Open access • Book Chapter • DOI:10.1007/978-3-642-19315-6_24

A multi-scale learning framework for visual categorization — Source link

Shao-Chuan Wang, Yu-Chiang Frank Wang

Institutions: Center for Information Technology

Published on: 08 Nov 2010 - Asian Conference on Computer Vision

Topics: Pyramid (image processing), Multiple kernel learning, Feature (computer vision), Visual Word and Contextual image classification

A Multi-Scale Learning Framework for Visual Categorization

Shao-Chuan Wang and Yu-Chiang Frank Wang

Research Center for Information Technology Innovation
Academia Sinica, Taipei, Taiwan

Abstract. Spatial pyramid matching has recently become a promising technique for image classification. Despite its success and popularity, no prior work has tackled the problem of learning the optimal spatial pyramid representation for the given image data and the associated object category. We propose a Multiple Scale Learning (MSL) framework to learn the best weights for each scale in the pyramid. Our MSL algorithm would produce class-specific spatial pyramid image representations and thus provide improved recognition performance. We approach the MSL problem as solving a multiple kernel learning (MKL) task, which defines the optimal combination of base kernels constructed at different pyramid levels. A wide range of experiments on Oxford flower and Caltech-101 datasets are conducted, including the use of state-of-the-art feature encoding and pooling strategies. Finally, excellent empirical results reported on both datasets validate the feasibility of our proposed method.

1 Introduction

Among existing methods for image classification, the bag-of-features model [1,2] has become a very popular technique and has demonstrated its success in recent years. It quantizes image descriptors into distinct visual words, and uses a compact histogram representation to record the numbers of occurrences of each visual word in an image. One of the major problems of this is the determination of visual words, since the widely-used strategy is to cluster local image descriptors into a set of disjoint groups, and thus the representative of each group is considered as a visual word of the given image data [1,2] (the collection of such visual words is called a *dictionary* (or a *codebook*)). However, the major concern of this technique is that it discards the spatial order of local descriptors. Lazebnik et al. [3] proposed a spatial pyramid matching (SPM) technique to address this concern by utilizing a spatial pyramid image representation, in which an image is iteratively divided into grid cells in a top-down way (i.e. from coarse to fine scales). Instead of constructing a codebook by vector quantization, Yang et al. [4] further extended the spatial pyramid representation and proposed a ScSPM framework with sparse coding of image descriptors and max pooling techniques. Since only linear kernels are required in their work, Yang's ScSPM is able to address large-scale classification problems with reasonable computation time.

R. Kimmel, R. Klette, and A. Sugimoto (Eds.): ACCV 2010, Part I, LNCS 6492, pp. 310–322, 2011.

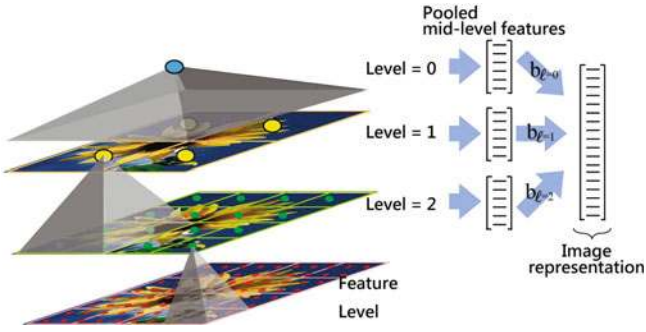

Fig. 1. An illustration of spatial pyramid image representation. Red dots are the encoded coefficient vectors of local image descriptors. Gray pyramids represent the *pooling* operations. Blue, yellow, and green dots are the pooled vectors at levels 0, 1 and 2, respectively. Each dot describes the patch statistics within the associated grid region. These pooled vectors are typically concatenated with predetermined weights (i.e. b_ℓ are fixed) as a single vector, which is the final spatial pyramid image representation. Given the image data and the associated object category, our multi-scale learning (MSL) framework aims at identifying the optimal weights for improved classification.

To the best of our knowledge, no prior work has addressed the determination of the *best* spatial pyramid representation of the given image data and the associated object category. Existing methods using SPM only focus on the designs of feature encoding methods, pooling strategies and the corresponding classifiers, and all prior work uses predetermined weights to concatenate mid-level representation in each scale (c.f. Fig. 1). It is not surprising that, for visual categorization, some object images are more discriminative at coarse levels, while others contain more descriptive information at finer scales. Therefore, we advocate the *learning* of the best spatial pyramid representation by approaching this problem as solving a multiple kernel learning (MKL) task, and we refer to our proposed method as a Multiple Scale Learning (MSL) framework. More specifically, given the image data and the associated object category, our MSL determines the optimal combination of base kernels constructed at different pyramid levels, and we will show that this task can be posed as a convex optimization problem and guarantees the global optimum. We will also show that the learned weights for each image scale provide descriptive and semantic interpretation of image data, and our proposed spatial pyramid representation significantly improves the recognition performance of image classification.

