Fast Kernel Learning for Spatial Pyramid Matching

Junfeng He, Shih-Fu Chang
Department of Electrical Engineering
Columbia University, New York, NY 10027
jh2700,sfchang@columbia.edu

Lexing Xie
IBM T.J. Watson Research Center
Hawthorne, NY 10532
xlx@us.ibm.com

Abstract

Spatial pyramid matching (SPM) is a simple yet effective approach to compute similarity between images. Similarity kernels at different regions and scales are usually fused by some heuristic weights. In this paper,we develop a novel and fast approach to improve SPM by finding the optimal kernel fusing weights from multiple scales, locations, as well as codebooks. One unique contribution of our approach is the novel formulation of kernel matrix learning problem leading to an efficient quadratic programming solution, with much lower complexity than those associated with existing solutions (e.g., semidefinite programming). We demonstrate performance gains of the proposed methods by evaluations over well-known public data sets such as natural scenes and TRECVID 2007.

1. Introduction

In the image classification field, much recent work has followed the framework of "bag of words", representing an image as a collection of local features [5, 4, 9, 1, 2]. Among them, spatial pyramid matching (SPM) has shown impressive results [2] by incorporating the spatial layout information of local features. SPM repeatedly subdivides the images to finer resolution and computes the histograms of local features at each grid point of each resolution, then measures similarity of two images by computing the histogram intersections. The final similarity of two images is obtained via summing all grid-level similarities with predefined weights.

The fusion weights set in SPM are usually diadic ($w_l = \frac{1}{2^{L-l+1}}$, where L is the total number of levels and l is the level index), or constant ($w_l = 1$). The former emphasizes matching results at higher resolutions while the latter treats contributions from each level uniformly. However, neither approach explores variation of weights over different spatial locations. In this paper, we hypothesize spatially adaptive weighting is important in discovering key features that characterize image difference between distinct classes. Take natural scene recognition task as an example. If the task is to classify "coast" from "open country", then the upper regions of images would be less useful, because in both classes upper regions are often about sky or cloud. But the bottom regions in images of "coast" are often about water, while the bottom regions in images of "open country" are often about grass, road, etc. So in this case, the bottom regions are more discriminative and hence should have higher weights. In this paper, we focus on efficient methods for automatically discovering the most discriminative weights for supervised learning tasks such as image classification. With our method, SPM is improved by finding the optimal kernel that fuses inputs from multiple scales, locations, and codebooks.

The main contributions of our paper are

1. We proposed the idea of discriminative SPM (DSPM) within a kernel matrix learning framework;
2. We developed a fast and effective approach to solve the kernel matrix learning problem based on quadratic programming;
3. The proposed method can be readily applied to find optimal fusion of inputs from a large variety of kernels, including spatial regions, resolutions, as well as codebooks constructed from different visual features.

Our paper is organized as follows: in section 2, we introduce the background and some related work; our main algorithms are discussed in section 3; experiments and analysis on several public data sets are shown in section 4; finally, discussions on extending our method to SPM with multiple codebooks are provided in section 5.

2. Related work

Local features (such as SIFT [20]) are becoming popular in recent image classification systems [5, 4, 11, 9, 1, 2, 6, 8, 7, 3]. One typical approach [5, 1, 2, 3] is to construct image similarity kernels by using local features, and then use kernel based classification methods such as SVM

for recognition. Grauman and Darrell proposed pyramid matching kernel in [1]. It measures the similarity of two images by computing weighted sum of feature matches, i.e., intersection of features fallen into the same bin in the feature space. Pyramid matching kernel has one disadvantage: it discards all spatial information. To overcome this problem, the SPM approach is proposed in [2]. It also performs pyramid matching, but in the two-dimensional image space instead of the feature space. Features are clustered at first to construct a codebook of codewords. SPM computes the grid similarity of two images by counting codeword matches, i.e., intersection of codewords fallen into the same grid in the two-dimensional image space. Grid similarities are then fused with predefined weights to obtain the image similarity. In [3], different fusion weights other than the heuristic ones in SPM are discussed based on a cross-validation strategy. However, in [3], weights are not spatially adaptive, i.e., their weights are still set to be uniform over different spatial locations and thus do not capture contributions of unique local features to specific classes (e.g., sky regions in the top of the "Open Country" class). Moreover, there is no efficient procedure for determining the optimal weights. Only a naive cross-validation approach is proposed, which can only deal with few parameters and would increase the computation complexity a lot.

