Learning a Representative and Discriminative Part Model with Deep Convolutional Features for Scene Recognition

Bingyuan Liu, Jing Liu, Jingqiao Wang, Hanqing Lu

Institute of Automation, Chinese Academy of Sciences, Beijing, China
{byliu, jliu, jqwang, luhq}@nlpr.ia.ac.cn

Abstract. The discovery of key and distinctive parts is critical for scene parsing and understanding. However, it is a challenging problem due to the weakly supervised condition, *i.e.*, no annotation for parts is available. To address above issues, we propose a unified framework for learning a representative and discriminative part model with deep convolutional features. Firstly, we employ selective search method to generate regions that are more likely to be centered around the distinctive parts, which is used as parts training set. Then, the feature of each part region is extracted by forward propagating it into the Convolutional Neural Network (CNN). The CNN network is pre-trained by the large auxiliary ImageNet dataset and then fine-tuned on the particular scene images. To learn the parts model, we build a mid-level part dictionary based on sparse coding with a discriminative regularization. The two terms, *i.e.*, the sparse reconstruction error term and the label consistent term, indicate the representative and discriminative properties respectively. Finally, we apply the learned parts model to build image-level representation for the scene recognition task. Extensive experiments demonstrate that we achieve state-of-the-art performances on the standard scene benchmarks, *i.e.* Scene-15 and MIT Indoor-67.

1 Introduction

The task of scene recognition remains one of the most important but challenging problems in computer vision and machine intelligence. To solve this problem, how to build a suitable image representation is very critical. Conventional methods take advantage of the well engineered local features, such as SIFT[1] and HOG[2], to build Bag-of-Features (BoF)[3] image representation. However, this representation mostly captures local edges without enough mid-level and high-level information, which hinders the performance.

Recently, deep Convolutional Neural Network (CNN) has achieved great success in image classification by showing substantially higher accuracy on the ImageNet Large Scale Visual Recognition Challenge[4][5]. It is considered that the CNN may be used as a universal feature extractor for various vision tasks[6]. A number of recent works have also shown that CNN trained on sufficiently large and diverse datasets such as ImageNet can be successfully transferred to other

visual recognition tasks, *e.g.*, human attribute classification[7] and object detection[8], with domain-specific fine-tuning by the limited amount of task-specific training data. In this paper, we generalize the CNN to the scene recognition domain and explore it for the distinctive part discovery and recognition.

To recognition scene categories, discovering distinctive parts to build image representation is very effective, such as screens in movie theater and tables in dining room. While the notion of part is widely used in object recognition, *e.g.* the Deformable Part Models (DPM)[9], it is still very difficult on the condition of scene classification, as there is only image-level label without further information on parts. For learning a good part model, two key requirements should be satisfied. One is the representative property, *i.e.*, the parts model should frequently occur within the dataset and typically indicate a particular category. The other one is the discriminative property. That is, the discovered mid-level part primitives are sufficiently different among diverse categories and help improve the final recognition task. In [10], they applied K-means to initialize the parts model and then train a linear SVM classifier for each cluster to select the most discriminative clusters. Juneja [10] proposed to initialize a set of parts by the selective search method[11] and then train part detectors to identify distinctive parts. Most previous methods adopted heuristic or iterative scheme, which firstly initial a representative model and then enhance its discrimination. However, we consider it is optimal to jointly encourage the two requirements and introduce a unified learning framework.

In this paper, we adapt the CNN features for parts discovery as analysis above, and introduce a unified learning framework jointly encouraging representative and discriminative properties. We firstly generate a particular parts training set that are more likely to be centered around distinctive parts by selective search[11], which is a method based on low-level image cues and over-segmentations. Then we employ affine warping to compute a fixed-size CNN input of each part proposal and obtain a fixed-length feature vector by the CNN forward operation, where the CNN network is pre-trained by the large auxiliary ImageNet dataset and fine-tuned on the particular scene image samples. Furtherly, we learn a mid-level part dictionary based on sparse coding, containing a sparse reconstruction error term and a label consistent regularization. The sparse reconstruction guarantees that the learned parts are significantly informative in the dataset, while the label consistent regularization encourages that different input from different categories have discriminative responses. Finally, we apply the learned parts model to build image-level representation for the scene recognition task. Extensive experiments on the benchmarks of Scene-15 and MIT Indoor-67 demonstrate the effectiveness of our method compared with related works. Combining with CNN features of the global image, we achieves state-of-the-art performances on both datasets.