2 Related Work

Our work is built upon the recent development of spatial pyramid representation [3, 5] and kernel learning techniques [6, 7] for image classification. As shown in Figure 1, Lazebnik et al. [3] suggested to partition an image into $2^\ell \times 2^\ell$ grids in different scales $\ell = 0, 1, 2$, etc. The histogram of visual words (or equivalently

the vectors pooled by the sum operation) within each grid is calculated. All histograms from different grids and levels are concatenated with a predetermined factor (e.g. 1 or $1/2^{2\ell}$). The final concatenated vector is thus considered as the spatial pyramid representation of the given image. We note that if the coarsest level $\ell = 0$ is used, SPM is simply the standard bag-of-features model.

To the best of our knowledge, existing approaches using SPM for image classification simply integrate visual word histograms generated at different levels of the pyramid in an *ad hoc* way, which might not be practical. We thus propose to construct the *optimal* spatial pyramid representation for each class by *learning* the weighting factors at each level in the pyramid. Our goal is not only to achieve better recognition performance, but provides an effective visualization of sematic and scale information for each object class. While it is possible to use cross validation to determine the optimal weights for each visual word histogram at different levels in the pyramid, it will significantly increases the computation complexity, especially if there is a large number of free parameters to be determined in the entire system. We note that existing work has utilized different learning or optimization strategies to address this type of problem, and the performance can be improved without sacrificing the computation load. More specifically, researchers in machine learning communities have proposed boosting techniques to select the optimal kernel or feature combination for recognition, regression, etc. problems [8, 9, 10, 11, 12]. Other methods like metric/similarity learning [13, 14], distance function learning [15, 16, 17, 18], and descriptor learning [19] also apply the latest optimization strategies to adaptively learn the parameters from the data. Recently, one of the successful examples in image classification and kernel learning is the fusion of heterogeneous features proposed by Gehler and Nowozin [10], and also by Bosch et al. [12]. Gehler and Nowozin proposed to combine heterogeneous features via multiple kernel learning as well as linear programming boosting methods (LPBoost), while Bosch fused shape and appearance features via MKL combined with a regions of interest preprocessing. Both reported attractive results on Caltech datasets.

Inspired by the above work, we propose to use a MKL framework to identify discriminating image scales, and thus weight the image representations accordingly. We will show in Sect. 4 that the performance of our proposed framework outperforms state-of-the-art methods using bag-of-features or SPM models using predetermined weighting schemes. It is worth repeating that, once the optimal weights for each scale are determined, one can easily extract significant scale information for each image object class. This provides an effective semantic interpretation for the given image data.

3 Multi-Scale Learning for Image Classification

Previously, Gehler et al. [10] associated image features with kernel functions, and transformed the feature selection/combination problem into a task of kernel selection. Similarly, Subrahmanya and Shin [20] performed a feature selection procedure by constructing base kernels using different group of features. Our proposed

MSL framework incorporates multi-scale spatial and appearance information to learn the optimal spatial pyramid representation for image classification.

In our MSL, we define a multi-scale kernel matrix, which is positive semi-definite and satisfies

$$K_{ij} \equiv K(\boldsymbol{x}_i, \boldsymbol{x}_j) = \sum_{\ell=0}^{L} b_\ell k^\ell(\boldsymbol{v}_i^\ell, \boldsymbol{v}_j^\ell), \tag{1}$$

where $\boldsymbol{x}_i$ is the image representation of i-th image, k^ℓ is the kernel function constructed at level ℓ in the spatial pyramid, b_ℓ is the associated weight, and $\boldsymbol{v}^\ell \in \mathbf{R}^{(2^{2\ell})\mathcal{K}}$ is the vector produced by concatenating all $2^{2\ell}$ pooled vectors at level ℓ. We note that if the base kernel is linear (as we did in this paper), b_ℓ will be super-linearly proportional to the number of grids in level ℓ, since the kernel output is the inner product between the two pooled vectors from each level.

The determination of the optimal weights in the above equation is known as the multiple kernel learning problem. Several algorithms have been proposed to solve the MKL problem and its variants. The reviews of MKL from an optimization viewpoint can be seen in [6, 7, 21, 22, 23], and we particularly employ the algorithm proposed by Sonnenburg et al. [23] due to its efficiency and simplicity of implementation.

