SEMANTIC-SPATIAL MATCHING FOR IMAGE CLASSIFICATION

Yupeng Yan[1] *Xinmei Tian*[1] *Linjun Yang*[2] *Yijuan Lu*[3] *Houqiang Li*[1]

[1] University of Science and Technology of China, Hefei Anhui, China
[2]Microsoft Corporation, One Microsoft Way, Redmond, WA, USA
[3]Texas State University, San Marcos, Texas, USA
yanyp@mail.ustc.edu.cn, {xinmei, lihq}@ustc.edu.cn, linjuny@microsoft.com, lu@txstate.edu

ABSTRACT

Spatial Pyramid Matching (SPM) has been proven a simple but effective extension to bag-of-visual-words image representation for spatial layout information compensation. SPM describes image in coarse-to-fine scale by partitioning the image into blocks over multiple levels and the features extracted from each block are concatenated into a long vector representation. Based on the assumption that images from the same class have similar spatial configurations, SPM matches the blocks from different images according to their spatial layout, by aligning all blocks from an image in a fixed spatial order. However, target objects may appear at any location in the image with various backgrounds. Therefore, the fixed spatial matching in SPM fails to match similar objects located different locations. To solve this problem, we propose an effective and efficient block matching method, Semantic-Spatial Matching (SSM). In this method, not only the spatial layout but also the semantic content is considered for block matching. The experiments on two benchmark image classification datasets demonstrate the effectiveness of SSM.

Index Terms— Spatial matching, image classification, bag-of-visual-words, semantic space

1. INTRODUCTION

Visual representation of images plays a fundamental role in image classification. In recent years, local feature representation has shown its superiority due to its robustness to backgrounds, occlusions, *etc*. Bag-of-visual-words (BOVW) [1] model has been widely used for local feature image representation and has demonstrated promising performance in many applications [2, 3, 4]. In BOVW, a visual codebook is constructed first by clustering a set of local descriptors, such as SIFT [5], extracted from a training image set. Then by quantizing all the local descriptors into the visual words in the codebook, each image can be represented as a histogram of the visual words count.

This work is supported in part by the NSFC 61201413, SRFDP 2100060003, the Fundamental Research Funds for the Central Universities No. WK2100060007 and No. WK2100100021 to Dr. Xinmei Tian, in part by the Texas State University Research Enhancement Program (REP), Army Research Office grant W911NF-12-1-0057, and NSF CRI 1058724 to Dr. Yijuan Lu.

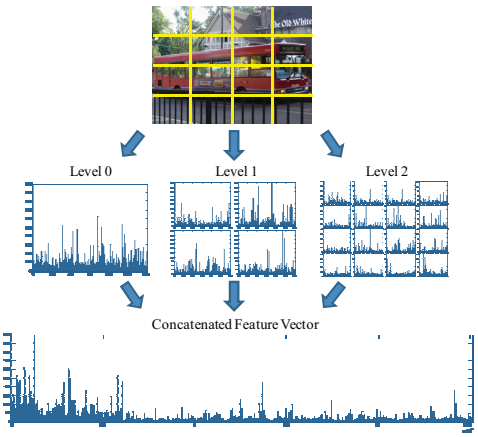

Fig. 1. Illustration of 3-level SPM image representation.

Although BOVW has shown its success and popularity, one big issue is that it represents an image as an orderless distribution of local features, which ignores the spatial layout of local features completely. Therefore, many efforts have been made to capture the spatial information, for example, computing visual word correlation [6], conducting spatial pyramid matching (SPM) [3] and spatial pooling [7, 8], bundling visual words in MSER regions [9, 10], and identifying visual phrases [11, 12], *etc*. With the advantages of simplicity and efficiency, SPM has drawn a lot of attentions and has been widely applied in many applications.

SPM describes an image in coarse-to-fine scale by partitioning the images into blocks over multiple levels and the features extracted from each block are concatenated into a long vector representation, as illustrated in Fig. 1. The underline assumption is that images from the same class have similar spatial configurations. Based on this assumption, when the vectors from blocks are concatenated, they are aligned according to their spatial locations in the image. In other words, the blocks from different images are spatially matched.

The assumption in SPM maybe works in certain situation, for example, the scene classification investigated in [3]. However, it is not true for real Web images which have rich and complex content. Fig. 2 shows several sample images from

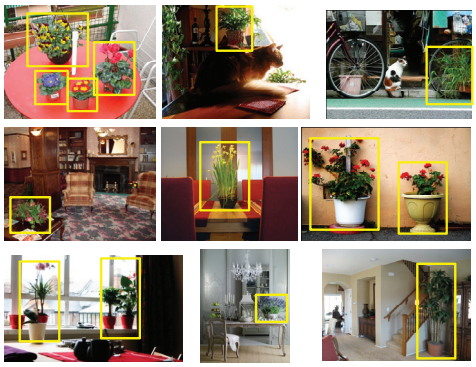

Fig. 2. Sample images from the "potted plant" category in VOC Challenge 2011 dataset. It shows that the target "potted plant" may locate at various positions. In other words, they are not spatially matched between different images.

