Learning Discriminative Part Detectors for Image Classification and Cosegmentation

Jian Sun
Xi'an Jiaotong University, INRIA, *

Jean Ponce
École Normale Supérieure, *

Abstract

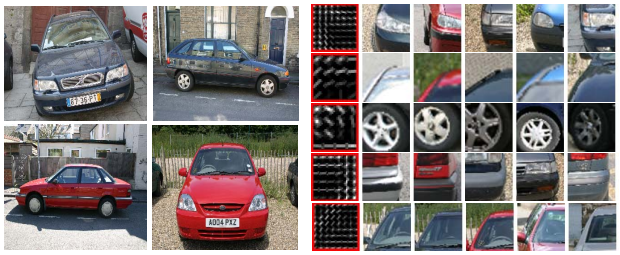
In this paper, we address the problem of learning discriminative part detectors from image sets with category labels. We propose a novel latent SVM model regularized by group sparsity to learn these part detectors. Starting from a large set of initial parts, the group sparsity regularizer forces the model to jointly select and optimize a set of discriminative part detectors in a max-margin framework. We propose a stochastic version of a proximal algorithm to solve the corresponding optimization problem. We apply the proposed method to image classification and cosegmentation, and quantitative experiments with standard benchmarks show that it matches or improves upon the state of the art.

1. Introduction

Part-based models have attracted much attention in computer vision recently [2, 4, 11, 15, 24, 32]. They represent objects or images by a set of important parts, and achieve state-of-the-art results for object detection [4, 15], action recognition [39], segmentation [2], etc.

Learning these models has, however, been a challenge. An essential question is how to efficiently learn and select object / image parts that are discriminative for the image categories of interest. Deformable part model (DPM) [15] represents objects by a set of discriminatively learned deformable parts. The positions and number of parts are heuristically initialized given the object bounding box. In poselet [4] and discriminative patch (DP) [11, 32] models, part detectors are separately learned by linear SVMs from image patch clusters. Discriminative parts are then selected by ranking the image parts and discarding unimportant ones.

In this work, we aim to learn class-specific discriminative part detectors from images of the same category (Figure 1). We propose a novel latent SVM model regularized by group sparsity to jointly select and optimize a set of discriminative part detectors in a single framework. We model part detectors as part template / threshold pairs. Given a large set of initial parts, the group sparsity regularizer forces the model to automatically select and optimize a small set of discriminative part detectors in a max-margin framework. The proposed model tends to select the parts that more frequently and strongly appear in positive training images than in the negative ones.



(a) Example images of "Car" category (b) Examples of learned part detectors

Figure 1. We learn discriminative part detectors for an image set with the same category label. The part detectors are applied to image classification and cosegmentation. (*Best viewed in color.*)

We apply the learned part detectors to image classification and cosegmentation. For classification, we encode an image by max-pooling over the responses of the learned part detectors to the image. For cosegmentation, we propose a novel model using the object cues provided by the learned part detectors in a discriminative clustering framework [16]. We achieve competitive or state-of-the-art performances on five classification and cosegmentation databases.

1.1. Related Work

Traditional image representations are primarily based on quantization of low-level features, e.g., bag-of-words (BoWs) [9] or sparse coding [38]. The image is then represented by spatially pooling the codes globally on a coarse grid (HOG [10]) or a spatial pyramid [20] for image classification. This approach achieves excellent results, but the dictionary of low-level features is rarely related to category semantics.

Object-bank [21] is an interesting attempt to represent image by high-level semantics. It represents images by pooling the responses of pre-trained object detectors to the image. This idea is also applied to action recognition [29],

*WILLOW project-team, Département d'Informatique de l'Ecole Normale Supérieure, ENS/INRIA/CNRS UMR 8548.

and achieves promising results, but it relies on a large set of pre-trained detectors to fully represent the objects / actions of interest.

Part-based models represent image by mid-level image parts. The deformable part model (DPM) [15] represents an object by a set of deformable parts learned from object bounding boxes. Strongly-supervised DPM [3] further incorporates human-annotated object parts to improve the performance. In poselet [4], a large number of object parts are learned from human-labelled keypoints in different poses. Discriminative patches (DP) [32] learn distinctive image parts using discriminative clustering. Both of the poselet and DP methods separately learn a set of part detectors using linear SVMs and select the distinctive ones by heuristically ranking their importance.

Cosegmentation [19, 25, 34] is a challenging task in computer vision, since it involves a weak form of supervision, i.e., images contain similar objects, to segment out these objects. Its multi-class extensions [17, 18] try to segment out multiple classes of objects from images. Recently, discriminative cosegmentation [8] has successfully been applied to image classification.

In this paper, we propose to learn class-specific discriminative part detectors based on category labels in a weakly supervised fashion. Contrary to part-based models [4, 15, 32] which heuristically select part detectors, our model is able to jointly select and optimize a set of discriminative part detectors in a single framework thanks to group sparsity regularization. This allows us to achieve state-of-the-art results in image classification and cosegmentation.

2. Learning Discriminative Part Detectors

In this section, we will propose a novel latent SVM model with group sparsity regularization to learn a set of discriminative part detectors for an image category.