2 Related Work

The introduction of some well engineered image descriptors (*e.g.* SIFT[1] and HOG[2]) have precipitated dramatic success and dominante most visual tasks in past decades. However, these kinds of features are unable to represent more complex mid and high level image structures. Over the recent years, a growing amount of researches focus on feature learning and selection[12][13], especially on building deep learning models for hierarchical image representations[14]. Deep Convolutional Neural Networks (CNN) is one of the most successful deep representation learning models, as it achieved great success in image classification by showing substantially higher accuracy on the ImageNet Challenge[4][5]. Some works still focus on further improving the CNN architectures and learning algorithm. Hinton *et al.*[15] proposed dropout by randomly omitting half of the feature detectors on each training case to prevent over-fitting, while Wan *et al.*[16] generalized this idea by setting a randomly selected subset of weights within the network to zero for regularizing large fully-connected layers. Some other works started to consider CNN as a universal image feature extractor for visual tasks[6]. Sun *et al.*[17] proposed to apply cascaded CNN for facial point detection, while Toshev *et al.*[18] adapted CNN for human pose estimation. In [8], CNN was explored as a region feature extractor and applied to solve object detection task. Most of these works use the highly effective "supervised pre-training/domain-specific fine-tuning" paradigm, which transfers CNN trained on sufficiently large and diverse datasets to other visual tasks. In this paper, we adapt the CNN features for the task of parts model learning in the scene recognition task.

Scene recognition and understanding is a fundamental task in computer vision. The key to solve this problem is how to obtain a suitable image representation. Many previous works are based on Bag-of-Features (BoF) model, which takes advantage of the traditional local features and the power of SVM classifier. Some efforts were made to improve the description power, such as quantizing local features with less information loss[19][20], building more effective codebook[21] and adopting kernel methods[22]. Other works attempted to incorporate some spatial information, such as the famous spatial pyramid match (SPM)[23]. Sharma *et al.* [24] defined a space of grids where each grid is obtained by a series of recursive axis aligned splits of cells and proposed to learn the spatial partition in a maximum margin formulation. An Orientational Pyramid Matching (OPM) model [25] was proposed to improve SPM, which uses the 3D orientations to form the pyramid and produce the pooling regions.

Since the distinctive parts are very important to recognize a typical scene, many researchers attempted to discovery and learn parts model for scene recognition. Zheng *et al.*[26] transfered the deformable part-based models[9] to build image representation. Singh *et al.*[10] used iterative procedure which alternates between clustering and training discriminative classifiers to discover discriminative patches, while Junejia *et al.*[27] apply exemplar SVM to train part detectors and iteratively identify distinctive parts from an initial set of parts. Lin *et al.*[28] proposed to jointly learn the part appearance and important spatial pooling re-

gions. Different from the iterative scheme, we use a unified framework, jointly incorporating representative and discriminative properties.

3 The proposed Model

In this section, we describe the details of our proposed model for learning distinctive parts for scene recognition. We will firstly present how to generate the particular training parts set prepared for parts learning and how to extract part features by CNN. Then our unified parts model learning algorithm is followed. At last, we will introduce how to employ the learned parts model to construct image-level representation.

3.1 Parts Training Set Generation

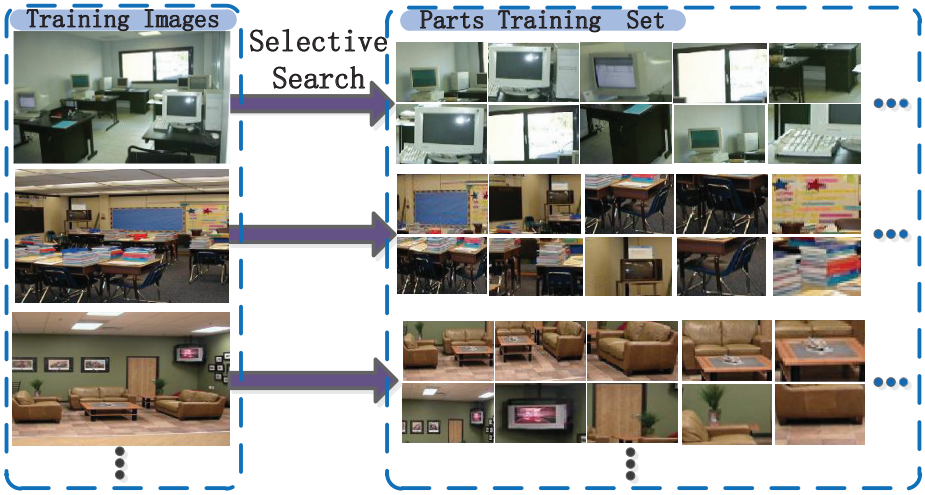

Fig. 1. The generation process of parts training set.

In our framework, an initial parts training set is needed to generate, prepared for part model learning. In the weakly supervised scene dataset (only image-level label without any label on parts), any sub-window in the training images is likely to contain a distinctive part. This simple way is to exhaustively include all the possible regions for parts learning. However, most of these regions don't contain valuable information, leading to high information redundancy and extra computation cost. We may also randomly sample a subset from all the possible regions to decrease the number, but this can not guarantee to cover all the useful regions.