The main technical problem addressed in this paper is to answer what is the optimum convex combination of predefined kernels when a set of labelled training data is available. Our work is inspired by previous work in kernel matrix learning field [12, 13, 14], which performs semi-definite programming (SDP) to align the combined kernel to the ideal kernel, i.e., the label similarity kernel. However these methods have the limitation of high computation complexity [**?**]. Other related work includes distance (metric) learning [7, 8], and semi-supervised kernel matrix learning [15, 16, 17, 18], which incorporates information from unlabelled data.

3. Fast kernel learning for spatial pyramid matching

3.1. Spatial pyramid matching

In SPM, we first extract local features, such as SIFT features for images, then we quantize all feature vectors into M types, each of which is called a code word in the codebook. It is assumed that features of the same code word can be perceived equivalent to one another. Spatial Pyramid matching works in L levels of image resolutions. In level 0, there is only one grid for the whole image, in level 1, the image is partitioned to 4 grids of the same size, and in level l, the image is partitioned to $(2^l)^2$ grids of the same size, etc. For two images I_1 and I_2, spatial pyramid matching kernel $K$is defined as:

$$K(I_1, I_2) = \sum_{l=1}^{L} \sum_{i=1}^{G_l} w_{l,i} K_{l,i}(I_1, I_2) \tag{1}$$

$$K_{l,i}(I_1, I_2) = \sum_{m=1}^{M} \min(H_{l,i}^m(I_1), H_{l,i}^m(I_2)) \tag{2}$$

Here, $w_{l,i}$ is the weight for the i-th grid in the l level. In [2], it is chosen as:

$$l > 0, w_{l,i} = \frac{1}{2^{L-l+1}}; l = 0, w_{l,i} = \frac{1}{2^L}. \tag{3}$$

L is the total number of levels and G_l is the total number of grids in level l. $H_{l,i}^m(I_1)$ is the number of code word m appearing in i-th grid of l-th level in image I_1. In practice, it is reported that $L = 2$ or $L = 3$ is enough [10].

3.2. Fast kernel learning for spatial pyramid matching

For unsupervised tasks such as image retrieval, weights in equation (3) seem reasonable. However, for supervised learning tasks such as image classification, besides the feature information, we also know the label information for each picture. Instead of using spatially uniform weights in equation (3), we would like to find the optimum weights $w_{l,i}$ that are most discriminative in separating images of distinct classes.

With a proof similar to that in [1], it is easy to see that each region-level similarity $K_{l,i}$ in SPM is a Mercel kernel matrix. So finding optimal weight in SPM is actually equivalent to the problem of fusing predefined kernels according to the label information, which is a standard kernel matrix learning problem. We proposed a new fast and effective method to solve it.

For clarity and simplicity, we rewrite equation (1) as:

$$K = \sum_{j=1}^{J} u_j K_j \tag{4}$$

where K_j is the similarity kernel in one grid of one resolution level and $J = 1 + ... + (2^L)^2$. K is used to represent the similarity among data. In the field of kernel matrix learning (or multiple kernel learning), it is believed that the target similarity matrix should be close to the label similarity. Therefore, we would like to find u_j, such that K is close to label similarity Y. Usually there are two ways to define label similarity:

$$Y_{i,j} = \delta(y_i, y_j) \tag{5}$$

where $\delta(y_i, y_j) = 1$ if $y_i = y_j$; $\delta(y_i, y_j) = 0$, otherwise. Or

$$Y = yy^T \tag{6}$$

where y is the vector consisting of all the labels. Depending on the label convention, the actual label value may be different. If we set y_i to be $\{+1, -1\}$, then the label similarity matrix in equation (6) may have negative value elements. In SPM, all elements in K are non-negative, so we choose the non-negative Y defined in equation (5).

Previous kernel matrix learning approaches try to maximize $cos(K,Y) = \frac{<K,Y>_F}{<K,K>_F<Y,Y>_F}$, leading to a semi-definite programming (SDP) problem [12, 14] with a very high computation complexity. In the following, we present a fast method for learning the optimal kernel matrix.