In order to learn the optimal kernels over image scales to represent an image, we convert the original MKL problem into the following optimization problem (in its primal form),

$$\begin{aligned} (P) \min_{\boldsymbol{w}_\ell, w_0, \boldsymbol{\xi}, \boldsymbol{b}} \quad & \tfrac{1}{2}(\textstyle\sum_{\ell=0}^{L} b_\ell \langle \boldsymbol{w}_\ell, \boldsymbol{w}_\ell \rangle)^2 + C \textstyle\sum_{i=1}^{N} \xi_i \\ \text{subject to} \quad & y_i(\textstyle\sum_{\ell=0}^{L} b_\ell \langle \boldsymbol{w}_\ell, \Phi(\boldsymbol{v}_i^\ell) \rangle + w_0) \geq 1 - \xi_i \\ & \textstyle\sum_{\ell=0}^{L} b_\ell = 1, \boldsymbol{b} \succeq 0, \boldsymbol{\xi} \succeq 0, \end{aligned} \tag{2}$$

where $\langle , \rangle$ represents the inner product in the L_2 Hilbert space, $b = (b_0, b_1, \ldots, b_L)^T$, and $\boldsymbol{\xi} = (\xi_1, \xi_2, \ldots, \xi_N)^T$. However, similar to the standard SVM optimization problem, the above optimization problem is not as explicit as its dual problem, which is shown as follows,

$$\begin{aligned} (D) \min_{\mathbf{a}, \gamma} \quad & \gamma - \textstyle\sum_i^N a_i \\ \text{subject to} \quad & 0 \preceq \boldsymbol{a} \preceq C, \textstyle\sum_i^N a_i y_i = 0 \\ & \tfrac{1}{2} \textstyle\sum_{ij}^N a_i a_j y_i y_j k_{ij}^\ell \leq \gamma, \forall \ell = 0, 1, ..., L, \end{aligned} \tag{3}$$

where $k_{ij}^\ell = k^\ell(\boldsymbol{v}_i^\ell, \boldsymbol{v}_j^\ell) = \langle \Phi(\boldsymbol{v}_i^\ell), \Phi(\boldsymbol{v}_j^\ell) \rangle$, and $\boldsymbol{a} = (a_1, a_2, \ldots, a_N)^T$. If the kernel is linear (as ours in this paper), $\langle \Phi(\boldsymbol{v}_i^\ell), \Phi(\boldsymbol{v}_j^\ell) \rangle$ is simply $\langle \boldsymbol{v}_i^\ell, \boldsymbol{v}_j^\ell \rangle$. Note that we have one quadratic constraint for each kernel k^ℓ, i.e., we have $L + 1$ constraints in total. Sonnenburg et al. [23] have shown that the above problem can be reformulated as a semi-infinite linear program (SILP),

$$\max_{\boldsymbol{b},\theta} \quad \theta \tag{4}$$
$$\text{subject to} \quad \sum_{\ell=0}^{L} b_\ell = 1, \boldsymbol{b} \succeq 0, \sum_{\ell=0}^{L} b_\ell S_\ell(\boldsymbol{a}) \geq \theta$$
$$\forall \boldsymbol{a} \in \mathbf{R}^N \text{ with } 0 \preceq \boldsymbol{a} \preceq C \text{ and } \sum_i y_i a_i = 0,$$

where $S_\ell(\boldsymbol{a}) \equiv \frac{1}{2}\sum_{ij}^{N} a_i a_j y_i y_j k_{ij}^{\ell} - \sum_i^N a_i$.

Note that the above SILP is actually a *linear* programming problem due to the fact that θ and $\boldsymbol{b}$ are linearly constrained with *infinite* constraints, i.e. there will be a constraint for each $\boldsymbol{a} \in \mathbf{R}^N$ satisfying $0 \preceq \boldsymbol{a} \preceq C$ and $\sum_i y_i a_i = 0$. To solve this problem, a wrapper algorithm [23] is proposed to alternatively optimize $\boldsymbol{a}$ and $\boldsymbol{b}$ in each iteration. When $\boldsymbol{b}$ is fixed, SILP turns into a single kernel SVM problem, which can be efficiently solved by many SVM solvers such as LibSVM [24]. On the other hand, when $\boldsymbol{a}$ is fixed, we need to solve a linear programming problem with finite constraints, which can be also efficiently solved by many linear programming solvers. As a result, this wrapper algorithm enjoys the benefit of easy and efficient implementation.