To reduce the number of training samples as possible and contain most distinctive parts meanwhile, we turn to the selective search method[11]. Based on low-level image cues, the selective search method combines the strength of both an exhaustive search and segmentation to capture all possible object locations. Extending from objects to parts, we find the generated sub-windows of selective search on scene images tend to be centered around distinctive parts we want, as shown in Fig. 1. In particular, each training image is firstly resized into multiple scales and then segmented into superpixels. A greedy algorithm is then employed which iteratively groups the two most similar regions together and calculates the similarities between this new region and its neighbors. The number of obtained region proposals ranges from 100 to 800 for each image. As evaluated in [11], this method is performed with very high recall, guaranteeing that almost every distinctive part is included. The overall generated part proposals for training are denoted as P in this paper.

3.2 Part Feature Extraction

Given a part proposal, we extract its feature by forward propagating it through the Convolutional Neural Network (CNN). We employ the Caffe[29] GPU implementation of the CNN architecture described in [4], which consists of five convolutional layers, two fully connected layers and uses the rectified linear unit (ReLu) as the activation function (please refer to [4] for more network details). Since the CNN requires inputs of a fixed $227 \times 227 \times 3$ pixel size, we resize each region to 227×227 RGB pixels, subtract the mean of the pixel values and then feed it into the network. The 4096-dimensional output of the seventh layer (the final fully connected layer) after the ReLU transformation is taken as the representation of the input part region (the performances of the previous layers are found worse in our experiments) .

To train the CNN network, we apply the very effective "supervised pre-training/domain-specific fine-tuning" scheme. Firstly, we discriminatively pre-train the CNN on a large auxiliary dataset, *i.e.* ImageNet 2012 benchmark, which includes roughly 1.2 million training samples from 1000 categories and $50,000$ images for evaluation. To generalize the pre-trained CNN to the new domain (*i.e.* scene recognition), we fine-tune the network on the task dataset (*e.g.* Scene-15 and MIT Indoor-67). The fine-tuning is performed by continuing the stochastic gradient descent learning process with new training samples, where all the CNN parameters is initialized by the pre-trained CNN except that the ImageNet-specific 1000-way softmax layer is replaced by a randomly initialized new softmax layer with the number of output units altered to the number of classes in the new dataset. As the domain dataset is small we further augment the training set by adding cropped, rotated and mirror samples. It is noted that the fine-tuning is started at a learning rate of 0.001 (1/10 of the initial pre-training rate), allowing the fine-tuning to make progress while not clobbering the initialization. Both the pre-training and fine-tuning in this paper is carried out using the Caffe GPU implementation.

3.3 Part Model Learning

Different from previous works, we hope to jointly encourage the representative and discriminative properties of the parts model. In another word, these two requirements mean that the learned parts frequently occur in each category and are able to distinguish different classes meanwhile. We denote the part training set as $P = \{p_1, p_2, \ldots, p_N\}$, where the number of training samples is denoted as N. The CNN feature of each part p_i is represented by a m-dimensional vector x_i. We directly impose the image label on the part proposals extracted within the image and denote the corresponding parts training label set as $Y = \{y_1, y_2, \ldots, y_N\}$.

With the parts training dataset prepared, we learn a mid-level parts dictionary based on sparse coding, while the discriminativity is motivated by a label consistent regularization. The unified part model learning objective with the two properties incorporated is defined as:

$$\begin{gathered} \min_{B,A,Z} \parallel X - BZ \parallel_2^2 + \alpha \parallel D - AZ \parallel_2^2 \\ s.t. \forall i, |z_i|_1 \preceq T \end{gathered} \tag{1}$$

This objective is comprised of two sections, *i.e.*, the reconstruction error term and label consistent regularization, where α controls the relative contribution. In Eq. 1, $B = [b_1, b_2, \ldots, b_k] \in R^{m \times K}$ represents the part dictionary and the size is denoted as K, while the latent code $Z = [z_1, z_2, \ldots, z_N] \in R^{K \times N}$ denotes the response vectors of the training part proposals. As the l_1 regularization constraint of z_i encourages it to be sparse (T denotes the sparse factor), the first term is a sparse coding term which is able to help learn a representative B. $D = [d_1, d_2, \ldots, d_N] \in R^{K \times N}$ in the second term are the discriminative sparse codes of training parts. We define that $d_i = [d_i^1, d_i^2, \ldots, d_i^K]^t = [0 \ldots 1, 1, \ldots 0]^t \in R^K$ is the discriminative code related to an training part x_i, if the non-zero values of d_i occur at those indices where the part and the model item b_k share the same label. For example, assuming there are 3 classes, 8 training samples and the dictionary size is 6, we denote $B = [b_1, b_2, \ldots, b_6]$ and $X = [x_1, x_2, \ldots, x_8]$. The x_1, x_2, b_1 and b_2 are from class 1, x_3, x_4, x_5, b_3 and b_4 are from class 2, and x_6, x_7, x_8, b_5 and b_6 are from class 3. Thus the D is set as:

$$D = \begin{pmatrix} 1\,1\,0\,0\,0\,0\,0\,0 \\ 1\,1\,0\,0\,0\,0\,0\,0 \\ 0\,0\,1\,1\,1\,0\,0\,0 \\ 0\,0\,1\,1\,1\,0\,0\,0 \\ 0\,0\,0\,0\,0\,1\,1\,1 \\ 0\,0\,0\,0\,0\,1\,1\,1 \end{pmatrix}_{6 \times 8} \tag{2}$$

In the second term, A is a linear transformation matrix, which transforms the latent sparse codes z to be most discriminative. As this term may measure the discriminative error of the latent sparse response and enforces that the sparse codes approximate the discriminative codes Q, it encourages the input from

different classes to have different responses, thus enhancing the discriminative property.

For the optimization, we utilize the algorithm similar to [30]. We firstly rewrite Eq. 1 as:

$$\begin{aligned} \min_{B,A,Z} \parallel [X^T \ \sqrt{\alpha}D]^T - [B^t \ \sqrt{\alpha}A]^t Z \parallel_2^2 \\ s.t. \forall i, |z_i|_1 \preceq T \end{aligned} \tag{3}$$

Then the problem can be solved by the standard K-SVD[31] algorithm to find the optimal solution for all the parameters.

3.4 Image-level Representation

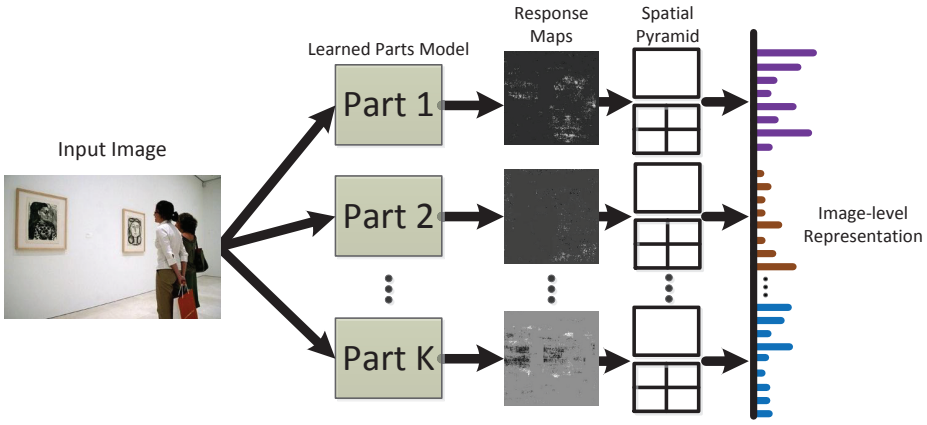

Fig. 2. The pipeline of building image-level representation.

With the learned part model, we regard it as a mid-level visual dictionary and build an image-level bag-of-part representation. As shown in Fig. 2, given a new image, we apply each part in the learned part dictionary as a template to slide over the input image in a densely sampled scheme and obtain the corresponding response map. The response value in each map is calculated by solving a sparse coding optimization:

$$\begin{aligned} \min_{z} \parallel x - Bz \parallel_2^2 \\ s.t. |z|_1 \preceq T \end{aligned} \tag{4}$$

where x is a sampled input patch, B is the learned parts dictionary and z is the obtained sparse codes where every dimension corresponds to a particular part template. In our implementation, the image is firstly sampled in multiple scales and response maps for each scale are obtained.

After the mid-level parts score maps are obtained, we employ max-pooling to build global image representation. The max-pooling is carried out in a spatial pyramid fashion[23] ($1 \times 1, 2 \times 2$ grids). The output vector of each spatial region in every scale is finally concatenated into a long vector as the final image representation. This resulting representation may be further combined with the global CNN features of the image to enhance the performance.

4 Experiments

In the experiments, we evaluate our method on two public scene recognition benchmarks, *i.e.* Scene-15 and MIT-Indoor 67. In the following subsections, we firstly in-depth analyze our model by visualizing the parts model we learn and then evaluate the recognition performances.