We use the following criteria to minimize the distance between K and Y, which would lead us to a quadratic programming problem as shown later :

$$||K - Y||_F^2 \tag{7}$$

Here $||X||_F^2 = tr(XX^T)$, where tr means $trace$ operation. More intuitively, $||K-Y||_F^2$ is the sum of element-wise distance of K and Y, i.e., $||K - Y||_F^2 = \sum_i \sum_j (K_{i,j} - Y_{i,j})^2$.

We will encounter a scale problem if we use the above formulation directly. Because the elements in K are not limited within $[0, 1]$, minimizing $||K - Y||_F^2$ may not guarantee a good result [1]. For example, suppose image i and j are from the same class, i.e., $Y_{i,j} = 1$. If $K_{i,j} > 1$, the higher $K_{i,j}$ is, the more penalty is given in equation (7). But in fact this case should be encouraged, because the two images are from the same class and their features are similar.

In order to solve the scaling problem, we constrain all the elements in K within $[0, 1]$ as: [2]

$$\begin{aligned}
&\min_{u,K} ||K - Y||_F^2 \\
&s.t. \\
&K = \sum_{j=1}^{J} u_j \overline{K_j}, \\
&u_j \geq 0, j = 1, ..., J, \\
&\sum_{j=1}^{J} u_j = 1,
\end{aligned} \tag{8}$$

Here $\overline{K_j}$ is a normalized variant of K_j, namely K_j divided by the largest absolute value in K_j. Since each element in $\overline{K_j}$ is between [0,1] and $\sum_{j=1}^{J} u_j = 1$, all the elements in K are also within $[0, 1]$. Additionally, as pointed out in [1], positive scaling or positive linear combination of kernel matrices are still kernel matrices. $\overline{K_j}$ and K are always kernel matrices, since they are positive scaling or positive linear combination of kernel matrices $K_j, j = 1, ..., J$.

To prevent overfitting in our learning procedure, a regularization term $||u||^2$ can be added to equation (8), i.e.,

$$\min_{u,K} ||K - Y||_F^2 + \lambda ||u||^2 \tag{9}$$

instead of $\min_{u,K} ||K - Y||_F^2$. Actually, the regularization term prefers more uniform solutions, but note that in the original SPM the weights are (spatially) uniform. So we specifically add this term to make sure we do not over-depend on the learned weights in the case of too few (or unbalanced) training data and unreliable learned weights. Thus, conceptually the regularization term is used to explore the tradeoff between trusting the learned optimal weights and favoring the uniform weights like those from the original SPM. λ is a tradeoff parameter chosen by users. In our experiments, the parameter λ is chosen within $[0.05tr(K_0 K_0^T), 0.5tr(K_0 K_0^T)]$, where K_0 is the kernel matrix of SPM. In most cases, we choose λ as $0.1tr(K_0 K_0^T)$.

Note that

$$\begin{aligned}
&||K - Y||_F^2 \\
&= || \sum_{j=1}^{J} u_j \overline{K_j} - Y||_F^2 \\
&= tr((\sum_{j=1}^{J} u_j \overline{K_j})^T (\sum_{j=1}^{J} u_j \overline{K_j}) - 2Y^T \sum_{j=1}^{J} u_j \overline{K_j} + Y^T Y) \\
&= u^T A u - 2b^T u + c
\end{aligned}$$

where

$$\begin{aligned}
A_{i,j} &= tr(\overline{K_i}^T \overline{K_j}), \\
b_j &= tr(Y^T \overline{K_j}) \\
c &= tr(Y^T Y)
\end{aligned}$$

[1] A criteria similar as in equation (7) was proposed in [19]. But the scaling problem is not addressed. Moreover, unlike in our paper, their approach was not proposed for optimal combination of predefined kernels.

