Neurocomputing 139 (2014) 415–422

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Spatial modeling via feature co-pooling and SG grafting

Feng Liu [a,*], Yongzhen Huang [b], Liang Wang [b], Wankou Yang [a], Changyin Sun [a]

[a] School of Automation, Southeast University, Nanjing 210096, China
[b] National Laboratory of Pattern Recognition (NLPR), Institute of Automation, Chinese Academy of Sciences (CASIA), Beijing 100190, China

ARTICLE INFO

Article history:
Received 27 May 2013
Received in revised form
22 January 2014
Accepted 10 February 2014
Communicated by X. Li
Available online 8 April 2014

Keywords:
Object classification
Spatial modeling
Feature selection

ABSTRACT

Spatial information is an important cue for visual object analysis. Various studies in this field have been conducted. However, they are either too rigid or too fragile to efficiently utilize such information. In this paper, we propose to model the distribution of objects' local appearance patterns by using their co-occurrence at different spatial locations. In order to represent such a distribution, we propose a flexible framework called spatial feature co-pooling, with which the relations between patterns are discovered. As the final representation resulted from our framework is of high dimensionality, we propose a semi-greedy (SG) grafting algorithm to select the most discriminative features. Experimental results on the CIFAR 10, UIUC Sports and VOC 2007 datasets show that our method is effective and comparable with the state-of-art algorithms.

1. Introduction

Spatial modeling is of great significance for both human visual system and computer vision. As shown in Fig. 1, if an object is described as 'grass around, wool in the middle', we will easily regard it as a sheep. If a sky pattern emerges in the upper part of a picture, we tend to consider this picture as an outdoor scene. Moreover, if such a pattern exists at both upper and lower positions of an image with a rigid object in the middle, it is highly possible to be inferred as an aeroplane. It is also easy to predict a boat by the water below, and judge a car by the road around. Besides, similar local patterns may appear in different locations of an image. For example, the two pink bounding boxes in the first image are both wool-like patterns. The process of using these meaningful appearance patterns and location distributions is specially called 'spatial modeling' in this paper.

There have been some studies aiming at modeling features' spatial distributions. We roughly divide them into four categories. The first one is building pooling regions according to some rules, and the final representation is the concatenation of each regions' pooling result [1,2]. Such methods are robust to small shifting of local features. However, the partition of regions is too rigid to dig out more information from features' locations, e.g., the relationship of two spatial non-adjacent features. The second category is to directly learn the spatial distribution from features' positions [3,4]. These approaches perform not well for the poorly aligned datasets. The third category exploits the co-occurrence information between visual vocabularies [5], but such a representation is complex and ignores the spatial distribution of features. The last one is sampling discriminative patches when extracting low level features [6,7]. However, most approaches use dense sampling for speed concerns.

Different from the above-mentioned methods, we propose a spatial feature co-pooling (SCP) framework in this paper. It first divides an image into several blocks. For each block, a standard pooling method is employed. Then adjacent blocks are pooled together to form a region, which is of multiple sizes to capture any patterns of different scales. Regions in different locations are further merged, and finally all of them are concatenated into the final representation. Our framework can model a wide range of spatial distributions of appearance patterns, and exploit different relations between them. Our method is more flexible than previous models since it can exploit the relationship between spatially adjacent and non-adjacent regions.

The image representation after spatial feature co-pooling is of high dimensionality. When the dictionary size is large, it is easy to become computationally infeasible. Therefore, feature selection is necessary to choose the most discriminative patterns. Optimizing the problem directly is difficult for a large scale visual task, e.g., object classification. We adopt grafting [8], an incremental feature selection algorithm, to achieve an approximate solution to this problem. To improve the speed of feature selection, a batch version of grafting is proposed. However, the performance of the algorithm deteriorates very fast as the batch size[1] becomes large. To further overcome this problem, we

* Corresponding author.
E-mail address: seu.liufeng@gmail.com (F. Liu).

[1] The number of features added in one iteration.

http://dx.doi.org/10.1016/j.neucom.2014.02.015
0925-2312/

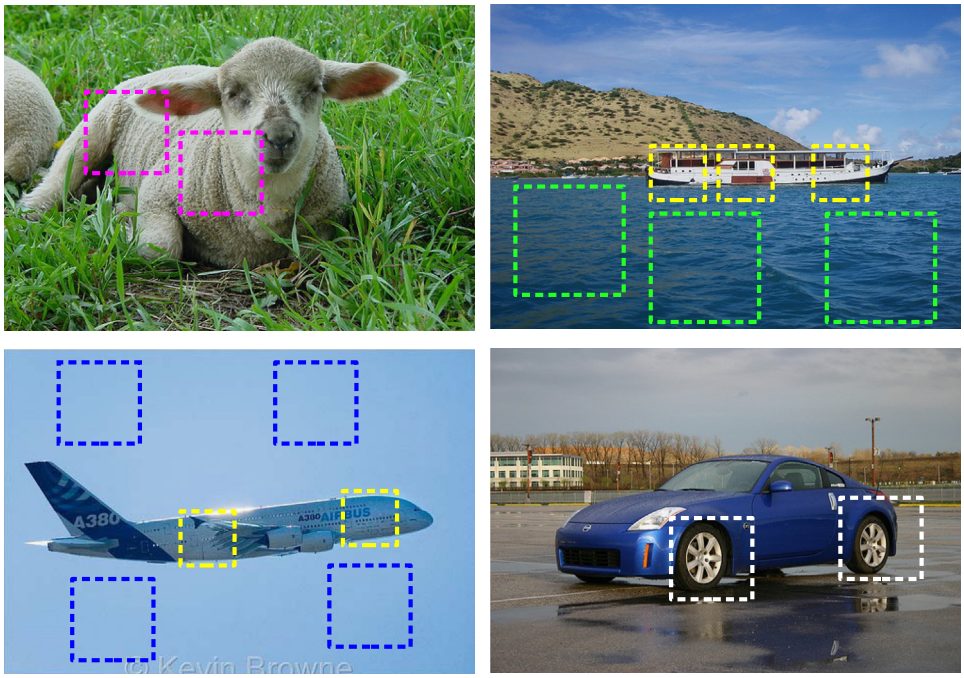


Fig. 1. Several examples of the role of spatial modeling for visual object analysis. Best viewed in color. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this paper.)

present a semi-greedy (SG) grafting algorithm. It filters features with little discriminative information, which is implemented by solving a $L_{2,1}$ regularized least square regression problem via half-quadric optimization.