Algorithm 1. Multi-scale learning for class-specific spatial pyramid representation

{1}. Building the kernels in all scales:
for $\ell = 0$ to L **do**
  for all i, j **do**
    $k_{ij}^{\ell} \leftarrow \langle \boldsymbol{v}_i^{\ell}, \boldsymbol{v}_j^{\ell} \rangle$ {for linear kernel}
  end for
end for
$\boldsymbol{k}^{\ell} \leftarrow \boldsymbol{k}^{\ell} / \mathbf{Tr}(\boldsymbol{k}^{\ell})$ {trace normalization}
{2}. Learning b_ℓ by solving a semi-infinite linear program [23]:
$(\boldsymbol{a}, \boldsymbol{b_\ell}) \leftarrow \text{SILP}(\boldsymbol{y}, \boldsymbol{k})$

Note that all kernel matrices have been normalized to unit trace in order to balance the contributions of base kernels. Algorithm 1 shows our proposed algorithm for learning class-specific spatial pyramid representations. Note that $\boldsymbol{a}, \boldsymbol{b_\ell}$ in Algorithm 1 are the MKL parameters; $\boldsymbol{a}$ represent the Lagrange multipliers for SVM, and $\boldsymbol{b_\ell}$ describe the optimal weights of each base kernel, indicating the preferable spatial pyramid image representation for each object category. In our implementation of the SILP solver, we integrate LibSVM [24] and the MATLAB function `linprog` for solving the single kernel SVM and the linear programming problems, respectively.

After solving the above optimization problem for the given image data, we obtain the estimated optimal weighting factors b_ℓ and equivalently acquire the significance of concatenated pooled vectors from different scales. This weighted and concatenated feature vector will be the final form for our spatial pyramid image representation.

Fig. 2. Example images from the Oxford flower dataset [25]

Fig. 3. Example images from the Caltech 101 dataset [26]

4 Experiments

4.1 Datasets

We conduct experiments on Oxford flower [25] and Caltech-101 [26] datasets in this paper. The Oxford flower dataset is a small-scale dataset containing 17 different types of flowers (80 images each). Fig. 2 shows some example images from this dataset. We randomly pick 40 images per category for training, and the remaining 40 for testing. To evaluate the feasibility and scalability of our proposed method, we further consider the Caltech-101 dataset. This dataset consists of 101 different object classes with variant numbers of images per object category (see Fig. 3 for example images). To compare our results to those reported in prior work, we use the same experimental setups such as the selection of training and test sets (15 to 30 training images per object category, and up to 50 images per category for testing), and the choice of the evaluation metric (i.e. the mean average precision (MAP)).

In our experiments on both datasets, SIFT descriptors are extracted from 16 $\times$ 16 pixel patches of an image, and the spacing between adjacent patches is 6 pixels (horizontally and vertically). We further resize the longer side of the image to 300 pixels if its width or height exceeds 300 pixels. Prior work on the Caltech-101 dataset also did similar operations [2, 4].

4.2 Dictionary Learning and Local Descriptor Encoding

We choose two dictionary learning scenarios for comparisons: vector quantization (VQ) and sparse coding (SC). We select $\mathcal{K} = 225$ and 900 as the sizes of the dictionary. To perform sparse coding, we use the SPAMS software package developed by Mairal et al. [27], and the parameter λ, which controls the sparsity of the encoded coefficient vectors $\boldsymbol{\alpha}$, is 0.2. We note that only training images are involved during the phase of dictionary learning.

4.3 Training

In our experiment, we adopt the one-vs-rest scheme to design multi-class MSL classifiers. Each classifier recognizes one class against all others, and thus learns the optimal weights b_ℓ of different image scales for the corresponding object category. Fig. 5 shows a visualization example of the learned b_ℓ, as well as the predetermined ones used in prior work. We consider only linear kernels for a major advantage that the computation complexity for training and testing will be significantly reduced compared to the cases using nonlinear kernels. Therefore, our proposed method is scalable to large-scale classification problems. The only free parameter to be determined is the regularization term C, and we apply a five-fold cross validation to search for its optimal value.

4.4 Results of the Oxford Flower Dataset

To compare our proposed MSL method with existing methods for image classification, we consider two different bag-of-features models as the baselines: the standard one without pyramid representation (i.e. level $L = 0$), and the SPM which concatenates pooled vectors from each scale with constant weights. Sum

Table 1. Mean average precision (MAP) comparison table for Oxford flower dataset. L: the maximal level in the spatial pyramid.