2.1. Part Detector Definition

Given an image I, we first extract dense features at fixed intervals over the image grid. An *image part* is a box whose top-left corner is positioned at z, and it is represented by a feature vector $\Phi(I, z)$ that concatenates all the feature vectors within the box. We further define a *part detector* $\Gamma_k = (\beta_k, \tau_k)$ $(k = 1, \cdots, K)$ as a pair of *part template* β_k / *part threshold* τ_k, and define its response to image part $\Phi(I, z)$ as

$$r_z(\Gamma_k, I) = [S(\beta_k, \Phi(I, z)) - \tau_k]_+, \qquad (1)$$

where $[a]_+ = \max(a, 0)$, and $S(\beta_k, \Phi(I, z))$ is the *matching score* between the part template and the image part. In this work, we simply define the matching score as $S(\beta_k, \Phi(I, z)) = \beta_k^T \Phi(I, z)$.

Based on Eq.(1), the part detector Γ_k has non-zero response to image I at position z only when the matching score $S(\beta_k, \Phi(I, z))$ is higher than τ_k. Furthermore, we say that the part Γ_k *appears in an image* I when there exists at least one position z that satisfies $r_z(\Gamma_k, I) > 0$. Figure 2 shows examples of part detectors. As shown in this figure, after thresholding the matching scores using Eq.(1), irrelevant image parts are suppressed and only significantly similar image parts have non-zero responses.

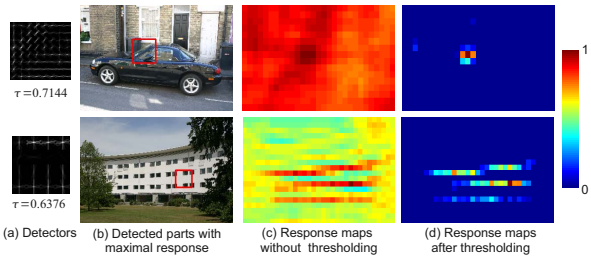


Figure 2. Examples of part detectors. With the learned part thresholds, part detectors can produce clean responses to images. (*Best viewed in color.*)

2.2. Learning Part Detectors by Group Sparsity

In this section, we aim to learn a set of image part detectors that best discriminate the positive and negative training examples for an image category. As shown in Figure 3, the input of our approach is an image set composed of positive and negative training examples. First, we automatically pick an initial set of candidate part detectors associated with the image category. They frequently appear in the positive training images but may not be discriminative. Then we use a novel latent SVM model to select and optimize final part detectors with group sparsity regularization.

2.3. Initialization of Part Detectors

To initialize the candidate part detectors for an image category, we randomly crop a large number of image parts (approximately ten thousands) from the positive training images. Then we perform k-means clustering (600 clusters in our implementation) over these sampled image parts. This is similar to the construction of a visual word dictionary in BoWs. We only retain sufficiently large clusters of size 10 or more. Assume that we have K clusters of image parts, then we initialize K part detectors $\{\Gamma_k\}_{k=1}^K$, and each part detector $\Gamma_k = \{\beta_k, \tau_k\}$ is defined as a pair of part template β_k and part threshold τ_k which are taken as the k-th cluster center and zero value respectively.

2.4. Learning Discriminative Part Detectors

With the above initialization, we now learn a set of part detectors that best discriminate the positive and negative training images. We require that the learned part detectors should appear more frequently and strongly in the positive training images than in the negative ones.

Figure 3. An illustration of our learning framework. Given a training set of positive and negative images for an image category, we first initialize a set of part detectors as discussed in Section 2.3. Then we jointly select and optimize a set of part detectors, i.e., part template / threshold pairs, by a novel latent SVM model regularized by group sparsity as discussed in Section 2.4.

Before introducing our learning method, let us first define the confidence of image I belonging to the current category given class-specific part detectors $\Gamma = \{\Gamma_k\}_{k=1}^K$:

$$g(I,\Gamma) = \sum_{k=1}^{K} [\beta_k^T \Phi(I, z_k) - \tau_k]_+, \tag{2}$$

where z_k is a latent variable indicating the image part position with maximum response:

$$z_k = \text{argmax}_{z \in \Omega_I} \, \beta_k^T \Phi(I, z), \tag{3}$$

and Ω_I defines the set of all possible part positions in I. Observe from Eq.(2) that $g(I,\Gamma) \geq 0$ is defined as sum of the maximum responses of all the part detectors to image I. Image I thus has higher confidence belonging to this category when more parts appear in I and have higher responses.

Next we learn part detectors using a latent SVM model with group sparsity regularization. The basic idea is to jointly select and optimize the part detectors by maximizing the margin of the confidence value $g(I,\Gamma)$ on positive and negative training images. Denote the training image set as $\{I_n, y_n\}_{n=1}^N$ where $y_n = 1$ if I_n belongs to the category and otherwise $y_n = -1$. The cost function is defined as:

$$E(\Gamma, b) = \frac{1}{N} \sum_{n=1}^{N} L(g(I_n,\Gamma), y_n, b) + \lambda R(B), \tag{4}$$

where $B = \{\beta_k\}_{k=1}^K$ is the set of all part templates and L is the squared hinge loss function:

$$L\left(g(I,\Gamma), y, b\right) = [1 - y(g(I,\Gamma) + b)]_+^2, \tag{5}$$

and b is the bias term of SVM. We have chosen this function because it is differentiable w.r.t. g and b. We could have used other differentiable losses, e.g., a logistic function.