The major contributions of this work are two folds: (1) We present a spatial feature co-pooling framework. It can be used to exploit a wide range of spatial structures of local appearance patterns. It can clearly describe the co-occurrence information of similar patterns at different spatial locations. (2) We provide a theoretical analysis to the original grafting algorithm, and propose a modified version of grafting named SG grafting for discriminative feature selection, which improves grafting with an embedded optimization algorithm.

The rest of the paper is organized as follows. In Section 2, we describe our algorithm platform and propose a spatial feature co-pooling scheme. In Section 3, we provide a theoretical analysis to the grafting algorithm from the viewpoint of the proximal gradient method [9,10], and present our improved grafting algorithm for the feature selection problem. Experimental results are reported in Section 4. At last, we summarize the paper in Section 5.

2. Modeling patterns' distributions

The spatial distribution of meaningful appearance patterns is an important cue for image analysis, as illustrated in Fig. 1. On the one hand, the spatial distribution reflects context and material information with spatial constraints, e.g., describing a sheep picture as 'grass around and wool in the middle', or a boat picture as 'sky above, water below and wood in the middle'. On the other hand, it contains some class-specific structures such as the distribution of non-adjacent eyes for a face and non-adjacent wheels for a car.

We choose the popular bag-of-feature (BoF) model as our algorithm platform, wherein each pattern in an image is ultimately represented by one or more visual words. Measuring the co-activation of them over different spatial regions is to mimic the visual perception of locating the objects' features. Before proposing the spatial feature co-pooling, we first briefly review the standard pooling method.

2.1. Pooling revisited

Spatial pooling is a key step in the BoF model. It integrates feature responses on each visual word into one value. Typically, average pooling or max pooling is used, which preserves the average or the maximum feature response.

As the BoF model ignores the spatial layout of features, some researchers try to model the spatial relations of local features at the pooling step. Lazebnik et al. propose a technique of spatial pyramid matching (SPM) [2], which first divides an image into cells of multiple resolutions and then concatenates the pooling results on each cell as the final representation. Based on cell division, Sharma et al. [11] use visual saliency as a weight for each cell and jointly learn the saliency scores and the large margin classifier, where the visual saliency consists both features saliency and spatial saliency. Huang et al. [12] develop a strategy called multiple spatial pooling to model the global spatial structure of objects. A feature can be pooled multiple times with different weights according to its relationship with the Gaussian distributions. Ji et al. [13] propose an unsupervised object-enhanced feature generation mechanism which highlights the features from regions of objects, which can be seen as given more weights to features of the objects during pooling. Krapac et al. [4] fit a GMM for each visual word and use a Fisher vector to represent the spatial information of a picture. Jia et al. [1] learn a best pooling region from a predefined over-completed region. Feng et al. [3] learn a class specific pooling function from features' geometric information. Yang et al. [14] use a co-occurrence matrix between visual words under particular spatial constraints to model spatial information.

Unlike their approaches, we propose to model the distribution of patterns by using the co-occurrence of appearance patterns

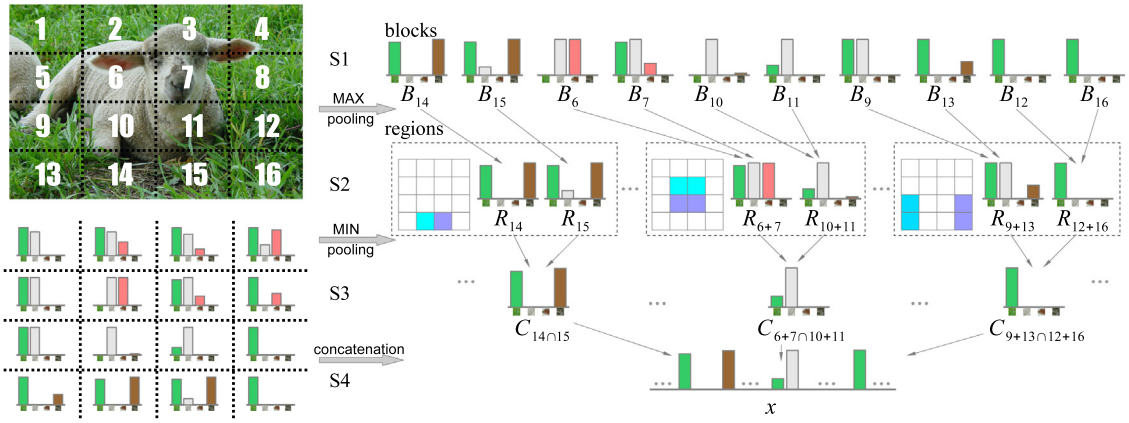


Fig. 2. A toy example of spatial feature co-pooling. The encoded features are first pooled onto the 16 blocks using max pooling ($B_1, \ldots, B_{16}$). Then adjacent blocks are pooled together to form regions (e.g., R_{14}, R_{6+7}, R_{9+13} are regions of size 1×1, 1×2, 2×1 blocks respectively). They are merged via max pooling. Two regions are further combined by min pooling which takes the minimum of the two inputs as output at each dimension (e.g., $C_{14\cap15}$ is the pooling result of the configuration of R_{14}, R_{15}). Finally the ultimate representation is the concatenation of responses over all configurations. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this paper.)

at distinctive spatial locations. Cao et al. [15] also try to model the pattern's distribution by using the segmentation result and embed it to SPM, where segmentation information serves as local information. In our approach, we exploit the global and the local layout of patterns, and no extra information is needed.