Encoding method		L	Pooling method	MSL	MAP $\mathcal{K}$=225	MAP $\mathcal{K}$=900
VQ	(a)	0	Sum Pooling	No	36.76%	40.00%
	(b)	2	Pyramid Sum Pooling	No	48.09%	49.26%
	(c)	2	Pyramid Sum Pooling	Yes	53.68%	55.74%
	(d)	0	Max Pooling	No	19.12%	40.59%
	(e)	2	Pyramid Max Pooling	No	55.29%	55.59%
	(f)	2	Pyramid Max Pooling	Yes	53.82%	57.35%
SC	(g)	0	Sum Pooling	No	42.94%	47.50%
	(h)	2	Pyramid Sum Pooling	No	50.74%	55.00%
	(i)	2	Pyramid Sum Pooling	Yes	55.00%	58.68%
	(j)	0	Max Pooling	No	40.74%	53.38%
	(k)	2	Pyramid Max Pooling	No	60.30%	62.79%
	(m)	2	Pyramid Max Pooling	Yes	60.15%	65.29%

and max pooling operations are used in each baseline method for the completeness of the comparison. The number of levels in the spatial pyramid is chosen as 3 (i.e. $\ell = 0$, 1 and 2) for the experiments on this dataset. The complete results and comparisons on the Oxford flower dataset are shown in Table 1.

As can been seen in Table 1, the MAP for all cases increases when the size of the dictionary (i.e. the number of visual words $\mathcal{K}$) grows. When using VQ to learn the dictionary, the cases using the max pooling strategy obtained better MAP values than those using the sum pooling one, except for the case of the standard bag-of-features model (MAP = 36.76% in Table 1(a) vs. 19.12% in (d)). We note that when the sum pooling method is applied to construct feature vectors for classification (i.e. Table 1(a) to (c)), the use of SPM improves the recognition performance, while our approach outperforms the one using predetermined weights (53.68% vs. 48.09% when $\mathcal{K} = 225$, and 55.74% vs. 49.26% when $\mathcal{K} = 900$). The max pooling strategy is also observed the same advantage of applying SPM for classification (see Table 1(d) to (f)), while both SPM methods obtained comparable MAP values.

When the dictionary is learned by SC (Table 1(g) to (m)), we observe significant improvements in MAP for all cases. Our method with max pooling strategy resulted in the *highest* MAP = 65.29% when a larger size of dictionary $\mathcal{K} = 900$ was used. Comparing to the standard SPM with constant weights, we obtained comparable MAP when $\mathcal{K} = 225$. We expect to see a significant improvement in MAP when a larger-scale classification problem is of concern (our test results on Caltech-101 support this).

4.5 Results of the Caltech-101 Dataset

Our test results on the Caltech-101 dataset and the comparison with different dictionary learning and feature pooling strategies are shown in Fig. 4. We note that the number of levels in the spatial pyramid is 4 (i.e. $\ell = 0, 1, 2, 3$). We now summarize our findings as follows:

a. The use of sparse coding for dictionary learning outperforms that learned by vector quantization. More specifically, the dictionary learned by sparse coding together with different feature pooling techniques consistently improves the recognition performance than those learned by vector quantization. This confirms the observation in [4].

b. Spatial pyramid representation significantly improves recognition accuracy. From Fig. 4, we see that the use of spatial pyramid representation with either sparse coding or vector quantization technique outperforms the pooled features from a single image scale. This is consistent with the findings in [3] and [4].

c. A larger size of the dictionary improves the performance when a single level of image representation is used. However, it produces negligible improvements when spatial pyramid representation is considered.

d. Our muti-scale learning (MSL) improves MAP with different feature encoding (sparse coding and vector quantization) and pooling strategies. In particular, our proposed framework together with sparse coding and the pyramid max pooling (PMP) strategy achieved the best MAP among all methods.