$R(B)$ is a regularization term over the part templates. We impose group sparsity [40] over part templates, where each template is considered as a group. This regularization forces the algorithm to automatically select a few discriminative part detectors with non-zero templates from a large set of candidate part detectors. Typical group sparsity terms include $l_{1,2}$ and $l_{1,\infty}$ regularizers [40]. We choose the $l_{1,2}$ structured sparsity norm in this paper, i.e., $R(B) = \sum_{k=1}^K ||\beta_k||_2$, which is the sum of l_2 norm of part templates, and is convex w.r.t. B. In summary, we learn the discriminative part detectors by solving:

$$\underset{\Gamma, b}{\text{argmin}} \left\{ \frac{1}{N} \sum_{n=1}^{N} [1 - y_n(g(I_n,\Gamma) + b)]_+^2 + \lambda \sum_{k=1}^{K} ||\beta_k||_2 \right\}, \tag{6}$$

where $g(I_n,\Gamma)$ depends on latent variables in Eq.(3).

The above SVM model tries to enforce that $g(I,\Gamma)+b \geq 1$ if I is positive training image, and $g(I,\Gamma)+b \leq -1$ if I is negative training image. This forces the learned part detectors to have larger responses to positive training images than to negative ones. It implies that the learned part detectors should be *discriminative*, i.e., more frequently and strongly trigger in the positive training images than in the negative ones. With group sparsity regularization, the optimization procedure will automatically discard the less discriminative part detectors among the initial ones.

Let us briefly compare our model to the latent SVM in [15]. First, our proposed latent SVM model is regularized by group sparsity which is able to automatically select discriminative part detectors from a large pool of initial detectors. Second, our learned part detectors are pairs of part template and part threshold. With the part thresholds, parts are not required to appear in every image of the category, which makes the detectors robust to intra-class variations caused by poses, sub-categories, etc.

3. Optimization Algorithm

The latent SVM model of Eq.(6) is semi-convex [15] w.r.t. the part detectors Γ, i.e., it is convex for the negative examples and non-convex for the positive examples. This can be justified by the following facts. First, $g(I,\Gamma)$ is convex w.r.t. $\Gamma = \{\beta_k, \tau_k\}_{k=1}^K$. This can be easily shown by noting that $g(I,\Gamma) = \sum_{k=1}^K \max\{\tilde{\beta}_k^T \tilde{\Phi}(I, z_k), 0\}$ if denoting $\tilde{\beta}_k = [\beta_k^T, \tau_k]^T$ and $\tilde{\Phi}(I, z_k) = [\Phi^T(I, z_k), -1]^T$, which is the maximum of linear functions. Second, the cost function in Eq.(6) is convex and non-decreasing w.r.t. $g(I,\Gamma)$ if I is a negative example (i.e., $y = -1$). Therefore the cost is convex w.r.t. Γ for the negative examples.

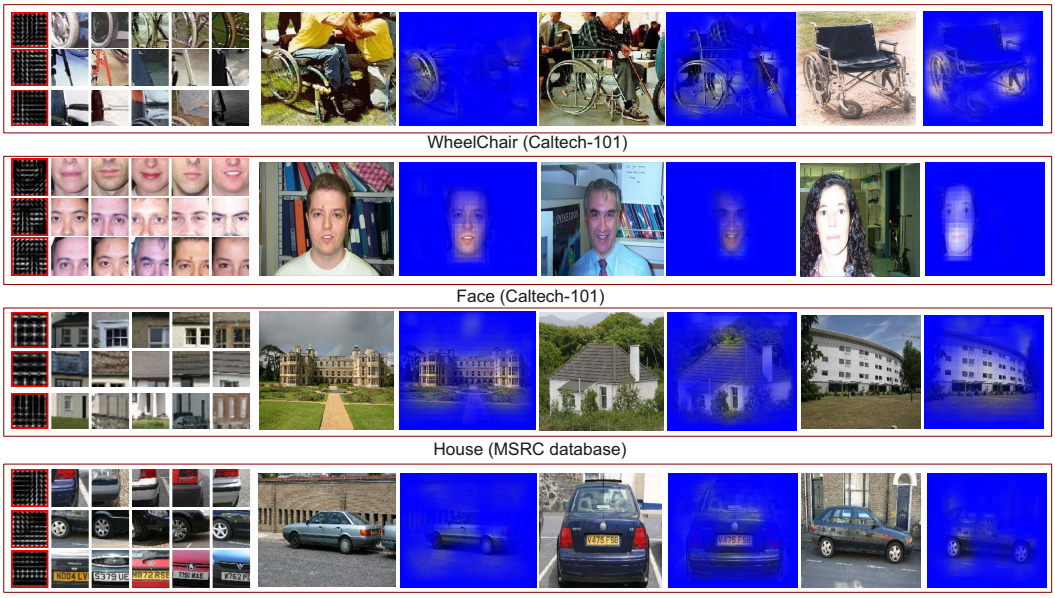


Figure 4. Examples of learned part detectors, detected parts and total response maps of part detectors to images. The learned part detectors have higher responses to the discriminative regions in each category. Response maps are shown as the original images masked by the linearly normalized total response maps in range of [0, 1]. (*Best viewed in color. More examples are shown in supplementary material.*)

However, it is non-convex for the positive examples.

Following [15], we optimize Eq. (6) by iteratively performing the following two steps. First, we update the latent variables for all the positive examples based on Eq. (3). Second, given the set of latent variables for all the positive examples (denoted as Z_p), we optimize part detectors $\{\beta_k, \tau_k\}_{k=1}^K$ and bias term b by minimizing the convex cost $E(\Gamma, b; Z_p)$ which is the cost function in Eq.(6) with fixed latent variables for positive examples. We stop the iteration when a maximal number of iterations is reached or the parameters do not change significantly any more.