2.2. Spatial feature co-pooling

Spatial feature co-pooling is a framework which can be used to exploit a wide range of spatial distributions of appearance patterns. Its core idea is to pool features of distinctive spatial locations together and use a pooling function to describe the patterns' relationships. As shown in Fig. 2, our framework consists of the following steps. First, features from adjacent blocks are pooled together to form a region. Then regions with different locations are pooled together and concatenated to form the ultimate representation of a picture. We explain each step in detail as follows.

A picture is first divided into several regular blocks. According to their locations in an image, features falling in the same block are pooled together by a standard pooling method, e.g., max pooling and average pooling. This is a spatial location quantization process, through which a feature gains some invariance to small shifting. At the same time, it makes feature matching available between different pictures, since they are always different in size.

Then several adjacent blocks are merged together to form a region to represent patterns of varying resolutions. Fig. 2 (S2) gives us an illustration of such merging with regions containing 1×1, 1×2, 2×1 blocks. The histogram representations of the blocks of the same color are combined by a pooling operation, and we call the combined representation as the region's representation. Typically, we use max pooling or average pooling in this step.

Afterward, the co-occurrence of patterns is discovered by taking a min pooling over regions of different positions, both adjacent and non-adjacent (S3). In this paper, we just study the configuration of pairwise regions to reduce complexity. It is the core part of the proposed spatial co-pooling method. Owing to a pattern that can be well represented by the responses of one or several visual words, measuring the co-activation of such visual words of different regions can best reflect the existence of co-occurrence patterns at this area.

However, it is difficult for the standard pooling method to find out the co-activation between visual words, because what they reflected is the accumulative information. Inspired by the intersection kernel, we propose a min pooling strategy, which takes the minimum value of the input vectors at each dimension. Only similar appearance patterns of the inputs are kept because the output value at each dimension is nonzero only if both its inputs are nonzero. As shown in Fig. 2, if a pattern only appears in one region, its responses of visual words will be suppressed after the min pooling operation under this region configuration.

Since the pooled representation of one region configuration can describe the patterns' co-occurrence at this area, we can roughly model the patterns' distribution on a whole image by concatenating the pooling results of different region configurations.

The processes above together are called spatial co-pooling. We are free to choose the pooling method on each step. We can even use several pooling methods on the pairwise region merging step, what we need is just to concatenate the representations of different pooling methods together. The proposed spatial co-pooling is a flexible framework, with different pooling methods used, different pattern relations can be reflected.

2.3. Model training

The output of spatial feature co-pooling is a histogram of a fixed length. Compared with the standard BoF model, the main difference is that we replace the pooling step with spatial feature co-pooling. We train the model by optimizing the following problem for each class separately:

$$\min_{\mathbf{w}} \frac{1}{n}\sum_{i=1}^{n} \max(0, 1-y_i\mathbf{w}^T\mathbf{x}_i)^2+\lambda\mathbf{w}^T\mathbf{w} \quad (1)$$

where $\mathbf{w}$ is the learned weight, $\mathbf{x}_i$ is each picture's representation, y_i is the label of $\mathbf{x}_i$ and n is the number of training pictures. The first term is the square of hinge loss which is also called L2 SVM loss [16]. This loss function preserves the large margin property and has a continuous first-order divertive, which can be solved quickly. The second term is a regularizer to control the model's complexity. Due to the high dimensionality of $\mathbf{x}$, training such a model directly is computationally infeasible for most personal computers. So a feature selection procedure is performed before training, and we will discuss it in the next section.

3. Feature selection

Usually, a pattern has its class specific distribution, e.g., frequently appearing locations. Also, not all the patterns are discriminative for classifying a specific class. Generally, only a part of visual words' responses under particular region configurations are

useful. Moreover, training a problem of high dimensionality is challenging. Thus, it is necessary to introduce a feature selection here. In particular, we propose two strategies for feature selection: the global manner and the local manner.

The global one selects features over all classes, formulated as the following problem:

$$\min_{\mathbf{W}} \frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{c}\max(0,1-y_{ij}\mathbf{w}_j^T\mathbf{x}_i)^2+\lambda_1\|\mathbf{W}\|_F^2+\lambda_2\|\mathbf{W}\|_{2,1} \quad (2)$$

where $\mathbf{W}=(\mathbf{w}_1,\ldots,\mathbf{w}_c)$ is a $d\times c$ weight matrix, d is the number of features, c is the number of classes, n is the number of training samples, $\mathbf{x}_i$ is the ith training sample and y_{ij} is a label indicator.[2] $\|\mathbf{W}\|_{2,1}$ is defined as the sum of the L_2 norm for each row of the weight matrix, and it is used to select features useful for all classes.

The local manner selects features for each class separately. Note that it can be seen as c two-class problems, and for each class the objective can be written as the formulation above. So in the subsections below, we just take (2) as the objective of our feature selection problem.

3.1. Grafting revisited

Optimizing (2) directly is usually difficult, so we choose to use an incremental algorithm named grafting [8] to obtain an approximate solution. Grafting is a greedy algorithm which iterates over a candidate feature collection step and a model retraining step. At each iteration, it computes the gradient of the loss function with respect to the features' coefficients and selects the one with the largest magnitude. Then the model is retrained using the features selected previously along with the new selected one.

Jia et al. [1] use grafting to handle a multi-class feature selection problem by replacing the original gradient magnitude with the L_2 norm of the objective's gradient with respect to a particular row of the coefficient matrix. Note that from the viewpoint of proximal gradient [9,10], the L_1 norm can be seen as a shrinkage and threshold function of the new updated coefficients, and a proximal gradient based algorithm can be interpreted as shrinking and filtering the model's coefficient iteratively. It is similar to grafting, which selects a feature with the largest gradient magnitude. Consequently, the gradient with respect to the $L_{2,1}$ regularizer term can be removed from the gradient calculation step. So the objective can be rewritten as

$$\mathcal{L}(\mathbf{W})=\frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{c}\max(0,1-y_{ij}\mathbf{w}_j^T\mathbf{x}_i)^2+\lambda\|\mathbf{W}\|_F^2 \quad (3)$$

In order to improve the speed of feature selection, usually more than one features will be added in each iteration.