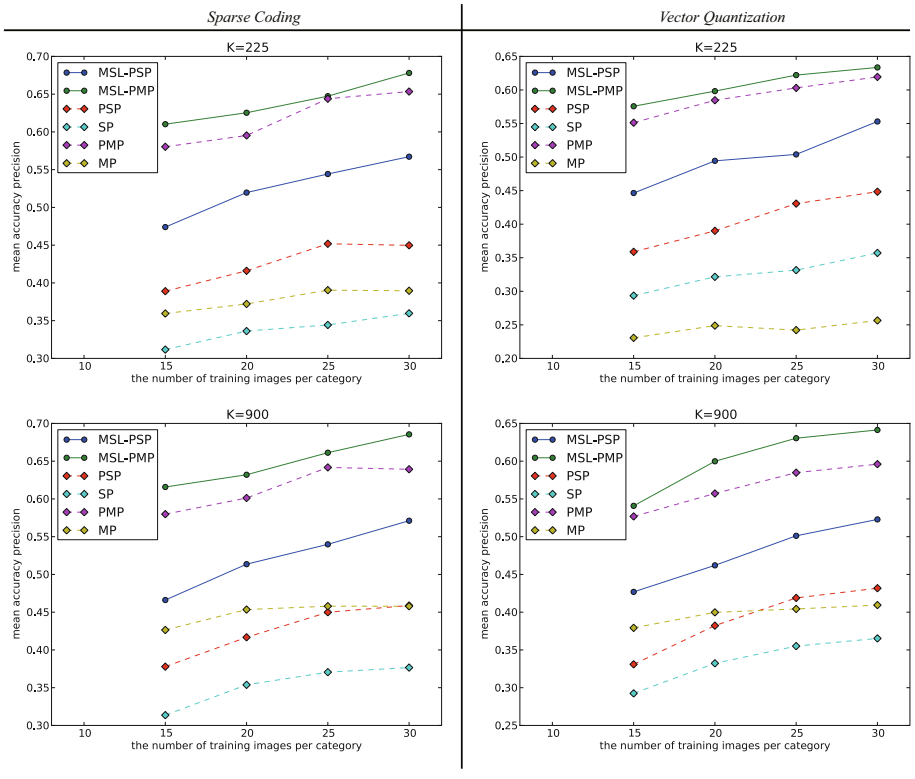

Fig. 4. Mean average precision comparison table for the Caltech-101 dataset. $\mathcal{K}$: the size of dictionary. MSL: Our Multiple Scale Learning (L=3). PSP: Pyramid Sum Pooling (L=3). SP: Sum Pooling (L=0). PMP: Pyramid Max Pooling (L=3). MP: Max Pooling (L=0). Best viewed in color.

We note that the method `PMP` in the left column of Fig. 4 is our implementation of ScSPM, which is proposed by Yang et al. [4]. We did not reproduce exactly the same results as reported in [4], probably due to the SIFT descriptor extraction, feature normalization process, etc. engineering details. Therefore, we choose to use the same implementation details for all methods on both datasets for comparisons.

To show the competitiveness of our methods, we also compare our approach with prior work on the Caltech 101 dataset. Our method outperforms many previously proposed methods, and we believe that this is because our approach is able to extract more salient properties of class-specific visual patterns across different image scales from different object categories. It is worth repeating that, different from many previous methods, our MSL framework only requires linear kernels and thus provides excellent scalability to large-scale image classification problems.

Fig. 5 illustrates the values of b_ℓ for all 101 object classes using different methods. The top row is the case of the standard bag-of-features model. Since

Table 2. Comparison with prior work on the Caltech-101 dataset. The number of training images per class for all methods is 15.

Method	MAP
Raina et al. [28]	46.6%
Berg et al. [29]	48%
Mutch and Lowe [30]	51%
Lazebnik et al. [3]	56.40%
Zhang et al. [31]	59.10%
Frome and Singer [16]	60.30 %
Lin et al. [32]	61.25 %
Our method	**61.43** %

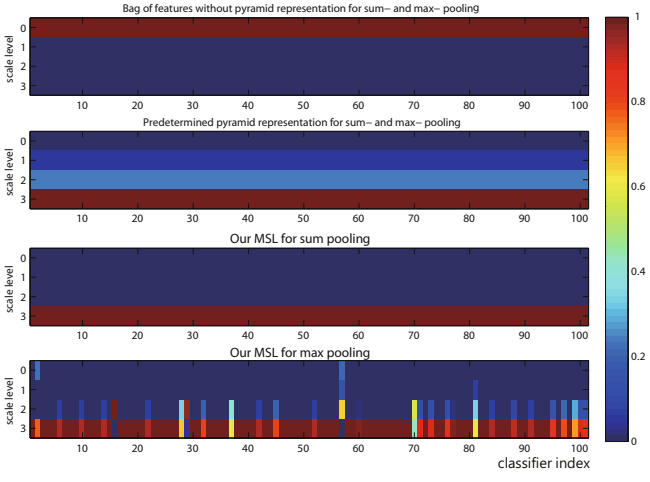

Fig. 5. Visualization of b_ℓ for different methods on the Caltech-101 dataset. The encoding method considered is vector quantization with $\mathcal{K} = 900$. Best viewed in color.

no pyramid information is used, we simply have $b_0 = 1$, and $b_\ell = 0$ otherwise. As for the SPM framework adopted by Yang et al. [4], in which the pooled vectors from each grid at each level are simply concatenated as the final image representation. It can been seen that finer scales in images are generally assigned larger (and fixed) weights due to the increasing number of grids in those pyramid levels. Finally, the last two rows in Fig. 5 present the b_ℓ learned by our method using pyramid sum and max pooling strategies, respectively. Together with the recognition performance reported, this visualization of our b_ℓ confirms that we are able to learn the optimal spatial pyramid representation given the image data, and our method can capture class-dependent salient properties of visual patterns in different image scales.