In the CNN network training, the architecture we adopt is similar to that used by Krizhevsky *et al.*[4]. The only difference is that the sparse connections applied in the layers $3, 4, 5$ of the network[4] (due to the model being split across 2 GPUs) are replaced with dense connections in our model. The model is firstly pre-trained on a large auxiliary dataset (ImageNet ILSVRC 2012) with image-level supervision. In particular, our pre-trained CNN obtain an average accuracy of about 4 percentage points lower on the validation set than [4], which may be due to the little difference of the architecture and learning process. To adapt the CNN to the new domain, we perform domain-specific fine-tuning on the two datasets respectively. The only difference of architecture in fine-tuning is that the number of the final softmax layer is varied from 1000 to the number of classes in the specific datasets (15 for Scene-15 and 67 for MIT-Indoor 67). The initial learning rate of fine-tuning is set as 1/10th of the initial pre-training rate. Since the domain dataset is relatively small, we augment the training set by further adding cropped, rotated and mirror samples. Both the pre-training and fine-tuning of CNN in our experiments are performed by the efficient Caffe[29] implementation.

In the part feature extraction, we employ the 4096-dimensional output of the seventh layer (fc7) as the representation. We also evaluate the performance of the sixth layer (fc6) and the final convolutional layer (pool5), which decrease the performance of about 2 and 3 percentage points respectively. The size of the part model and the regularization term α are determined by cross validation. In the construction of image-level representations, the response maps are obtained with a spatial stride of 4 pixels at four scales, defined by setting the width of the sampling regions to $40, 60, 80$ and 100 pixels. After that, we turn to train linear SVM classifiers in one-versus-others strategy and a new image is classified into the category with the largest score. The SVM regularization term is chosen via 5-fold cross validation on the training samples.

We mainly compare our method with traditional descriptors (*i.e.* HOG[2]), the method without any discriminative motivation (*i.e.* K-means and sparse coding), and related works of mid-level parts mining[10][27] and scene recognition models[23][28][25].

4.1 Results on Scene-15 dataset

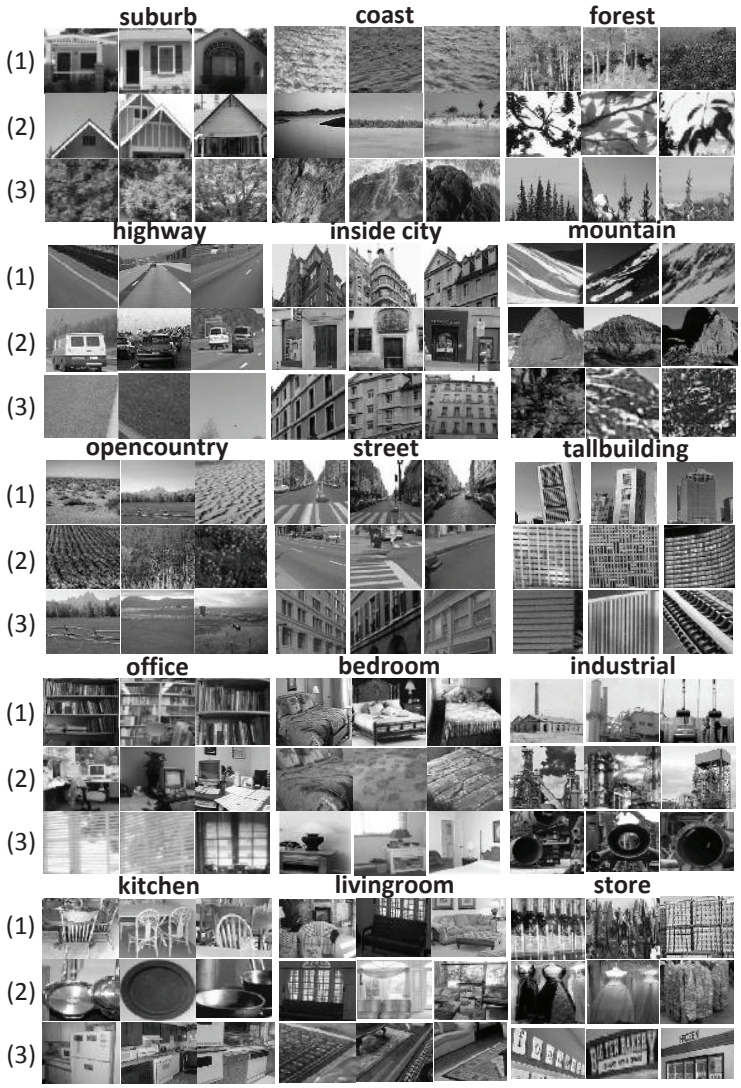

Fig. 3. The visualization of part model learned on Scene-15. We give 3 parts learned for each category, where each row in the figure corresponds to a particular part. Note how these capture key visual aspects of a typical scene.

We firstly experiment with a popular scene classification benchmark, *i.e.*, Scene-15 dataset, which is complied by several researchers[32][23]. This dataset is comprised of 15 classes of different indoor and outdoor scenes (*e.g.* kitchen,

coast, highway), including totally 4,485 gray-scale images with the number of each category ranging from 200 to 400. Following the standard experiment setup of Lazebnik *et al.*[23], we take 100 images per class for training and the rest for testing. The regularization term α in Eq. 3 is set to 5 and the sparsity factor T is 10 according to our implementation. We repeat the procedure 5 times and report the mean and standard deviation of the mean accuracy.