3.2. SG grafting

The algorithm above does well only when the batch size (the number of features selected in one iteration) is small. When the batch size becomes larger, the performance begins to deteriorate, which is possibly caused by

- The model coefficients in grafting are initialized as zeros. Selecting a large batch of features may lead to an inaccurate model.
- In a greedy algorithm, no feature will be discarded once selected, even though it is less discriminative. This can also lead to an inaccurate model.

To solve the above problems, we propose the semi-greedy (SG) grafting algorithm. The proposed SG algorithm is still a greedy algorithm but it will throw away some features from those selected previously and currently by adding a filter step. Nondiscriminative features will be dropped out during this process. The importance of features is determined by the magnitude of their weights, i.e., discriminative features tend to have larger weights and vice versa. This idea is inspired by the Recursive Feature Elimination (RFE) [17]. We have interpreted it from the viewpoint of proximal gradient. More specifically, we put the newly added features and the features with the smallest $\|\mathbf{W}_F^j\|_2$ to the filter step,[3] during which an embedded feature selection algorithm is performed to remove the unstable features.

The filter step preserves robust features by solving a standard $L_{2,1}$ regularized least square regression problem:

$$\min_{\mathbf{W}}\|\mathbf{Y}-\mathbf{X}\mathbf{W}\|_F^2+\lambda\|\mathbf{W}\|_{2,1} \quad (4)$$

where n, d, c in the data matrix $\mathbf{X}\in R^{n\times d}$, the weight matrix $\mathbf{W}\in R^{d\times c}$ and the label matrix $\mathbf{Y}\in R^{n\times c}$ correspond to the number of training instances, the number of features and the number of classes respectively. Since the predicted label of a *one-vs-all* SVM is determined by the maximizer of the scores $\mathbf{w}_i^T\mathbf{x}$, both *one-vs-all* SVM and the least square regression problem pursue to maximize the score of the target class and suppress others. So we can replace the original L2 SVM loss with the square loss. Our method is summarized in the following algorithm.

Algorithm 1. SG grafting.

Inputs: Data matrix $\mathbf{X}$, label matrix $\mathbf{Y}$, the number of features wanted T, the batch size b, the revaluation set size k
Outputs: The indices of selected features F and the learned weight $\mathbf{W}_F$
Initialize: Free set $F=\emptyset$, zero set $Z=\{1,\ldots,d\}$, weight matrix $\mathbf{W}=\mathbf{0}$
While $|F|<T$
1. Set active set $A=\emptyset$
2. Compute the gradient of (3) w.r.t. $\mathbf{W}_Z$, and denote as $\nabla\mathcal{L}(\mathbf{W}_F)$
3. Put b features with the largest $lVert\nabla\mathcal{L}(\mathbf{W}_F)^i\|_2$ to the active set A , $i\in Z$
4. Filter Step
   a. Set revaluation set $R=\emptyset$
   b. Sorting $\mathbf{W}_F$ and put the k features with the smallest $\|\mathbf{W}_F^i\|_2$ to the revaluation set R
   c. $A=A\bigcup R, F=F-R, Z=Z\bigcup R$
   d. Solve the optimization problem of (4) using Algorithm 2 and remove features which satisfy $\mathbf{W}_A^i=0$ from A, $i\in A$
5. $F=F\bigcup A, Z=Z-A$
6. Retrain $\mathbf{W}_F$ by solving (3) via gradient descent.

3.3. HQ optimization

By replacing the loss function with a square loss, we are able to adopt the framework proposed in [18] to solve the problem of (4) via half-quadratic (HQ) optimization [19]. As the minimizer function of $L_{2,1}$ norm is unpredictable near the origin, $\phi(x)=\sqrt{\varepsilon+x^2}$ is often used to replace the absolute value function, where ε is a smoothing term. And (4) can be rewritten as

$$\min_{\mathbf{W}}\|\mathbf{Y}-\mathbf{X}\mathbf{W}\|_F^2+\lambda\sum_{i=1}^{d}\phi(\|\mathbf{W}^i\|_2) \quad (5)$$

[2] If sample i belongs to class j, $y_{ij}=1$, otherwise $y_{ij}=-1$.

[3] The superscript denotes the jth row, and the subscript F denotes the free set. A free set is a set of already selected features. Accordingly, zero set is a set of unselected features.

According to Lemma 1 in [18], (5) can be reformulated to

$$\min_{\mathbf{W}} \|\mathbf{Y}-\mathbf{X}\mathbf{W}\|_F^2+\lambda\,\mathrm{Tr}(\mathbf{W}^T\mathbf{Q}\mathbf{W}) \tag{6}$$

where $\mathbf{Q}=\mathrm{diag}(\mathbf{q})$, $\mathbf{q}\in R^d$ is an auxiliary vector, which is uniquely determined by the minimizer function $\delta(\cdot)$. Based on HQ, (5) can be solved by alternately minimizing:

$$q_i^t=\frac{1}{\sqrt{\|\mathbf{W}^i\|_2^2+\varepsilon}} \tag{7}$$

$$\mathbf{W}^t=(\mathbf{X}^T\mathbf{X}+\lambda\mathbf{Q})\backslash\mathbf{X}^T\mathbf{Y} \tag{8}$$

where (8) is obtained by setting the derivative of the objective in (6) with respect to $\mathbf{W}$ to zero. We summarize the above algorithm in Algorithm 2.

Algorithm 2. Solving (4) via HQ optimization.

Inputs: Data matrix $\mathbf{X}\in R^{n\times d}$, label $\mathbf{Y}\in R^{n\times c}$
Outputs: weight matrix $\mathbf{W}\in R^{d\times c}$
Initialize: $\mathbf{W}\leftarrow 0$ and $t\leftarrow 1$;
Repeat
1. Compute auxiliary vector $\mathbf{q}^t$ according to (7);
2. Compute $\mathbf{W}^t$ according to (8);
Until converges

Compared with the gradient-based method, the above alternately minimizing algorithm usually takes fewer iterations for convergence. As the main computational cost in Algorithm 1 is to solve the linear problem in (8), it is usually very fast when the number of features is not very large. So it fits our framework well wherein this algorithm is only used in the filter step during which just a small set of features are involved.