[2] If we do not constrain all the elements in K within $[0, 1]$, an alternative approach is to use hinge loss, which will lead to a linear programming problem as follows:

$$\begin{aligned}
&\min_{u,K} \sum_{p,q} \varepsilon_{p,q} \\
&s.t. \\
&K = \sum_{j=1}^{J} u_j K_j, \\
&K_{p,q} \geq Y_{p,q} - \varepsilon_{p,q}, if Y_{p,q} = 1 \\
&K_{p,q} \leq Y_{p,q} + \varepsilon_{p,q}, if Y_{p,q} \neq 1 \\
&\varepsilon_{p,q} \geq 0, \\
&u_j \geq 0, j = 1, ..., J,
\end{aligned}$$

However there are N^2 constraints in this linear programming problem (N is the number of the data), making it much slower than the one based on quadratic programming. So this paper focuses on the quadratic programming based approach.

So optimal solution u can be obtained as following:

$$\begin{aligned} &\min_{u} u^T(A+\lambda I)u - 2b^T u \\ &s.t., \\ &u_j \geq 0, j = 1, ..., J \\ &\sum_{j=1}^{J} u_j = 1 \end{aligned} \tag{10}$$

Here, I is the identical matrix with the same size of A.

The total number of the unknowns in our optimization problem is J. Recall that $J = 1 + ... + (2^L)^2$. Usually in SPM, L is chosen as 2, hence J is 21. So the above optimization only needs to solve a quadratic programming problem with 21 unknowns, which is quite fast, efficient, and easy to implement.

4. Experiments

In this section, we report experiment results using two datasets: natural scene and TRECVID 2007.

4.1. Natural scene data set

Our first dataset is the natural scene data set of thirteen scene categories provided by Fei-Fei and Perona in [9]. There are 200 to 400 images with average image size of 300×250 pixels in each category. Some example images are shown in figure 1. Here, we consider several binary "one vs. one" classification problems, e.g., "Open Country vs. Coast", "Open Country vs. Forest", etc., so that we can discover the most discriminative regions and features between two classes, and show them in an intuitive way. In this experiment, 50 images for each class is used as training data, and the rest as test data. SIFT descriptors are computed from each 16 x 16 overlapping pixel patch with uniform spacing of 8 pixels. Then K-means clustering is used to form the visual codebook over a randomly selected subset of patches.

Example weights of each region obtained by our fast kernel learning approach is shown in the first row of figure 2. As a comparison, weights obtained by the SDP methods in [13] with the same experiment setup are shown in the second row of figure 2. Note $L = 2$ is used and thus the highest resolution of grids is 4×4 . The results are very encouraging – the discovered weights by our approach are more informative and intuitive. They indicate the specific regions and features that capture the most salient difference between two image classes. For example, the first image in the first row of figure 2 shows the most important feature distinguishing "Open Country" and "Coast" are in the lower part of the images (corresponding to grass field or beach). Likewise, the most important regions distinguishing "Open Country vs. Forest" are in the upper area of an image, as shown in the second image in the first row of figure 2. Classification accuracy of SVM using SPM kernels with different weights are shown in table 1. Not surprisingly, the more discriminative weights obtained by our method do improve the classification performance.

4.2. Trecvid 2007 data set

We also apply SPM and our method on a more difficult problem: concept detection on TRECVID 2007 data set [21]. In TRECVID 2007 data set, each image is labelled with one or several labels from a list of 36 concepts, such as "Indoor/Outdoor", "People", "Speech", "Mountain" and so on. Some example pictures are shown in figure 3. For a classification task, TRECVID 2007 data set is much more difficult than natural scene or Caltech 101 data set. First of all, it is a multi-labelled problem. Secondly the images are from different sources with various quality. Moreover, unlike in Caltech 101 data set, objects in TRECVID images are often small, sometimes incomplete. Finally, some concepts in TRECVID 2007 represent some high-level knowledge, such as "people marching", "police", or "studio", and hence are quite hard for classification. Nontheless, it would be interesting to test the performance of our method on this challenging problem.

17520 images are used to form the training set, 4012 images are used as the test set. Because the number of images and the number of clusters are both huge, K-Means is too slow for clustering in this case. Instead, we apply a fast clustering method similar to that discussed in [24] for codebook construction. We construct a codebook with about 5000 codewords based on SIFT features. "One vs all" strategy is performed: for each concept, 100 positive images, i.e. those labelled with the concept, 100 negative images, i.e. those not labelled with the concept, are randomly selected as the training set. We apply SVM as the classifier after obtaining the similarity kernel. Inferred average precision (IAP) [22], the standard evaluation measure in TRECVID 2007 data set [23], is used to evaluate the performance. The above experiment is repeated 5 times and mean and standard deviation of the accuracy are computed.