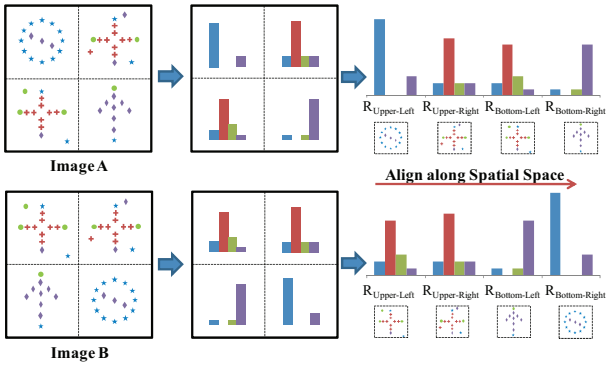

Fig. 3. Illustration of spatial matching on images A and B. Images A and B have similar content, but the SPM feature vectors are dissimilar.

the "potted plant" category in the PASCAL VOC Challenge 2011 [13], which consists of images collected from Flickr. The target object "potted plant" may occur at any location in the image with various backgrounds. In this case, if we simply apply SPM to concatenate the histograms from all blocks in a fixed spatial order, false matching problem will arise. Here we take a toy example for further illustration. In Fig. 3, there are two images A and B, both containing four objects. But these objects' locations in A and B are different. For simplicity, we only take the Level 1 of SPM as an example. In SPM, the features extracted from four blocks are concatenated along a fixed spatial space order (from upper-left to upper-right and from bottom-left to bottom-right). As shown in Fig. 3, we find that although images A and B are very similar, their feature vectors derived from SPM are totally different. Obviously, such fixed spatial space matching in SPM cannot match similar objects located in different locations in images.

The major reason why SPM fails is that it doesn't take the blocks' content into consideration. Two blocks from different images should be matched mainly because they have similar content (*semantic matching*), not just because they have same locations (upper-left, for example) in the images (*spatial matching*). Inspired by this, we propose a new matching method, *Semantic-Spatial Matching* (SSM), which considers not only the spatial layout, but also the blocks' content information. To conduct the semantic matching efficiently, SSM constructs a unified semantic space and all blocks are mapped into this space for alignment. For spatial matching, the sophisticated SPM is directly applied. Then, the semantic matching and spatial matching are fused via linear kernel combination to derive SSM. SSM has the advantage of high efficiency, simple implementation, and robust to rotation, flipping, translation variances.

The rest of this paper is organized as follows. In Section 2, the related works are briefly introduced. SSM is described in Section 3. In Section 4, experimental results are reported, followed by the conclusion in Section 5.

2. RELATED WORK

To recover the spatial information ignored in BoVW representation, many methods have been proposed. The popular SPM [3] solves this problem by partitioning images into coarse-to-fine sub-blocks and concatenating the histograms extracted from all blocks. In SPM, the images are partitioned into non-overlapped blocks with equal size. Cao *et al.* [8] proposed spatial-bag-of-features to extend SPM by introducing two different ways for image partition. The first one is *linear ordered bag-of-features*, in which image is partitioned into straps along a line with an arbitrary angle. The second one is *circular ordered bag-of-features*, in which a center point is given and then the image is evenly divided into several sectors with the same radian. By enumerating different line angles (ranging from 0° to 360°) and center locations, a family of linear and circular ordered bag-of-features can be obtained. Spatial-bag-of-features still concatenates features from divided blocks/straps/sectors in a fixed spatial order. The difference between SPM and spatial-bag-of-features is the way they partition the images. The additional problem of spatial-bag-of-feature is that it needs enumerating a huge number of possible line angles and center locations, therefore resulting in an extremely high dimensional histogram representation for an image, which suffers from high computational cost in real-time application. Li *et al.* [7] also proposed several spatial pooling methods, including *spatial pyramid ring*, *reordered SPM*, and *relative SPM*, dealing with the rotation, flipping, and translation variance respectively. Spatial pyramid ring partitions the image into concentric rings on the polar coordinate, while relative SPM partitions the image in a similar way as SPM but adjusts the partitioning center along with the objects' positions. In reordered SPM, the image is partitioned exactly the same as that in SPM, but the visual words are ordered based on their frequency in different regions. Again, those methods still belong to the spatial matching category, *i.e.*, partitioning images into sub-regions in different ways and then concatenating their features in a fixed spatial order. Xu *et al.* [14] proposed *spatially aligned pyramid matching* method for near duplicate image identifi-

cation. In this method, it partitions images into blocks and examines the optimal block matching between any two-image pair by using Earth Mover Distance [15]. This method lacks unified matching order for all images to derive a general visual representation, and has the drawback of high computational cost.