4. Experiments

We first introduce the datasets, evaluation protocol and implementation details in Section 4.1, and then compare the performance of the original grafting algorithm and our proposed SG grafting. We also discuss the influence of the local and global feature selection methods. At last, we report our results on the CIFAR 10, UIUC Sports, and VOC 2007 datasets.

4.1. Experimental datasets and settings

Three datasets are used to evaluate spatial modeling algorithms. They are detailed below.

- CIFAR 10 [20] is an object classification dataset with 10 object categories. It consists of 60,000 32×32 color images. Among them, 50,000 images are for training and the remaining 10,000 are for testing.
- UIUC Sports [21] is a static event recognition dataset with 8 sport events. For each event we randomly choose 70 pictures for training and 60 pictures for testing. We repeat the experiment five times, and mean accuracy and standard deviation are reported.
- VOC 2007 [22] is a challenging dataset for object classification. It consists thousands of images of real world scenes over 20 classes. The split `train` and `val` is used for training, and `test` is used to evaluate algorithms.

We follow a standard BoF paradigm [23] to train the classification algorithms, and the settings for every step are detailed below.

Basic settings: Whitened 6×6 pixel color image patches [24] are used as features on the CIFAR 10 dataset. SIFT features [25] extracted at scales 4, 6, 8, 10 with a stride of three pixels are adopted on the other two datasets. K-means clustering algorithm is employed to generate visual dictionaries from one million randomly selected features. Triangle coding [24] is chosen to encode the local features on the CIFAR 10 dataset, while locality-constrained linear coding (LLC) [26] is used on the other two datasets.

Spatial modeling: Our work mainly focus on the spatial modeling process. The spatial pyramid matching (SPM) [2] and the receptive field learning (RFL) [1] are chosen as the baseline algorithms. For SPM, partition $1\times 1, 1\times 3, 2\times 2$ is used on the VOC 2007 dataset, and that of $1\times 1, 2\times 2, 4\times 4$ is used on other datasets. For receptive field learning (RFL) [1] we follow the same configuration as in [1]. As to the proposed spatial co-pooling (SCP) algorithm, a 4×4 partition is used to build the blocks on non-VOC datasets, and multi-resolution with a partition of 2×2 and 3×3 is exploited on the VOC 2007 dataset. Candidate regions can be of sizes $1\times 1, 1\times 2, 1\times 3, 1\times 4, 2\times 1, 2\times 2, 2\times 3, 2\times 4, 3\times 1, 3\times 2, 4\times 1, 4\times 2$ blocks. Configuration of regions is any two regions with the same size and without overlapping.

4.2. Grafting vs SG grafting

In this section we compare the original grafting algorithm with our algorithm under different batch sizes on the CIFAR 10 dataset with a dictionary size of 100. We used the proposed improved algorithm to select features for all classes simultaneously which we call a global feature selection. λ_1 and λ_2 (for SG grafting) are set to 0.52 and 0.2 respectively (via cross validation). The size of revaluation set size k is set to half of the batchsize, which gives a best tradeoff between the learning speed and the performance. After feature selection we train a standard L2 SVM using the

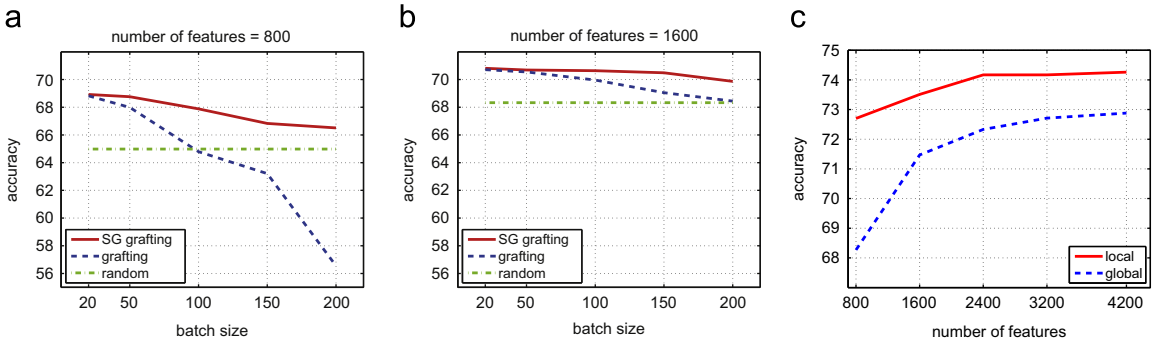


Fig. 3. Comparisons of SG grafting and grafting, the global manner and the local manner on the CIFAR 10 dataset. (a) and (b) show the improvement of our algorithm over original grafting algorithm under different batch sizes. (c) displays the performances of global and local feature selection strategies.

training set and test on the testing set. Their accuracies are reported in Fig. 3a and b respectively.

The proposed method clearly improves the original grafting on all batch sizes. The improvement becomes more apparent when the batch sizes are larger. Such an improvement benefits from our strategy of discarding features from the already selected ones and filtering the new ones.

Fig. 3a and b also shows that the speed that the performance drops is closely related with the ratio between the batch size and the number of preserved features. It is clear that the performance deteriorates much faster when this ratio is larger (and vice vera). So when the total number of features to be preserved is large, we can increase the batch size accordingly.

Note that discarding features will take more iterations for convergence. For example, if it takes a hundred iterations for the original algorithm to select enough features, the proposed method will take 5–7 iterations more. However, such filtering mainly happened in the first several iterations from our observation. The possible reason is that the gradient magnitude cannot fully reflect the true importance of a feature with all weights initialized as zero. With more features are selected, their corresponding weights help the model to make the right decision. Since the main computation cost is retraining the model, as well as the retraining process is quite fast when the number of features is small, the SG grafting algorithm takes far less time than the original one to achieve the same accuracy (we can use a larger batchsize).