We now discuss how to minimize $E(\Gamma, b; Z_p)$ given Z_p. This cost function is smooth for b and piecewise-smooth for Γ. Therefore, we utilize a gradient descent method to optimize b and a subgradient method to optimize $\Gamma = \{\beta_k, \tau_k\}_{k=1}^K$ simultaneously. Due to the group sparsity regularization for $\{\beta_k\}_{k=1}^K$, we utilize a proximal method [13] to optimize β_k. It is known to be an effective approach to the optimization of convex loss functions with sparse regularization, and the basic procedure is to update the parameters using a proximal operator which can be shown to be $\text{Prox}_\mu(\boldsymbol{\beta_k}) = \frac{1}{||\boldsymbol{\beta_k}||_2}\boldsymbol{\beta_k}[||\boldsymbol{\beta_k}||_2 - \mu]_+$ for $l_{1,2}$ regularizer.

In summary, we minimize the energy $E(\Gamma, b; Z_p)$ by iteratively updating the parameters: ${\beta_k}^{t+1} = \text{Prox}_{\lambda\gamma}(\beta_k^t - \gamma\frac{1}{N}\sum_{n=1}^N \frac{\partial L_n}{\partial {\beta_k}^t})$, $b^{t+1} = b^t - \gamma\frac{1}{N}\sum_{n=1}^N \frac{\partial L_n}{\partial b^t}$, ${\tau_k}^{t+1} = \tau_k^t - \gamma\frac{1}{N}\sum_{n=1}^N \frac{\partial L_n}{\partial \tau_k}$, where γ is the step-size, and $L_n = L(g(I_n, \Gamma), y_n, b)$. The involved gradient (w.r.t. b) and subgradients (w.r.t. β_k, τ_k) are computed as:

$$\begin{aligned}
\frac{\partial L_n}{\partial b} &= \begin{cases} -\eta_n y_n & \text{if } y_n(g(I_n,\Gamma)+b) < 1 \\ 0 & \text{otherwise,} \end{cases} \\
\frac{\partial L_n}{\partial \beta_k} &= \begin{cases} -\eta_n y_n \Phi(I_n, z_{n,k}) & \text{if } \mathbf{C} \text{ is satisfied} \\ 0 & \text{otherwise,} \end{cases} \\
\frac{\partial L_n}{\partial \tau_k} &= \begin{cases} \eta_n y_n & \text{if } \mathbf{C} \text{ is satisfied} \\ 0 & \text{otherwise,} \end{cases}
\end{aligned} \tag{7}$$

where $\eta_n = 2(1 - y_n(g(I_n, \Gamma) + b))$, $z_{n,k}$ is the k-th latent variable for image I_n, $\mathbf{C}$ denotes the conditions of $\beta_k^T \Phi(I_n, z_{n,k}) > \tau_k$ and $y_n(g(I_n, \Gamma) + b) < 1$. The optimization of $E(\Gamma, b; Z_p)$ is a large-scale and high-dimensional convex optimization problem. To make it tractable, we propose to use a stochastic algorithm in which a subset (six random samples) of training images are sampled to approximate the gradients / subgradients [13].

After optimization, non-discriminative part templates are set to zero due to the $l_{1,2}$ regularization. We discard these part detectors with zero part templates and derive a set of discriminative part detectors. To illustrate the learned part detectors, we define the *response map of a part detector* Γ_k to an image I as the weighted sum of all the detected parts appearing in the image pyramid, i.e.,

$$R(\Gamma_k, I) = \sum_s \sum_{z \in \Omega_{I^s}} r_z(\Gamma_k, I^s) M_z(I^s), \tag{8}$$

where I^s is the image at scale s, $r_z(\Gamma_k, I^s)$ is the response value defined in Eq.(1), $M_z(I^s)$ is the binary mask of I^s

indicating the region occupied by image part located at position z. The part mask $M_z(I^s)$ is re-scaled by $\frac{1}{s}$, therefore the response map $R(\Gamma_k, I)$ has the same resolution as I. In our implementation, we construct an image pyramid in five scaling factors, i.e., $s \in \{2^{-1}, 2^{-\frac{1}{2}}, 1, 2^{\frac{1}{2}}, 2\}$.

Figure 4 shows examples of learned part detectors and detected parts. As shown in Figure 4(a), the learned detectors are discriminative for the categories considered, e.g., wheelchairs, faces, buildings and cars. Figure 4(b) shows *total response maps* of part detectors by summing $R(\Gamma_k, I)$ over all the learned part detectors. It shows that the learned part detectors have large responses to the salient regions which are discriminative for the image category, and have low responses to the cluttered backgrounds. It indicates that our algorithm can effectively derive a set of discriminative part detectors and discard the unimportant ones. Please see *supplementary material* for more examples.

4. Applications

Discriminative part detectors provide a mid-level and discriminative representation for an image category. We now apply them to image classification and cosegmentation.

4.1. Image Classification

Given an image database, we learn class-specific part detectors for each category using one-vs-all training. We denote all the learned part detectors from different categories as $\Gamma = \{\Gamma_k\}_{k=1}^K$, K is the total number of part detectors. Based on our learning method for part detectors, an image I can be naturally encoded by a vector of codes $\{c_k\}_{k=1}^K$, and each code $c_k = [\max_{z \in \Omega_I} \beta_k^T \Phi(I, z) - \tau_k]_+$, which is the max-pooling over the responses of part detector Γ_k to all the image parts in I.