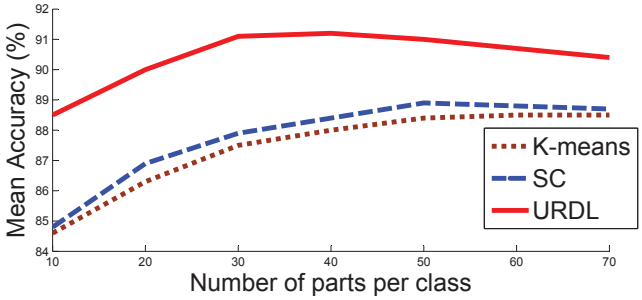

Fig. 4. The classification performance comparisons on Scene-15 with varying the number of parts per class.

In Fig. 3, we show some parts we have learned for each category. Most representative and distinctive parts we obtained are displayed for each class. To visualize the parts model, we compute the response scores of the training proposals on each part model and then sort these scores from highest to lowest. In this figure, each row of a class corresponds to a particular part and the top 3 scoring regions are displayed for each part. In another word, we visualize the part by evaluating which kinds of inputs it fires on. It is shown that our model is able to capture the key parts for each category. For example, the house and roof in the suburb, the water in the coast, the road and car in the highway and the bed in the bedroom are all captured. Also, different parts capture various visual aspects of a typical scene and different parts for different classes are visually discriminative.

In Fig. 4, we compare our model with K-means and sparse coding which neglect the label consistent regularization and investigate the variation of classification accuracy with number of parts selected per class. In this figure, "URDL" represents our unified representative and discriminative learning model, while "SC" denotes the method of sparse coding. It is shown that our method consistently outperform K-means and sparse coding which are both unsupervised clustering algorithms, demonstrating the label consistent term in our method improves the discriminative property. The mean accuracy increases as more parts are learned for the representation, but the peak is at around 40 parts per category for this dataset. This may relate to the diversity of the particular class and dataset, while too many number of parts may bring about some information redundancy or noise in the representation.

Table 1. Mean accuracy (%) comparison of parts representations on Scene-15.

Methods	Mean Accuracy
SPM[23]	81.4
Hybrid-Parts[26]	84.7
ISPR[28]	85.08 ± 0.01
CENTRIST[33]	83.88 ± 0.76
URDL(HOG feature)	85.92 ± 0.08
URDL	$\mathbf{91.15 \pm 0.02}$

Table 1 shows the performance comparisons of our proposed method with related part learning models. It is shown that our efficient method outperform some very complex models, *e.g.* Hybrid-Parts[26] and ISPR[28]. "URDL(HOG feature)" denotes that we use traditional HOG feature to represent the parts instead of CNN features. This result shows that the CNN feature significantly improves the performance of more than 5 percentage points, as CNN feature is able to represent high-level image information. We compare our approach with more public reported performances on Scene-15 in Table 2. The "CNN-SVM" denotes the method of applying the CNN features of the whole images as inputs to train SVM classifiers, while "CNN+SPM-SVM" indicates the method of extracting the CNN features of the spatial partitioned regions and then concatenating them to train SVM classifiers which is incorporated with some spatial layout information. Our best performance is reached when combining the part representation with the global CNN features. It is shown that the mean accuracy of our best result is about 96%, which beats all the previous public performances and achieves stat-of-the-art in this dataset.

Table 2. Mean accuracy (%) performance comparison on Scene-15.

Methods	Mean Accuracy
SPM[23]	81.40
Object Bank[34]	80.90
VC+VQ [35]	85.4
Hybrid-Parts+GIST-color+SP[26]	86.3
CENTRIST+LCC+Boosting[33]	87.8
LScSPM[20]	89.75 ± 0.50
IFV[36]	89.20 ± 0.09
ISPR+IFV[28]	91.06 ± 0.06
URDL	91.15 ± 0.02
CNN-SVM	92.20 ± 0.05
CNN+SPM-SVM	92.83 ± 0.04
URDL+CNN	$\mathbf{96.16 \pm 0.03}$

4.2 Results on MIT Indoor-67 dataset

Table 3. Mean accuracy (%) comparison of variant with number of parts per class on MIT Indoor-67.

Methods	Number of parts per class						
	10	20	30	40	50	60	70
URDL	57.5	58.0	58.9	60.5	61.0	61.2	60.8
URDL+CNN	67.9	68.4	69.1	70.8	71.5	71.9	71.3

The MIT Indoor-67 dataset[37] is the currently largest indoor scene recognition dataset, including 15, 620 images in 67 categories. The categories are loosely divided into stores (*e.g.* bakery, toy store), home (*e.g.* bedroom, kitchen), public spaces (*e.g.* library, subway), leisure (*e.g.* restaurant, concert hall) and work (*e.g.* hospital, TV studio). The similarity of the objects present in different indoor scenes makes MIT-Indoor an especially difficult dataset compared to outdoor scene datasets. Following the protocol of [37], we use the same training and test split where each category has about 80 training images and 20 test images. The regularization term α in Eq. 3 is set to 3 and the sparsity factor T is 20 according to our implementation. Performances are reported in terms of mean classification accuracy as in [37].