4.3. Global vs local

In the literature [27], sharing features between classes may potentially improves the overall performance, since it is a way to build relations between classes. However, in our cases, we find that the spatial distribution of a particular pattern is class specific. We design an extra experiment to verify this viewpoint. The global

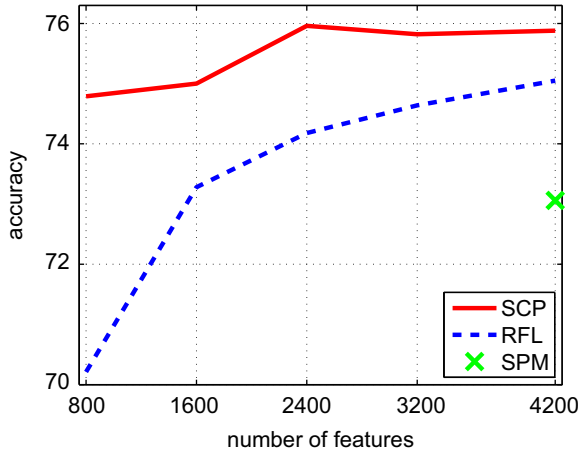


Fig. 4. Performance on the CIFAR 10 dataset.

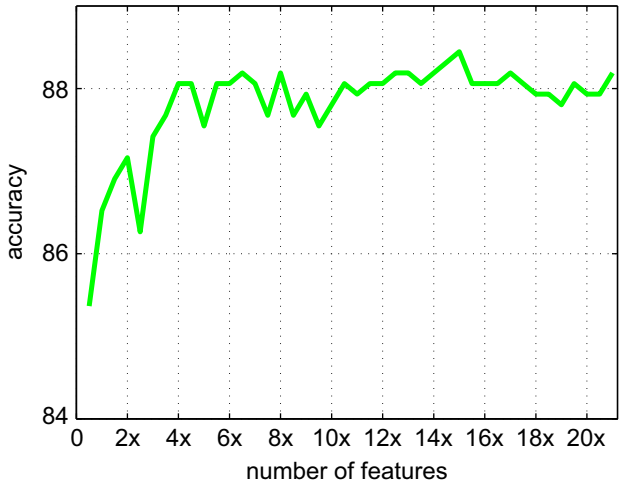


Fig. 6. Relation between the accuracy and the number of selected features.

Table 1
Performance comparison on the UIUC sports dataset.

Method	LLC	LLC+SPM(21 ×)	LLC+SCP(4 ×)[a]	LLC+SCP(8 ×)	LLC+SCP(21 ×)
Accuracy	78.33 ± 1.44	87.09 ± 1.26	89.04 ± 0.63	89.06 ± 0.77	89.11 ± 0.38

[a] '$a\times$' means the number of features is a times the dictionary size.

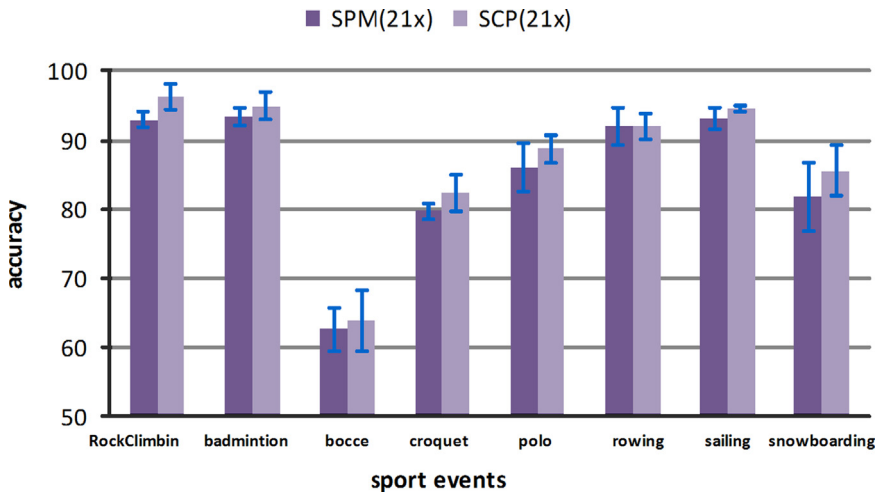


Fig. 5. Performance of each class on the UIUC sports dataset.

and the local feature selection are defined in Section 3. After feature selection, we train an L2 SVM using the selected features for each class separately. The experimental results on CIFAR 10 dataset with a dictionary size of 200 are reported in Fig. 3c.

It can be seen that the local method can achieve a much higher accuracy at a smaller number of preserved features while the global one needs more features to achieve the same accuracy. This observation is consistent with our prediction that the features' distribution is class specific. So we adopt a local feature selection method in the subsequent experiments.

4.4. Comparison on different datasets

4.4.1. Results on CIFAR 10

In this section, our method is compared with standard SPM and RFL with a dictionary size of 200. In our spatial feature co-pooling, both max pooling and min pooling are employed when merging the regions. All results are shown in Fig. 4. Our method (SCP) performs comparably with the best results of RFL with only 800 features. When more features are selected, we yield better performance than SPM and RFL. We have analyzed the selected features and find that more features are obtained by min pooling, especially when the number of features to be preserved is small, which indicates that these features are more discriminative.

4.4.2. Results on UIUC sports

In this section, we compare our method with SPM. We use LLC as our coding method at a dictionary size of 1024. The performance of SPM and our method is listed in Table 1 and Fig. 5. Our method shows superiority over SPM on all classes.

We also plot a curve to show the relationship between the accuracy and the number of features in Fig. 6. It is interesting that the performance improves fast in the beginning followed by a significant decline. As more features are added, the performance improves again. It indicates that the features selected in the beginning are widely shared by all samples in a class. The features added subsequently are shared more by only training samples, so the performance begins to drop. As more features are selected, more class-specific information is discovered, and the performance is improved again.