Following object-bank [21], we improve the above coding method by the following steps. We resize the image resolution in five scaling factors ($\{2^{-1}, 2^{-\frac{1}{2}}, 1, 2^{\frac{1}{2}}, 2\}$) to capture image parts in different scales. Then for each image in each scale, we use spatial pyramid matching (SPM) [20] dividing the image region into spatial cells in three levels. Finally, the image I is coded by concatenating all the codes computed over the image regions in each spatial cell and each scale. This coding method will produce a feature vector with the length of $5MK$, where M is the number of cells in spatial pyramid. Given the image codes, we use a linear SVM with squared hinge loss function to produce the classification results.

4.2. Image Cosegmentation

For cosegmentation, we aim to segment the common objects in an image set with the same category label. Given an image set $\{I_n\}_{n=1}^N$ with the same category of objects, we first learn discriminative part detectors $\Gamma = \{\Gamma_k\}_{k=1}^K$ from a training set with the input images as positive examples and a set of diverse background images as negative examples. As shown in Figure 4(b) and Figure 5(b), the discriminative part detectors response more strongly and frequently in the common objects of the image set, which provides a high-level object cue for cosegmentation.

Figure 5. Cosegmentation example (the image comes from "sign" category of MSRC database). (b) Total response map. (c) Initial segmentation mask. (d) Final segmentation boundary.

Given image I, we aim to assign labels $X = \{\boldsymbol{x}_i\}$ to pixels, and $\boldsymbol{x}_i = 1$ for foreground pixel and $\boldsymbol{x}_i = 0$ for background pixel. It can be considered as a weakly supervised clustering problem. Discriminative clustering has achieved state-of-the-art performance on cosegmentation [16, 17]. In this work, we design a novel cosegmentation algorithm by embedding the object cue provided by part detectors into the discriminative clustering framework.

We denote image feature as v_i for pixel i, and $\Psi(v_i)$ is a mapping of v_i into a high-dimensional Hilbert space $\mathcal{F}$. Discriminative clustering [16] tries to jointly infer the segment labels X and non-linear separating surface $f \in \mathcal{F}$ based on kernel SVM by minimizing:

$$E_c(X, f, d|I) = \frac{1}{|\Omega_I|} \sum_{i \in \Omega_I} [1 - \boldsymbol{x}_i(f^T \Psi(v_i) + d)]_+ + \alpha_c ||f||^2, \quad (9)$$

where d is bias term, and α_c is regularization parameter.

Discriminative clustering is an unsupervised method for segmentation. In our approach, we incorporate the object cue provided by part detectors and label smoothness into the above formulation, then the optimization problem is:

$$\min_{X,f,d} E(X, f, d) = E_c(X, f, d|I) + \frac{1}{|\Omega_I|} \sum_{i \in \Omega_I} [E_o(\boldsymbol{x}_i|\Gamma, I) + \alpha_s \sum_{j \in N(i)} E_s(\boldsymbol{x}_i, \boldsymbol{x}_j|I)], \quad (10)$$

where $N(i)$ is the neighborhood of i. E_o is defined based on the common object cue shared by the image set:

$$E_o(\boldsymbol{x}_i|\Gamma, I) = \begin{cases} R_i(\Gamma_k, I) - \zeta & \text{if } \boldsymbol{x}_i = 0 \\ 0 & \text{if } \boldsymbol{x}_i = 1, \end{cases} \quad (11)$$

where R_i is the value of response map in Eq.(8) at pixel i. Obviously, this model prefers to assign foreground label to pixel with $\sum_k R_i(\Gamma_k, I) > \zeta$, and ζ is automatically set for each image by enforcing that pixels above this threshold occupy at most 40% of the image area. E_s

is the smoothness term defined as $E_s(\boldsymbol{x}_i, \boldsymbol{x}_j|I) = |\boldsymbol{x}_i - \boldsymbol{x}_j| \exp(-\frac{||v_i^c - v_j^c||_2^2}{2\sigma})$ [28] where v_i^c is color vector at pixel i, and σ is the mean of the squared distances between adjacent colors over the image. E_s is submodular and encourages the segmentation boundary to align with strong edges.

We optimize Eq.(10) by alternatively inferring the SVM parameters $\{f, d\}$ and the segmentation label X. Given X, $\{f, d\}$ can be found by minimizing E_c since it is the only term that depends on f and d in Eq.(10). This can be done by a standard kernel SVM algorithm. Given $\{f, d\}$, the segmentation label X can be computed by minimizing Eq.(10) with fixed f, d, which can be efficiently optimized by graph cuts [7]. We initialize X by solving:

$$\text{argmin}_X \{ \sum_{i \in \Omega_I} [E_o(\boldsymbol{x}_i|\Gamma, I) + \alpha_s \sum_{j \in N(i)} E_s(\boldsymbol{x}_i, \boldsymbol{x}_j|I)] \}, \tag{12}$$

which is based on the object cue and label smoothness.

In our implementation, feature vector v is the concatenation of HOG feature v^h and color feature v^c with length of L_h and L_c respectively. Color values are scaled to [0, 1]. In kernel SVM, we utilize kernel $K(v_i, v_j) = \exp(-\lambda_c(\frac{1}{L_h}||v_i^h - v_j^h||_2^2 + \frac{1}{L_c}||v_i^c - v_j^c||_2^2))$ with $\lambda_c = 5$. It is a valid kernel since it is multiplication of two radial basis kernels. When we optimize X using graph cuts, we utilize superpixels [1] to define the graph, in which per-pixel costs E_c, E_o are averaged in each superpixel. Figure 5 shows an example of initial and final cosegmentation results.