Experiment results are shown in figure 4. The classification accuracy is improved by the proposed method on most concepts, especially on those related to spatial layout such as "sky" (by 10% relatively), "road" (by 15% relatively), and "building" (by 26% relatively). The overall mean average precision (MAP) across all concepts is increased by around 10% relatively. Surprisingly, performance is also improved significantly on concepts like "crowd" (by 30% relatively) "TV-Screen" (by 50% relatively), "Car" (by 20% relatively), etc., which do not have consistent spatial layout. One possibility is that those concepts benefit from the context spatial information. For example, "Car" is usually on the "road", and "road" is usually located at the bottom part of the images.

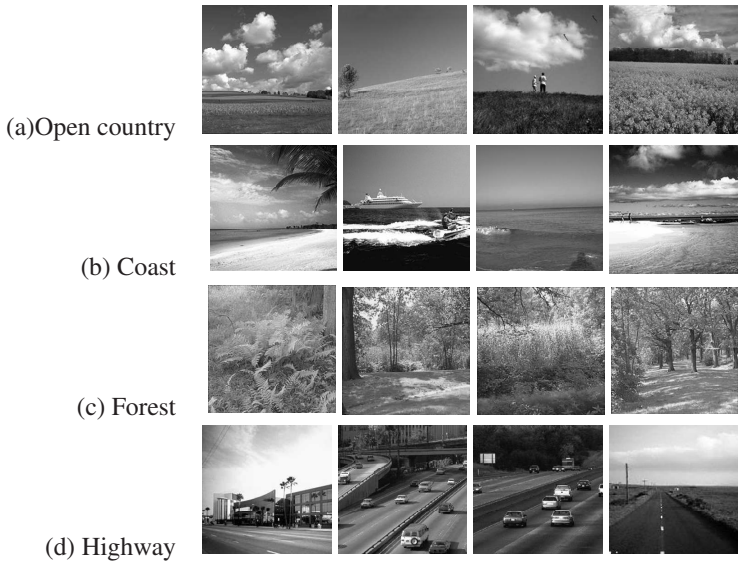
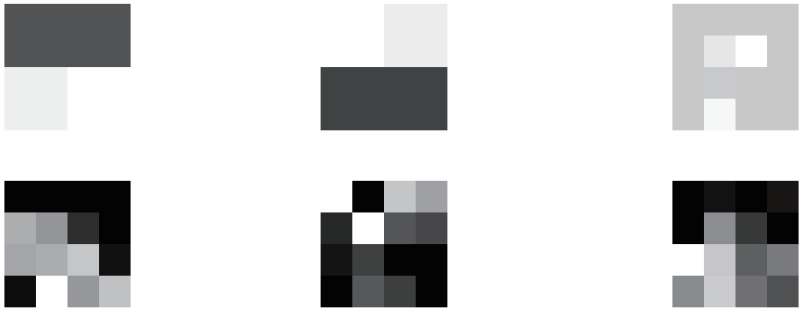


Figure 1. Examples of pictures in the natural scene dataset



(a)Open Country vs. Coast (b)Open Country vs. Forest (c) Open Country vs. Highway

Figure 2. Comparison of optimal weights obtained by our method and by SDP based method in [13] for some one vs. one classification on natural scene data set. Images of the first row show the weights computed with our method, while images of the second row show the weights obtained by the kernel learning method in [13]. The brighter the region is, the higher weight it has and the more influence it has on distinguishing the image classes. Our method is able to reveal discriminative regions, such as the upper part of the images for distinguishing "Open Country" and "Forest", while the method in [13] provides more random weights.