We would like to point out that we are aware of recent work which proposed to combine multiple types of descriptors or features for classification, and thus very

promising results were reported [10,11,12,33]. Our MSL framework can be easily combined with these ideas, since multiple feature descriptors can be integrated into our proposed framework and can still be solved by MKL techniques. In such cases, we expect a significantly greater improvement on the recognition accuracy over state-of-the-art classification methods.

5 Conclusion

We presented a novel MSL framework that automatically learns the optimal spatial pyramid image representation for visual categorization, which is done by solving a MKL problem which determines the optimal combination of base kernels constructed by features pooled from different image scales. Our proposed method is able to capture class-specific salient properties of visual patterns in different image scales, and thus improves the recognition performance. Among different dictionary learning and pooling strategies, our proposed framework based on sparse coding and pyramid max pooling strategies outperforms prior methods on Oxford flower and Caltech 101-datasets. In addition, through the visualization of the weights learned for each image scale and for each object category, our MSL framework produces a class-specific spatial pyramid image representation, which cannot be achieved by the standard SPM. Finally, since only linear kernels are required in our proposed learning framework, our method is computationally feasible for large-scale image classification problems.

Acknowledgement. We are grateful for the anonymous reviewers for their helpful comments. This work is supported in part by the National Science Council of Taiwan under NSC98-2218-E-001-004 and NSC99-2631-H-001-018.

References

1. Csurka, G., Bray, C., Dance, C., Fan, L.: Visual categorization with bags of keypoints. In: ECCV Workshop on Statistical Learning in Computer Vision, pp. 1–22 (2004)
2. Jurie, F., Triggs, B.: Creating efficient codebooks for visual recognition. In: ICCV 2005: Proceedings of the Tenth IEEE International Conference on Computer Vision (ICCV 2005), Washington, DC, USA, vol. 1, pp. 604–610. IEEE Computer Society, Los Alamitos (2005)
3. Lazebnik, S., Schmid, C., Ponce, J.: Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In: CVPR 2006: Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Washington, DC, USA, pp. 2169–2178. IEEE Computer Society, Los Alamitos (2006)
4. Yang, J., Yu, K., Gong, Y., Huang, T.S.: Linear spatial pyramid matching using sparse coding for image classification. In: CVPR 2009: Proceedings of the 2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 1794–1801. IEEE Computer Society, Los Alamitos (2009)

5. Grauman, K., Darrell, T.: The pyramid match kernel: discriminative classification with sets of image features. In: ICCV 2005: Proceedings of the Tenth IEEE International Conference on Computer Vision (ICCV 2005), vol. 2, pp. 1458–1465. IEEE Computer Society, Los Alamitos (2005)
6. Lanckriet, G.R.G., Cristianini, N., Bartlett, P., Ghaoui, L.E., Jordan, M.I.: Learning the kernel matrix with semidefinite programming. J. Mach. Learn. Res. 5, 27–72 (2004)
7. Bach, F.R., Lanckriet, G.R.G., Jordan, M.I.: Multiple kernel learning, conic duality, and the SMO algorithm. In: ICML 2004: Proceedings of the Twenty-First International Conference on Machine Learning, p. 6. ACM, New York (2004)
8. Crammer, K., Keshet, J., Singer, Y.: Kernel design using boosting. In: Advances in Neural Information Processing Systems 15, pp. 537–544. MIT Press, Cambridge (2003)
9. Hertz, T., Hillel, A.B., Weinshall, D.: Learning a kernel function for classification with small training samples. In: ICML 2006: Proceedings of the 23rd International Conference on Machine Learning, pp. 401–408. ACM, New York (2006)
10. Gehler, P.V., Nowozin, S.: On feature combination for multiclass object classification. In: ICCV 2009: Proceedings of the Tenth IEEE International Conference on Computer Vision (ICCV 2009). IEEE Computer Society, Los Alamitos (2009)
11. Varma, M., Ray, D.: Learning the discriminative power-invariance trade-off. In: Proceedings of the IEEE International Conference on Computer Vision. Rio de Janeiro, Brazil (2007)
12. Bosch, A., Zisserman, A., Munoz, X.: Image classification using ROIs and multiple kernel learning. In: IJCV 2008 (2008) (submitted)
13. Chopra, S., Hadsell, R., LeCun, Y.: Learning a similarity metric discriminatively, with application to face verification. In: CVPR 2005: Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. IEEE Computer Society, Los Alamitos (2005)
14. Babenko, B., Branson, S., Belongie, S.: Similarity metrics for categorization: from monolithic to category specific. In: ICCV 2009: Proceedings of the Tenth IEEE International Conference on Computer Vision (ICCV 2009), Kyoto, Japan. IEEE Computer Society, Los Alamitos (2009)
15. Hertz, T., Bar-Hillel, A., Weinshall, D.: Learning distance functions for image retrieval. In: CVPR 2004: Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 2 (2004)
16. Frome, A., Singer, Y., Malik, J.: Image retrieval and classification using local distance functions. In: Schölkopf, B., Platt, J., Hoffman, T. (eds.) Advances in Neural Information Processing Systems 19, pp. 417–424. MIT Press, Cambridge (2007)
17. Weinberger, K.Q., Saul, L.K.: Distance metric learning for large margin nearest neighbor classification. J. Mach. Learn. Res. 10, 207–244 (2009)
18. Yang, L., Jin, R., Sukthankar, R., Liu, Y.: An efficient algorithm for local distance metric learning. In: AAAI 2006: Proceedings of the 21st National Conference on Artificial Intelligence, pp. 543–548. AAAI Press, Menlo Park (2006)
19. Winder, S., Brown, M.: Learning local image descriptors. In: CVPR 2007: Proceedings of the 2007 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 1–8. IEEE Computer Society, Los Alamitos (2007)
20. Subrahmanya, N., Shin, Y.C.: Sparse multiple kernel learning for signal processing applications. IEEE Transactions on Pattern Analysis and Machine Intelligence 99 (2009)