5. Experiments

To learn part detectors, we extract dense HOG features at eight-pixel intervals, and each image part is represented as concatenation of all HOG features in the corresponding region. We utilize multiple sizes of part templates (8×8, 6×6, 4×4 feature cells) to capture features at different scales. The discriminative part detectors are learned in one-vs-all mode for each database. The regularization parameter λ controls the sparsity of the solution. We have fixed it to 0.005 in all experiments, which retains 10-15% of the part detectors. Please see next section for a preliminary investigation of the effect of λ on the number of parts and classification performance.

5.1. Image Classification

We test our classification method on four representative image databases for scene categorization (15-Scenes [20], MIT-indoor [27]), object recognition (Caltech-101 [14]) and event categorization (UIUC-Sports [22]). We use mean average precision (mAP) to measure the accuracy.

Table 1 shows comparison results on 15-Scenes (100 training images per category). Our discriminative part detectors perform significantly better than the low-level visual words in [20, 38] and high-level object detectors in [21].

Table 1. Comparison on 15-Scenes database.

Single feature		*Multiple features*	
Methods	mAP	Methods	mAP
Sparse-coding [38]	80.3 ± 0.9	Object-bank [21]	80.9
SPM [20]	81.4 ± 0.5	BSPR [36]	$\mathbf{88.9 \pm 0.6}$
Graph-matching [12]	82.1 ± 1.1	Su et al. [33]	87.8 ± 0.5
DSS [31]	$\mathbf{85.5 \pm 0.6}$	Xiao et al. [35]	88.1
LPR [30]	$\mathbf{85.8}$		
Ours	$\mathbf{86.0 \pm 0.8}$		

Our algorithm performs well compared to the algorithms using single feature. The state-of-the-art result on this database is 88.9% in BSPR [36] which is based on multiple features and dense sampling of pooling regions. Our method can potentially be improved by incorporating comparable advanced pooling methods beyond SPM.

Table 2. Comparison on MIT-indoor 67 scenes categorization.

Methods	mAP
DPM [26]	30.4
DPM + GIST + SPM [26]	43.1
Object-bank [21]	37.6
DiscPatches [32]	38.1
LPR-LIN [30]	44.8
Hybrid-parts [41]	39.8
Hybrid-parts + GIST + SPM [41]	47.2
Ours	**51.4**

Table 2 shows the comparison of our method with state-of-the-art algorithms on the challenging MIT-indoor database. For each category, we follow the same setting as in [27] and use approximately 80 images for training and 20 images for testing. We learn a total of 4926 (12% of the number of initial detectors) part detectors for 67 classes, and achieve 51.43% in mAP using a single HOG feature. Compared to discriminative patches learned by discriminative clustering [32] (14070 patches are learned), we perform significantly better, which shows the advantage of our learning method. Per-category accuracies are shown in *supplementary material*.

Table 3. Comparison on UIUC-Sports database.

Methods	mAP
Hybrid-parts [41]	84.5
Object-bank [21]	76.3
Sparse-coding [38]	82.7 ± 1.74
LPR [30]	86.25
LSA[23]	82.3 ± 1.84
Ours	$\mathbf{86.4 \pm 0.88}$

Table 4. Comparison on Caltech-101 database using single feature.

Methods	mAP
SPM [20]	64.4 ± 0.8
Macro-feature [5]	75.7 ± 1.1
Sparse-coding [38]	73.2 ± 0.5
Multi-way pooling[6]	77.1 ± 0.7
Graph-matching[12]	$\mathbf{80.3 \pm 1.2}$
Ours	78.8 ± 0.5

Tables 3 and 4 show comparison results on UIUC-Sports and Caltech-101, in which 70 and 30 images per-category are used for training respectively. Our algorithm achieves state-of-the-art results on UIUC-Sports and competitive results on Caltech-101 using a single feature[1]. Graph-

[1]The state-of-the-art result on Caltech-101 using multiple features is 84.3% achieved in [37] by multiple kernel learning.

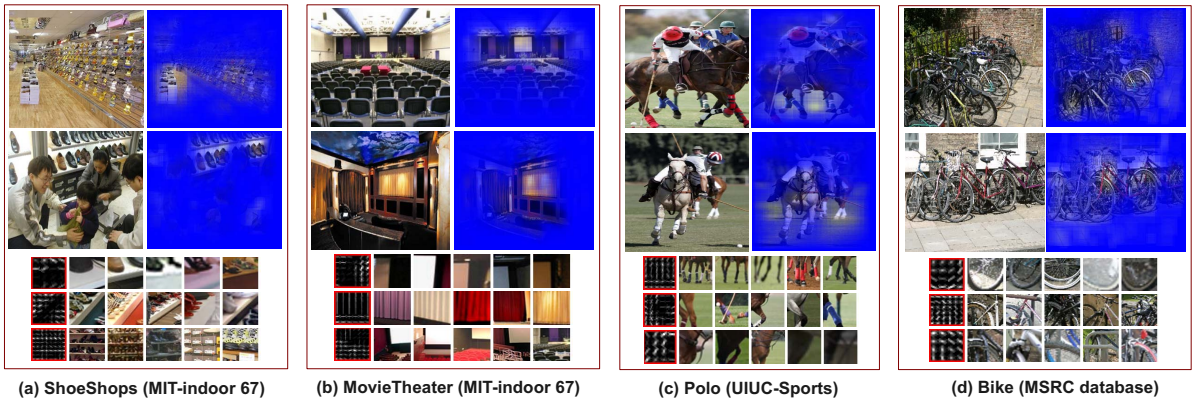


Figure 6. Examples of class-specific part detectors and their total response maps to images. (*Best viewed in color.*)

matching [12] performs better than ours on Caltech-101 using kernel method defined by dense matching. However it achieves significantly lower results on 15-Scenes in Table 1, probably because objects in Caltech-101 are well aligned and can be densely matched with higher accuracy.