5. Discussion on multiple codebooks

Up to now, SPM is based on single codebook, usually constructed with SIFT features. Though SIFT has impressive advantages such as robust to illumination, viewpoint change, etc., for some regions which is not geometrically salient, other features may be more effective. For example, for regions about "sky", color might be the most important feature, and for regions about "grass", texture would be more useful. Hence, instead of using one code book, we can build spatial pyramid matching on multiple codebooks, for example, one code book created with SIFT features, one with color features, and one with texture features,etc. Then we can extend the SPM kernel fusion model in equation (1) to combine kernels computed based on different code books:

$$K(I_1, I_2) = \sum_{l=1}^{L} \sum_{i=1}^{G_l} \sum_{d=1}^{D} w_{l,i,d} K_{l,i,d}(I_1, I_2) \qquad (11)$$

$$K_{l,i.d}(I_1, I_2) = \sum_{m=1}^{M} \min(H^m_{l,i,d}(I_1), H^m_{l,i,d}(I_2)) \qquad (12)$$

where $H^m_{l,i,d}(I_1)$ is the total number of code word m in i-th grid of l-th level in image I_1 when using d-th codebook.

Our proposed optimization approach can be readily applied to handle this case to obtain the optimal weights. We

	Open Country vs. Coast	Open Country vs. Forest	Open Country vs. Highway
Heuristic weights in SPM [2]	0.811 ± 0.008	0.875 ± 0.002	0.802 ± 0.018
Weights by the method in [13]	0.804 ± 0.008	0.874 ± 0.004	0.815 ± 0.018
Weights by our method	0.824 ± 0.005	0.881 ± 0.005	0.822 ± 0.010

Table 1. Classification accuracy of SVM using SPM kernels with different weights. The experiment is repeated 5 times, and average accuracy and standard deviation is reported.

Figure 3. Example images from TRECVID 2007 data set. Most images in this data set are multi-labelled. For example, the first image of the first row is labelled as: "office, face, person, TV screen";the first image of the second row is labelled as: "outdoor, building, vegetation, road, sky"; the first image of the third row is labelled as: "face, person, crowd, police, military".

would like to further investigate whether multiple codebooks would be helpful for SPM in the future work.

6. Conclusion

A fast kernel matrix learning approach for spatial pyramid matching has been developed in this paper. In image classification tasks, it improves SPM by revealing the most discriminative scales and locations. We only need to solve a quadratic programming problem with few unknowns, leading to a very fast solution.

However, similar as in other kernel matrix learning methods, problems may occur when the training data is heavily unbalanced. For example, if positive examples dominate the training data, then the optimal kernel fusion process in our approach may be dominated by the positive class, resulting in inaccurate and non-discriminative weights. In this case, reweighing the loss on positive and negative data might be helpful. Another possible improvement is the extension of our method to multi-labelled data. Especially we will study methods to take advantage of the concept correlation in multi-labelled data.

Acknowledgments

This work has been supported by the U.S. Government. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the U.S. Government.

References

[1] K. Grauman and T. Darrell. The Pyramid Match Kernel: Discriminantive Classification with Sets of Image Features. *Proceedings of ICCV*, 2005. 1, 2, 3

[2] S. Lazebnik, C. Schmid and J. Ponce. Beyond Bags of Features, Spatial Pyramid Matching for Recognizing Natural Scene Categories. Proceedings of CVPR, 2006. 1, 2, 6

[3] A. Bosch, Andrew Zisserman and X. Munoz. Representing shape with a spatial pyramid kernel. Proceedings of CIVR, 2007. 1, 2

[4] J. Sivic and A. Zisserman. Video Google: A Text Retrieval Approach to Object Matching in Videos. Proceedings of ICCV, 2003. 1

[5] C. Wallraven, B. Caputo, and A. Graf. Recognition with local features: the kernel recipe. Proceedings of ICCV, 2003. 1

[6] J. Mutch and D. G. Lowe. Multiclass object recognition with sparse, localized features. Proceedings of CVPR, 2006. 1

[7] Andrea Frome, Yoram Singer, Fei Sha, Jitendra Malik. Learning Globally-Consistent Local Distance Functions for Shape-Based Image Retrieval and Classification. Proceedings of ICCV, 2007. 1, 2