4.5. Results on the PASCAL VOC 2007 dataset

At last, we evaluate the proposed algorithm on the PASCAL VOC 2007 dataset. The locality-constrained linear coding at a dictionary size of 1024 is used to encode local features. Comparison of the algorithms of spatial co-pooling and spatial pyramid matching is detailed in Table 2, and the spatial co-pooling has employed a block division of 2×2 and 3×3. The total number of selected features is 21 times the number of visual words. For the SPM, we follow the common configurations on this dataset with a region partition of 1×1, 1×3, and 2×2. Other configurations are also compared and their results are reported in Fig. 7. From the table we can find that the spatial co-pooling has improved the precision almost on all classes (except person and plant), which demonstrates the importance of co-occurrence patterns for classifying objects. The decrease of performances on the person and plant

Table 2
Results on the VOC 2007 dataset.

Class	Aero	Bike	Bird	Boat	Bottle	Bus	Car	Cat	Chair	Cow
SPM	65.27	54.39	33.78	58.93	19.59	48.03	69.95	48.66	44.27	33.24
SCP(21 ×)	**68.18**	**56.50**	**41.81**	**59.71**	**21.42**	**49.50**	**70.64**	**51.18**	**44.77**	**37.73**
Class	Table	Dog	Horse	Mbike	Person	Plant	Sheep	Sofa	Train	TV
SPM	36.36	35.46	69.20	50.81	**76.85**	**19.32**	34.02	41.65	63.74	45.07
SCP(21 ×)	**42.45**	**37.70**	**72.97**	**56.65**	76.08	17.33	**37.93**	**45.03**	**69.46**	**46.10**

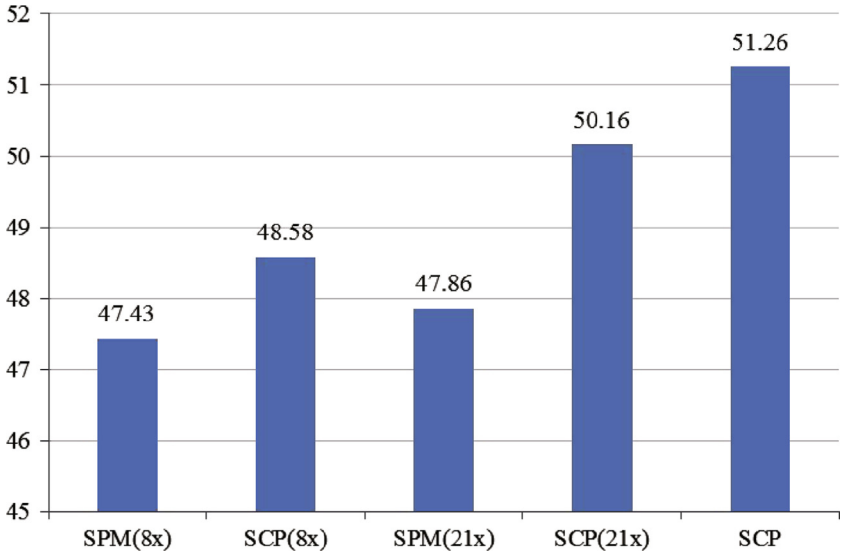


Fig. 7. Comparison of different spatial modeling algorithms on the VOC 2007 dataset. The SCP column shows the result of the spatial co-pooling algorithm without feature selection.

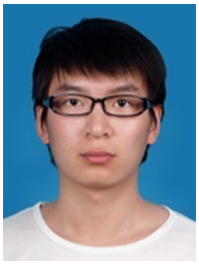
classes is caused by the overfitting of the feature selection algorithm since the strength of regularization is chosen globally. A proof for this is the average precision on these two classes are 78.46 and 21.80 before feature selection.

5. Conclusion

In this paper, we have proposed a spatial feature co-pooling framework and used a min pooling scheme to describe the spatial distributions of patterns. Unlike traditional pooling methods, min pooling can capture the co-occurrence of patterns at different spatial locations, which is an important cue demonstrated by extensive experimental results. By using blocks as the basis instead of each feature, our method gains robustness to small feature shifting and shows good performance on both aligned (CIFAR 10) and unaligned datasets (UIUC sports, VOC 2007). Future work includes designing more efficient region configurations and improving the efficiency of the proposed SG grafting algorithm.