Table 4. Mean accuracy (%) performance comparisons on MIT Indoor-67.

Methods	Mean Accuracy
ROI[37]	26.05
DPM[38]	30.40
CENTRIST[33]	36.90
Object Bank[34]	37.60
Patches[10]	38.10
Hybrid-Parts [26]	39.80
BoP [35]	46.10
ISPR [28]	50.10
Hybrid-Parts+GIST-color+SP[26]	47.20
BoP+IFV [35]	63.10
ISPR+IFV[28]	68.50
CNNaug-SVM[6]	69.00
URDL	61.20
CNN-SVM	68.50
CNN+SPM-SVM	69.09
URDL+CNN	**71.90**

In the experiments on this dataset, we also evaluate the effect of the number of parts per class, as shown in Table 3. Similar to the results on Scene-15, the performances are improved as we increase the number of learned parts per class. The peak is reached at around 60 parts per category, which is more than that in Scene-15 due to the larger varieties and number of samples in MIT Indoor-67. Table 4 lists the performances comparison of our method with public results. It is noted that the performances of single-feature approaches are weak on this challenging dataset, such as ROI[37], DPM[38], CENTRIST[33], Object Bank[34], Patches[10], BoP [35] and ISPR [28]. Our performance of applying the single part representation (denoted as "URDL") beats all these methods and yields 61.20%. When combining with the CNN feature of the global image, the performance outperforms all the public results and achieves 71.90%.

Fig. 5. Examples of the learned parts templates and detections on the test set images. (a) Learned parts templates. (b) Detections on the test set images.

Fig. 6. Example images from classes with highest and lowest classification accuracy from MIT Indoor-67 dataset. The top 2 rows show example images from classes with highest accuracy, while the bottom row displays example images from classes with lowest accuracy.

Fig. 5 shows examples of the learned parts and detections on some example images in the test set. It is displayed that the learned parts capture the key aspect of a particular scene, such as the screen in the movie theater and the bed in the bedroom. Compared to the results in [27], our model may endure more appearance variants, *i.e*, the diversity in the size, viewing angle, illumination and poses. This is mainly because CNN features capture very high-level image information. In Fig. 6, we display some example images from classes with highest and lowest classification accuracy from the MIT Indoor-67 dataset. Our method performs well on most classes with little clutter (like bowling and pool inside) or scenes with consistent key parts (like florist and cloister), and less successful on classes with extremely large intra-class variation (like art studio and office). Besides different scene categories may share similar key parts, such as the computer frequently appears in both office and computer room. Overall, our model improves the recognition performances for most categories.

5 Conclusion

In this paper, we generalize the CNN features for the task of weakly supervised parts learning for the scene recognition. The CNN network is firstly pre-trained by the large auxiliary ImageNet dataset and then fine-tuned on the particular scene datasets. Then we introduce a unified part learning framework together with both representative and discriminative properties. The extensive experiments show that our model is able to capture various key and distinct parts for the typical scenes, and the recognition performances outperform related works. When combing with the CNN features of the global image, we achieves state-of-the-art performances on the two standard scene benchmarks, *i.e.* Scene-15 and MIT Indoor-67.

Acknowledgement. This work was supported by 863 Program (2014AA015104) and National Natural Science Foundation of China (61332016, 61272329, 61472422, and 61273034).

References

1. Lowe, D.: Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision **60** (2004) 91–110
2. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: CVPR. (2005)
3. Csurka, G., Dance, C., Fan, L., Willamowski, J., Bray, C.: Visual categorization with bags of keypoints. In: ECCV 2004 Workshop on statistical learning in computer vision. (2004)
4. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: NIPS. (2012)
5. Zeiler, M.D., Fergus, R.: Visualizing and understanding convolutional networks. CoRR **abs/1311.2901** (2013)