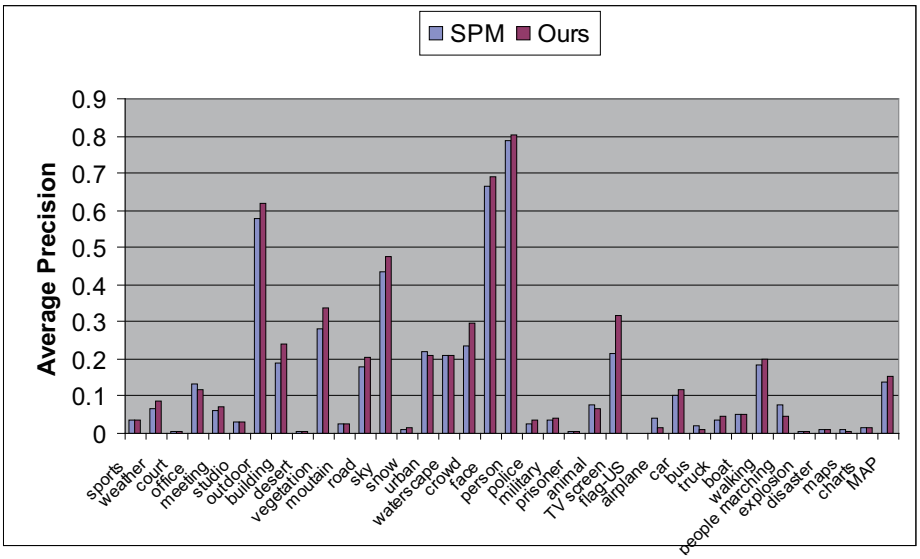


Figure 4. Comparison of performance obtained by baseline SPM and our Discriminative SPM (DSPM) with the proposed kernel learning method. The accuracy is measured in Inferred average precision (IAP) [22]. The experiment is repeated 5 times, and average accuracies of two methods on each class are reported.

[8] Andrea Frome, Yoram Singer, Jitendra Malik. Image Retrieval and Recognition Using Local Distance Functions. Proceedings of NIPS, 2006. 1, 2

[9] L. Fei-Fei and P. Perona. A Bayesian hierarchical model for learning natural scene categories. Proceedings of CVPR, 2005. 1, 4

[10] A. Bosch, A. Zisserman. Scene classification via plsa. Proceedings of ECCV, 2006. 2

[11] 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. CVPR Workshop of Generative Model Based Vision, 2004. 1

[12] N. Cristianini, J. Taylor, A. Elisseeff, and J. Kandola. On kernel-target alignment. Proceedings of NIPS, 2002. 2, 3

[13] F. R. Bach, G. R. G. Lanckriet, and M. I. Jordan. Multiple kernel learning, conic duality, and the SMO algorithm. Proceedings of ICML, 2004. 2, 4, 5, 6

[14] G. Lanckriet, N. Cristianini, P. Bartlett, L. E. Ghaoui, and M. Jordan. Learning the kernel matrix with semidefinite programming. Journal of Machine Learning Research, 2004. 2, 3

[15] Zhu, X., Ghahramani, Z., and Lafferty, J. Semisupervised learning using gaussian fields and harmonic functions. Proceedings of ICML, 2003. 2

[16] X. Zhu, J. Kandola, Z. Ghahramani, and J. Lafferty. Nonparametric transforms of graph kernels for semi-supervised learning. Proceedings of NIPS, 2005. 2

[17] Hoi, S. C. H., Lyu, M. R., and Chang, E. Y. Learning the unified kernel machines for classification. Proceedings of KDD, 2006. 2

[18] Steven C.H. Hoi, Rong Jin and Michael R. Lyu. Learning Non-Parametric Kernel Matrices from Pairwise Constraints. Proceedings of ICML, 2007. 2

[19] F. De la Torre Frade and O. Vinyals. Learning Kernel Expansions for Image Classification. Proceedings of CVPR, 2007. 3

[20] D. Lowe. Object Recognition from Local Scale-Invariant Features. Proceedings of ICCV, 1999.. 1

[21] TRECVID. http://www-nlpir.nist.gov/projects/tv2007/tv2007.html. 4

[22] Emine Yilmaz and Javed A. Aslam. Estimating Average Precision with Incomplete and Imperfect Judgments. Proceedings of the Fifteenth ACM International Conference on Information and Knowledge Management (CIKM), 2006. 4, 7

[23] Inferred average precision website. http://www-nlpir.nist.gov/projects/tv2006/infAP.html. 4

[24] Frank Moosmann and Bill Triggs and Frederic Jurie. Randomized clustering forests for building fast and discriminative visual vocabularies. Proceedings of NIPS, 2006. 4