21. Bach, F.R., Thibaux, R., Jordan, M.I.: Computing regularization paths for learning multiple kernels. In: Saul, L.K., Weiss, Y., Bottou, L. (eds.) Advances in Neural Information Processing Systems 17, pp. 73–80. MIT Press, Cambridge (2005)
22. Rakotomamonjy, A., Bach, F., Canu, S., Grandvalet, Y.: More efficiency in multiple kernel learning. In: ICML 2007: Proceedings of the 24th International Conference on Machine Learning, pp. 775–782. ACM, New York (2007)
23. Sonnenburg, S., Rätsch, G., Schäfer, C., Schölkopf, B.: Large scale multiple kernel learning. J. Mach. Learn. Res. 7, 1531–1565 (2006)
24. Chang, C.C., Lin, C.J.: LIBSVM: a library for support vector machines (2001), `http://www.csie.ntu.edu.tw/~cjlin/libsvm`
25. Nilsback, M.E., Zisserman, A.: A visual vocabulary for flower classification. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, vol. 2, pp. 1447–1454 (2006)
26. Fei-Fei, L., Fergus, R., Perona, P.: Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. Comput. Vis. Image Underst. 106, 59–70 (2007)
27. Mairal, J., Bach, F., Ponce, J., Sapiro, G.: Online dictionary learning for sparse coding. In: ICML 2009: Proceedings of the 26th Annual International Conference on Machine Learning, pp. 689–696. ACM, New York (2009)
28. Raina, R., Battle, A., Lee, H., Packer, B., Ng, A.Y.: Self-taught learning: transfer learning from unlabeled data. In: ICML 2007: Proceedings of the 24th International Conference on Machine Learning, pp. 759–766. ACM, New York (2007)
29. Berg, A.C., Berg, T.L., Malik, J.: Shape matching and object recognition using low distortion correspondence. In: CVPR 2005: Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 1, pp. 26–33. IEEE Computer Society, Los Alamitos (2005)
30. Mutch, J., Lowe, D.G.: Multiclass object recognition with sparse, localized features. In: CVPR 2006: Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Washington, DC, USA, pp. 11–18. IEEE Computer Society, Los Alamitos (2006)
31. Zhang, H., Berg, A.C., Maire, M., Malik, J.: SVM-KNN: Discriminative nearest neighbor classification for visual category recognition. In: CVPR 2006: Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 2126–2136. IEEE Computer Society, Los Alamitos (2006)
32. Lin, Y.Y., Liu, T.L., Fuh, C.S.: Local ensemble kernel learning for object category recognition. In: CVPR 2007: Proceedings of the 2007 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 1–8. IEEE Computer Society, Los Alamitos (2007)
33. Cao, L., Luo, J., Liang, F., Huang, T.S.: Heterogeneous feature machines for visual recognition. In: ICCV 2009: Proceedings of the Tenth IEEE International Conference on Computer Vision (ICCV 2009). IEEE Computer Society, Los Alamitos (2009)