References

[1] Y. Jia, C. Huang, T. Darrell, Beyond spatial pyramids: receptive field learning for pooled image features, in: CVPR, 2012, pp. 3370–3377.
[2] S. Lazebnik, C. Schmid, J. Ponce, Beyond bags of features: spatial pyramid matching for recognizing natural scene categories, in: CVPR, vol. 2, 2006, pp. 2169–2178.
[3] J. Feng, B. Ni, Q. Tian, S. Yan, Geometric l_p-norm feature pooling for image classification, in: CVPR, 2011, pp. 2609–2704.
[4] J. Krapac, J. Verbeek, F. Jurie, Modeling spatial layout with fisher vectors for image categorization, in: ICCV, 2011, pp. 1487–1494.
[5] S. Savarese, J. Winn, A. Criminisi, Discriminative object class models of appearance and shape by correlations, in: CVPR, vol. 2, 2006, pp. 2033–2040.
[6] F. Moosmann, D. Larlus, F. Jurie, et al., Learning saliency maps for object categorization, in: International Workshop on ECCV 2006, 2006.
[7] D. Parikh, C.L. Zitnick, T. Chen, Determining patch saliency using low-level context, in: Computer Vision-ECCV 2008, Springer, 2008, pp. 446–459.
[8] S. Perkins, K. Lacker, J. Theiler, Grafting: fast, incremental feature selection by gradient descent in function space, J. Mach. Learn. Res. 3 (2003) 1333–1356.
[9] A. Beck, M. Teboulle, A fast iterative shrinkage-thresholding algorithm for linear inverse problems, SIAM J. Imaging Sci. 2 (1) (2009) 183–202.
[10] X. Chen, W. Pan, J. Kwok, J. Carbonell, Accelerated gradient method for multi-task sparse learning problem, in: ICDM, 2009, pp. 746–751.
[11] G. Sharma, F. Jurie, C. Schmid, Discriminative spatial saliency for image classification, in: 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, 2012, pp. 3506–3513.
[12] Y. Huang, Z. Wu, L. Wang, C. Song, Multiple spatial pooling for visual object recognition, Neurocomputing 129 (2014) 225–231.
[13] Z. Ji, J. Wang, Y. Su, Z. Song, S. Xing, Balance between object and background: object enhanced features for scene image classification, Neurocomputing 120 (2013) 15–23.
[14] Y. Yang, S. Newsam, Spatial pyramid co-occurrence for image classification, in: ICCV, 2011, pp. 1465–1472.
[15] L. Cao, R. Ji, Y. Gao, Y. Yang, Q. Tian, Weakly supervised sparse coding with geometric consistency pooling, in: CVPR, 2012, pp. 3578–3585.
[16] C. Lin, R. Weng, S. Keerthi, Trust region Newton method for logistic regression, J. Mach. Learn. Res. 9 (2008) 627–650.
[17] I. Guyon, J. Weston, S. Barnhill, V. Vapnik, Gene selection for cancer classification using support vector machines, Mach. Learn. 46 (1) (2002) 389–422.
[18] R. He, T. Tan, L. Wang, W. Zheng, $l_{2,1}$ regularized correntropy for robust feature selection, in: CVPR, 2012, pp. 2504–2511.
[19] M. Nikolova, M. Ng, Analysis of half-quadratic minimization methods for signal and image recovery, SIAM J. Sci. Comput. 27 (3) (2005) 937–966.
[20] A. Krizhevsky, G. Hinton, Learning multiple layers of features from tiny images (Master's thesis), Department of Computer Science, University of Toronto.
[21] L. Li, L. Fei-Fei, What, where and who? classifying events by scene and object recognition, in: ICCV, 2007, pp. 1–8.
[22] M. Everingham, L. Van Gool, C.K.I. Williams, J. Winn, A. Zisserman, The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results, ⟨http://www.pascal-network.org/challenges/VOC/voc2007/workshop/index.html⟩.
[23] K. Chatfield, V. Lempitsky, A. Vedaldi, A. Zisserman, The devil is in the details: an evaluation of recent feature encoding methods, in: BMVC, 2011.
[24] A. Coates, A. Ng, The importance of encoding versus training with sparse coding and vector quantization, in: ICML, vol. 8, 2011, p. 10.
[25] D. Lowe, Distinctive image features from scale-invariant keypoints, Int. J. Comput. Vision 60 (2) (2004) 91–110.
[26] J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, Y. Gong, Locality-constrained linear coding for image classification, in: CVPR, 2010, pp. 3360–3367.
[27] J. Liu, S. Ji, J. Ye, Multi-task feature learning via efficient l2, 1-norm minimization, in: The Twenty-Fifth Conference on Uncertainty in Artificial Intelligence, 2009, pp. 339–348.

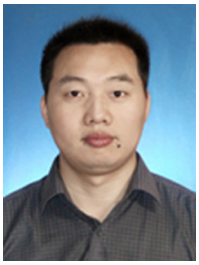


Feng Liu received the B.S. degree in the China University of Mining and Technology, Xuzhou, China, in 2010. From 2011 to 2013, he was with the School of Automation, Southeast University as a Master student. His research interests include computer vision and machine learning.

Yongzhen Huang received the B.E. degree from Huazhong University of Science and Technology in 2006 and the Ph.D. degree from Institute of Automation, Chinese Academy of Sciences (CASIA) in 2011. In July 2011, he joined the National Laboratory of Pattern Recognition (NLPR), CASIA, where he is currently an Associate Professor. He has published more than 30 papers in the areas of computer vision and pattern recognition at international journals and conferences such as IEEE TPAMI, TSMC-B, CVPR, NIPS, ICIP and ICPR. His current research interests include pattern recognition, computer vision, machine learning and biologically inspired vision computing. He is a member of IEEE.

Liang Wang received both the B.Eng. and M.Eng. degrees from Anhui University in 1997 and 2000 respectively, and the Ph.D. degree from the Institute of Automation, Chinese Academy of Sciences (CASIA) in 2004. From 2004 to 2010, he has been working as a Research Assistant at Imperial College London, United Kingdom and Monash University, Australia, a Research Fellow at the University of Melbourne, Australia, and a Lecturer at the University of Bath, United Kingdom. Currently, he is a full Professor of Hundred Talents Program at the National Lab. of Pattern Recognition, CASIA. His major research interests include machine learning, pattern recognition and computer vision. He has widely published at highly ranked international journals such as IEEE TPAMI and IEEE TIP, and leading international conferences such as CVPR, ICCV and ICDM. He is an Associate Editor of IEEE Transactions on SMC-B, International Journal of Image and Graphics, Signal Processing, Neurocomputing and International Journal of Cognitive Biometrics. He is currently a Senior Member of IEEE.

Wankou Yang received the B.S., M.S. and Ph.D. degrees in the School of Computer Science and Technology, Nanjing University of Science and Technology (NUST), China, respectively in 2002, 2004, and 2009. From July 2009 to August 2011, he worked as a Postdoctoral Fellow in the School of Automation, Southeast University, China. Since September 2011, he has been an Assistant Professor in the School of Automation, Southeast University. His research interests include pattern recognition, computer vision and machine learning.



Changyin Sun is a Professor in the School of Automation at the Southeast University, China. He received the M.S. and Ph.D. degrees in Electrical Engineering from the Southeast University, Nanjing, China, respectively, in 2001 and 2003. His research interests include Intelligent Control, Neural Networks, SVM, Pattern Recognition, and Optimal Theory. He has received the First Prize of Nature Science of Ministry of Education, China. He has published more than 40 papers. He is a Member of an IEEE, an Associate Editor of IEEE Transactions on Neural Networks, Neural Processing Letters and International Journal of Swarm Intelligence Research, Recent Patents on Computer Science.