6. Razavian, A.S., Azizpour, H., Sullivan, J., Carlsson, S.: Cnn features off-the-shelf: an astounding baseline for recognition. In: CVPR. (2014)
7. Zhang, N., Paluri, M., Ranzato, M., Darrell, T., Bourdev, L.: Panda: Pose aligned networks for deep attribute modeling. In: CVPR. (2014)
8. Girshick, R., Donahue, J., Darrell, T., Malik, J.: Rich feature hierarchies for accurate object detection and semantic segmentation. In: CVPR. (2014)
9. Felzenszwalb, P.F., Girshick, R.B., McAllester, D., Ramanan, D.: Object detection with discriminatively trained part-based models. IEEE Transactions on Pattern Analysis and Machine Intelligence **32** (2010) 1627–1645
10. Singh, S., Gupta, A., Efros, A.A.: Unsupervised discovery of mid-level discriminative patches. In: ECCV. (2012)
11. Uijlings, J., van de Sande, K., Gevers, T., Smeulders, A.: Selective search for object recognition. International Journal of Computer Vision **104** (2013) 154–171
12. Li, Z., Liu, J., Yang, Y., Zhou, X., Lu, H.: Clustering-guided sparse structural learning for unsupervised feature selection. IEEE Transactions on Knowledge and Data Engineering **26** (2014) 2138–2150
13. Li, Z., Yang, Y., Liu, J., Zhou, X., Lu, H.: Unsupervised feature selection using nonnegative spectral analysis. In: AAAI. (2012)
14. Bengio, Y., Courville, A.C., Vincent, P.: Representation learning: A review and new perspectives. IEEE Transactions on Pattern Analysis and Machine Intelligence **35** (2013) 1798–1828
15. Hinton, G.E., Srivastava, N., Krizhevsky, A., Sutskever, I., Salakhutdinov, R.: Improving neural networks by preventing co-adaptation of feature detectors. CoRR **abs/1207.0580** (2012)
16. Wan, L., Zeiler, M., Zhang, S., LeCun, Y., Fergus, R.: Regularization of neural networks using dropconnect. In: ICML. (2013)
17. Sun, Y., Wang, X., Tang, X.: Deep convolutional network cascade for facial point detection. In: CVPR. (2013)
18. Toshev, A., Szegedy, C.: Deeppose: Human pose estimation via deep neural networks. In: CVPR. (2014)
19. Yang, J., Yu, K., Gong, Y., Huang, T.: Linear spatial pyramid matching using sparse coding for image classification. In: CVPR. (2009)
20. Gao, S., Tsang, I.W.H., Chia, L.T., Zhao, P.: Local features are not lonely - laplacian sparse coding for image classification. In: CVPR. (2010)
21. Wu, J., Rehg, J.M.: Beyond the euclidean distance: Creating effective visual codebooks using the histogram intersection kernel. In: ICCV. (2009)
22. Wang, P., Wang, J., Zeng, G., Xu, W., Zha, H., Li, S.: Supervised kernel descriptors for visual recognition. In: CVPR. (2013)
23. Lazebnik, S., Schmid, C., Ponce, J.: Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In: CVPR. (2006)
24. Sharma, G., Jurie, F.: Learning discriminative spatial representation for image classification. In: BMVC. (2011)
25. Xie, L., Wang, J., Guo, B., Zhang, B., Tian, Q.: Orientational Pyramid Matching for Recognizing Indoor Scenes. In: CVPR. (2014)
26. Zheng, Y., Jiang, Y.G., Xue, X.: Learning hybrid part filters for scene recognition. In: ECCV. (2012)
27. Juneja, M., Vedaldi, A., Jawahar, C.V., Zisserman, A.: Blocks that shout: Distinctive parts for scene classification. In: CVPR. (2013)
28. Lin, D., Lu, C., Liao, R., Jia, J.: Learning important spatial pooling regions for scene classification. In: CVPR. (2014)

29. Jia, Y.: Caffe: An open source convolutional architecture for fast feature embedding. http://caffe.berkeleyvision.org/ (2013)
30. Jiang, Z., Lin, Z., Davis, L.S.: Learning a discriminative dictionary for sparse coding via label consistent k-svd. In: CVPR. (2011)
31. Aharon, M., Elad, M., Bruckstein, A.: K-svd: An algorithm for designing overcomplete dictionaries for sparse representation. IEEE Transactions on Signal Processing **54** (2006) 4311–4322
32. Fei-Fei, L., Perona, P.: A bayesian hierarchical model for learning natural scene categories. In: CVPR. (2005)
33. Yuan, J., Yang, M., Wu, Y.: Mining discriminative co-occurrence patterns for visual recognition. In: CVPR. (2011)
34. Li, L.J., Su, H., Xing, E.P., Fei-Fei, L.: Object bank: A high-level image representation for scene classification and semantic feature sparsification. In: NIPS. (2010)
35. Li, Q., Wu, J., Tu, Z.: Harvesting mid-level visual concepts from large-scale internet images. In: CVPR. (2013)
36. Vedaldi, A., Fulkerson, B.: VLFeat: An open and portable library of computer vision algorithms. http://www.vlfeat.org/ (2008)
37. Quattoni, A., Torralba, A.: Recognizing indoor scenes. In: CVPR. (2009)
38. Pandey, M., Lazebnik, S.: Scene recognition and weakly supervised object localization with deformable part-based models. In: ICCV. (2011)