Figure 6 shows examples of learned part detectors and their total response maps. As shown in Figure 6(a,b), the shoes and movie screens are effectively detected by our learned part detectors for categories of "ShoeShops" and "MovieTheater" in MIT-indoor database. Our learned part detectors can effectively detect the discriminative image parts and suppress the cluttered backgrounds.

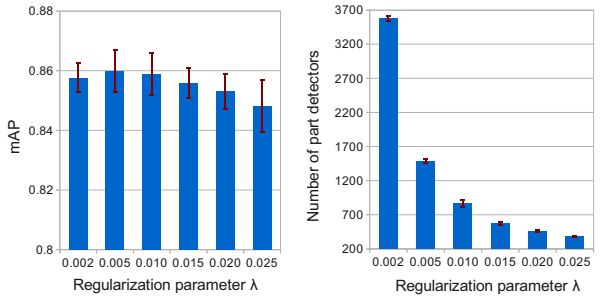


Figure 7. The effect of regularization parameter on the classification performance (tested on 15-Scenes database).

Effect of regularization parameter λ on performance: λ in Eq.(6) determines the number of selected part detectors. Figure 7 shows the effect of λ on the performance tested on 15-Scenes database. With the increase of λ, we observe that the number of learned part detectors decreases fast, and the classification accuracy increases then decreases, however, is quite stable to λ when $0.002 \leq \lambda \leq 0.015$.

5.2. Image Cosegmentation

We test our algorithm on MSRC database and compare with the state of the art. This database is commonly used for testing binary cosegmentation algorithms [16, 19, 25]. The parameters of cosegmentation model in Eq. (10) are set as $\alpha_c = 1, \alpha_s = 0.25$. We utilize intersection-over-union score as in [17] to measure the segmentation accuracy. Table 5 shows comparison results between our algorithm and the state-of-the-art cosegmentation algorithms. The algorithm of [25] fails to converge on four classes. Our initial segmentation based on object cues alone already achieves better results than the method in [19]. Our full algorithm achieves the highest accuracy on this database. Figure 8 shows examples of our cosegmentation results.

Table 5. Comparison of the proposed cosegmentation method with Joulin et al. [16, 17], Kim et al. [19], and Mukherjee et al. [25]. "Ours_init" indicates the initial segmentation of our approach.

Datasets	Images	[16]	[17]	[19]	[25]	Ours_init	Ours
Bike	30	42.3	43.3	29.9	42.8	46.5	**50.7**
Bird	30	33.2	**47.7**	29.9	–	22.8	31.0
Car	30	59.0	59.7	37.1	52.5	55.0	**61.5**
Cat	24	30.1	31.9	24.4	5.6	36.5	**48.0**
Chair	30	37.6	39.6	28.7	39.4	39.4	**48.9**
Cow	30	45.0	**52.7**	33.5	26.1	38.2	45.6
Dog	26	41.3	41.8	33.0	–	32.4	**46.6**
Face	30	**66.2**	70.0	33.2	40.8	48.4	50.3
Flower	30	50.9	51.9	40.2	–	50.2	**75.7**
House	30	50.5	51.0	32.2	**66.4**	51.1	61.5
Plane	30	21.7	21.6	25.1	**33.4**	28.2	28.1
Sheep	30	60.4	66.3	60.8	45.7	47.8	**65.2**
Sign	30	55.2	58.9	43.2	–	50.9	**69.9**
Tree	30	60.0	67.0	61.2	55.9	55.8	**70.1**
Average		46.7	50.2	36.6	–	43.1	**53.8**

6. Conclusion

We have proposed a novel latent SVM model to learn discriminative part detectors for image categories. It achieves promising results for image classification and cosegmentation. We have shown that discriminative part detectors provide mid-level cues to determine the position of objects. In the future, we are interested in organizing these part detectors in graph structure for object detection.

Acknowledgement

This work was supported by the European Research Council (VideoWorld project). Jian Sun was partially supported by the 973 program (2013CB329404), NSFC projects (61003144, 11131006) and NCET-12-0442.

Figure 8. Cosegmentation results on categories of "Tree", "Face", "Car", "Bike", "Sheep", "Sign" in MSRC database.

References

[1] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Susstrunk. Slic superpixels compared to state-of-the-art superpixel methods. *IEEE T. PAMI*, 34(11):2274–2282, 2012.

[2] P. Arbeláez, B. Hariharan, C. Gu, S. Gupta, L. Bourdev, and J. Malik. Semantic segmentation using regions and parts. In *CVPR*, 2012.

[3] H. Azizpour and I. Laptev. Object detection using strongly-supervised deformable part models. In *ECCV*, 2012.

