisher networks for large-scale image classification. In: Proceedings Advances in Neural Information Processing Systems (NIPS). (2013)
8. Donahue, J., Jia, Y., Vinyals, O., Hoffman, J., Zhang, N., Tzeng, E., Darrell, T.: Decaf: A deep convolutional activation feature for generic visual recognition. arXiv preprint arXiv:1310.1531 (2013)
9. Girshick, R., Donahue, J., Darrell, T., Malik, J.: Rich feature hierarchies for accurate object detection and semantic segmentation. arXiv preprint arXiv:1311.2524 (2013)
10. Oquab, M., Bottou, L., Laptev, I., Sivic, J., et al.: Learning and transferring mid-level image representations using convolutional neural networks. In: Arxiv. (2013)
11. Sermanet, P., Eigen, D., Zhang, X., Mathieu, M., Fergus, R., LeCun, Y.: Overfeat: Integrated recognition, localization and detection using convolutional networks. arXiv preprint arXiv:1312.6229 (2013)
12. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: ImageNet: A large-scale hierarchical image database. CVPR (2009)
13. Lazebnik, S., Schmid, C., Ponce, J.: Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. CVPR (2006)
14. Perronnin, F., Dance, C.R.: Fisher kernels on visual vocabularies for image categorization. CVPR (2007)
15. Jégou, H., Douze, M., Schmid, C., Pérez, P.: Aggregating local descriptors into a compact image representation. In: Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on, IEEE (2010) 3304–3311
16. Wang, J., Yang, J., Yu, K., Lv, F., Huang, T., Gong, Y.: Locality-constrained linear coding for image classification. CVPR (2010)
17. Csurka, G., Dance, C., Fan, L., Willamowski, J., Bray, C.: Visual categorization with bags of keypoints. In: ECCV Workshop on Statistical Learning in Computer Vision. (2004)
18. Sivic, J., Zisserman, A.: Video Google: A text retrieval approach to object matching in videos. In: ICCV. (2003)
19. Grauman, K., Darrell, T.: The pyramid match kernel: Discriminative classification with sets of image features. In: In ICCV. (2005) 1458–1465
20. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. International journal of computer vision 60(2) (2004) 91–110
21. Zeiler, M.D., Fergus, R.: Visualizing and understanding convolutional neural networks. arXiv preprint arXiv:1311.2901 (2013)
22. Lee, H., Grosse, R., Ranganath, R., Ng, A.Y.: Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations. In: Proceedings of the 26th Annual International Conference on Machine Learning, ACM (2009) 609–616
23. Jia, Y.: Caffe: An open source convolutional architecture for fast feature embedding. http://caffe.berkeleyvision.org/ (2013)
24. Akata, Z., Perronnin, F., Harchaoui, Z., Schmid, C., et al.: Good practice in large-scale learning for image classification. IEEE Transactions on Pattern Analysis and Machine Intelligence (2013)
25. Xiao, J., Hays, J., Ehinger, K.A., Oliva, A., Torralba, A.: Sun database: Large-scale scene recognition from abbey to zoo. In: Computer vision and pattern recognition (CVPR), 2010 IEEE conference on, IEEE (2010) 3485–3492
26. Pandey, M., Lazebnik, S.: Scene recognition and weakly supervised object localization with deformable part-based models. In: IEEE International Conference on Computer Vision (ICCV), IEEE (2011) 1307–1314
27. Jégou, H., Douze, M., Schmid, C., Perez, P.: Aggregating local descriptors into a compact image representation. CVPR (2010)
28. Perronnin, F., Sanchez, J., Mensink, T.: Improving the fisher kernel for large-scale image classification. ECCV (2010)
29. Perronnin, F., Liu, Y., Sánchez, J., Poirier, H.: Large-scale image retrieval with compressed Fisher vectors. In: CVPR. (2010)
30. Quattoni, A., Torralba, A.: Recognizing indoor scenes. CVPR (2009)
31. Jégou, H., Douze, M., Schmid, C.: Hamming embedding and weak geometric consistency for large-scale image search. ECCV (2008)
32. Sanchez, J., Perronnin, F., Mensink, T., Verbeek, J.: Image Classification with the Fisher Vector: Theory and Practice. IJCV 105(3) (2013) 222–245
33. Singh, S., Gupta, A., Efros, A.A.: Unsupervised discovery of mid-level discriminative patches. In: European Conference on Computer Vision. (2012)
34. Juneja, M., Vedaldi, A., Jawahar, C.V., Zisserman, A.: Blocks that shout: Distinctive parts for scene classification. In: IEEE Conference on Computer Vision and Pattern Recognition. (2013)
35. Doersch, C., Gupta, A., Efros, A.A.: Mid-level visual element discovery as discriminative mode seeking. In: Neural Information Processing Systems (NIPS). (2013)
36. Jégou, H., Chum, O.: Negative evidences and co-occurrences in image retrieval: the benefit of PCA and whitening. In: ECCV - European Conference on Computer Vision, Firenze, Italy (October 2012)
37. Tolias, G., Avrithis, Y., Jégou, H.: To aggregate or not to aggregate: selective match kernels for image search. In: ICCV - International Conference on Computer Vision. (2013)