[4] L. Bourdev and J. Malik. Poselets: Body part detectors trained using 3d human pose annotations. In *ICCV*, 2009.

[5] Y. Boureau, F. Bach, Y. LeCun, and J. Ponce. Learning mid-level features for recognition. In *CVPR*, 2010.

[6] Y. Boureau, N. Le Roux, F. Bach, J. Ponce, and Y. LeCun. Ask the locals: multi-way local pooling for image recognition. In *ICCV*, 2011.

[7] Y. Boykov and V. Kolmogorov. An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. *IEEE T. PAMI*, 26(9):1124–1137, 2004.

[8] Y. Chai, E. Rahtu, V. Lempitsky, L. Van Gool, and A. Zisserman. Tricos: A tri-level class-discriminative co-segmentation method for image classification. In *ECCV*, 2012.

[9] G. Csurka, C. Dance, L. Fan, J. Willamowski, and C. Bray. Visual categorization with bags of keypoints. In *ECCV workshop on statistical learning in computer vision*, 2004.

[10] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *CVPR*, 2005.

[11] C. Doersch, S. Singh, A. Gupta, J. Sivic, and A. Efros. What makes paris look like paris? *ACM TOG*, 31(4):101, 2012.

[12] O. Duchenne, A. Joulin, and J. Ponce. A graph-matching kernel for object categorization. In *ICCV*, 2011.

[13] J. Duchi and Y. Singer. Efficient learning using forward-backward splitting. In *NIPS*, 2009.

[14] L. Fei-Fei, R. Fergus, and P. Perona. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. In *CVPR workshop on generative-model based vision*, 2004.

[15] P. Felzenszwalb, R. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part-based models. *IEEE T. PAMI*, 32(9):1627–1645, 2010.

[16] A. Joulin, F. Bach, and J. Ponce. Discriminative clustering for image co-segmentation. In *CVPR*, 2010.

[17] A. Joulin, F. Bach, and J. Ponce. Multi-class cosegmentation. In *CVPR*, 2012.

[18] G. Kim and E. P. Xing. On multiple foreground cosegmentation. In *CVPR*, 2012.

[19] G. Kim, E. P. Xing, L. Fei-Fei, and T. Kanade. Distributed cosegmentation via submodular optimization on anisotropic diffusion. In *ICCV*, 2011.

[20] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *CVPR*, 2006.

[21] L. Li, H. Su, E. Xing, and L. Fei-Fei. Object bank: A high-level image representation for scene classification and semantic feature sparsification. In *NIPS*, 2010.

[22] L.-J. Li and L. Fei-Fei. What, where and who? classifying events by scene and object recognition. In *ICCV*, 2007.

[23] L. Liu, L. Wang, and X. Liu. In defense of soft-assignment coding. In *ICCV*, 2011.

[24] M.Juneja, A. Vedaldi, C. V. Jawahar, and A. Zisserman. Blocks that shout: Distinctive parts for scene classification. In *CVPR*, 2013.

[25] L. Mukherjee, V. Singh, and J. Peng. Scale invariant cosegmentation for image groups. In *CVPR*, 2011.

[26] M. Pandey and S. Lazebnik. Scene recognition and weakly supervised object localization with deformable part-based models. In *ICCV*, 2011.

[27] A. Quattoni and A.Torralba. Recognizing indoor scenes. In *CVPR*, 2009.

[28] C. Rother, V. Kolmogorov, and A. Blake. Grabcut: Interactive foreground extraction using iterated graph cuts. *ACM TOG*, 23(3):309–314, 2004.

[29] S. Sadanand and J. J. Corso. Action bank: A high-level representation of activity in video. In *CVPR*, 2012.

[30] F. Sadeghi and M. F. Tappen. Latent pyramidal regions for recognizing scenes. In *ECCV*, 2012.

[31] G. Sharma, F. Jurie, and C. Schmid. Discriminative spatial saliency for image classification. In *CVPR*, 2012.

[32] S. Singh, A. Gupta, and A. Efros. Unsupervised discovery of mid-level discriminative patches. In *ECCV*, 2012.

[33] Y. Su and F. Jurie. Visual word disambiguation by semantic contexts. In *ICCV*, 2011.

[34] S. Vicente, C. Rother, and V. Kolmogorov. Object cosegmentation. In *CVPR*, 2011.

[35] J. Xiao, J. Hays, K. A. Ehinger, A. Oliva, and A. Torralba. Sun database: Large-scale scene recognition from abbey to zoo. In *CVPR*, 2010.

[36] S. Yan, X. Xu, D. Xu, S. Lin, and X. Li. Beyond spatial pyramids: A new feature extraction framework with dense spatial sampling for image classification. In *ECCV*, 2012.

[37] J. Yang, Y. Li, Y. Tian, L. Duan, and W. Gao. Group-sensitive multiple kernel learning for object categorization. In *CVPR*, 2009.

[38] J. Yang, K. Yu, Y. Gong, and T. Huang. Linear spatial pyramid matching using sparse coding for image classification. In *CVPR*, 2009.

[39] B. Yao, X. Jiang, A. Khosla, A. Lin, L. Guibas, and L. Fei-Fei. Human action recognition by learning bases of action attributes and parts. In *ICCV*, 2011.

[40] M. Yuan and Y. Lin. Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society: Series B*, 68(1):49–67, 2005.

[41] Y. Zheng, Y.-G. Jiang, and X. Xue. Learning hybrid part filters for scene recognition. In *ECCV*, 2012.