3. SEMANTIC-SPATIAL MATCHING

In spatial matching, each image is partitioned into a set of sub-regions (blocks/straps/sectors/rings). The key problem is how to match regions from different images correctly. One straightforward solution is to compare all regions from different images pair-wisely and find the best match via certain optimization criteria [14]. For example, the pair-wise region matching result of the toy example (Fig. 3) is given in Fig. 4. However, such pair-wise matching approach has the following drawbacks. First, it is time consuming since all regions need to be compared pair-wisely and a complex programming problem needs to be solved. Second, it lacks unified matching order, therefore suffering the problem that images cannot be represented by a common feature vector for further applications. Third, the matching is not perfectly "one versus one". For a region in one image, there may not exist any matched region in other images, or may have multiple regions matched. It is not easy to deal with the "multiple versus multiple" region matching problem.

To solve these problems, we propose a new matching method, named semantic matching (SM). In this method, we construct a unified semantic space and all regions are mapped into this space for alignment. This method is very easy for implementation and highly efficient.

Semantic Space Construction: The key step in SM is the semantic space construction. We achieve this by clustering all regions into groups, and each cluster center represents one semantic subspace. Specifically, given a set of images, each image is partitioned into sub-regions, as done in SPM. Let $\mathcal{R} = \{\mathbf{r}_1, \mathbf{r}_2, \cdots, \mathbf{r}_N\}$ be the set of all regions obtained, where $\mathbf{r}_i \in \mathcal{R}^D$ is the feature vector (region histogram) of the i-th region. We apply the K-means clustering algorithm to construct the semantic space by solving the following problem,

$$\min_{\mathcal{S}} \sum_{i=1}^{N} (\min_{k=1,\cdots,K} ||\mathbf{r}_i - \mathbf{s}_k||^2), \tag{1}$$

where $\mathcal{S} = \{\mathbf{s}_1, \mathbf{s}_2, \cdots, \mathbf{s}_K\}$ is the set of the K cluster centers obtained, which represents the semantic space. $\mathbf{s}_k$ represents the k-th semantic subspace.

Semantic Matching: With the constructed semantic space $\mathcal{S}$, each region can be assigned a semantic label by finding its nearest neighbor in $\mathcal{S}$. Regions from different images are defined as matched if they have the same semantic label. Instead of conducting the pair-wise region matching via semantic labels, we can define a fixed semantic order (for example $\mathbf{s}_1 \rightarrow \mathbf{s}_2 \rightarrow, \cdots, \rightarrow \mathbf{s}_K$) and all regions from an image can be aligned according to this order.

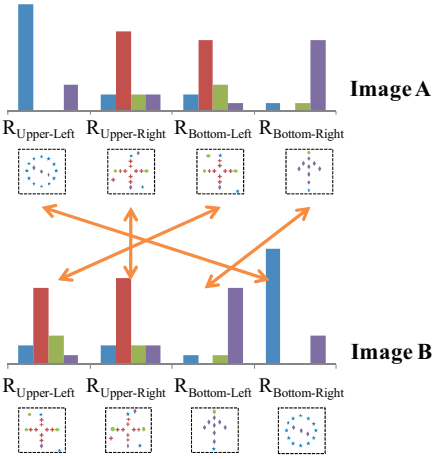

Fig. 4. The ideal region matching between images A and B.

Mathematically, given an image I, it is first partitioned into M regions, $\mathcal{R}_I = \{\mathbf{r}_1, \mathbf{r}_2, \cdots, \mathbf{r}_M\}$, where $\mathbf{r}_i \in \mathcal{R}^D$ is the feature vector (region histogram) of the i-th region. Each $\mathbf{r}_i$ is mapped to a semantic subspace by finding its nearest neighbor in $\mathcal{S}$. If $\mathbf{s}_k$ is the nearest neighbor of $\mathbf{r}_i$, we call the semantic label of $\mathbf{r}_i$ is $\mathbf{s}_k$, denoted as $\mathbf{r}_i \in \mathbf{s}_k$. When all the regions in $\mathcal{R}_I$ have been assigned their corresponding semantic labels, these regions are aligned and their histograms are concatenated in a fixed semantic order ($\mathbf{s}_1 \rightarrow \mathbf{s}_2 \rightarrow, \cdots, \rightarrow \mathbf{s}_K$). The final concatenated representation of image I is,

$$V_I(\mathbf{SM}) = [\mathbf{r}_{\mathbf{s}_1}^T, \mathbf{r}_{\mathbf{s}_2}^T, \cdots, \mathbf{r}_{\mathbf{s}_K}^T]^T \tag{2}$$

where $\mathbf{r}_{\mathbf{s}_k} = \sum_{i, \mathbf{r}_i \in \mathbf{s}_k} \mathbf{r}_i$ is the sum of regions' histograms with the same semantic label $\mathbf{s}_k$. If there is no region labeled as $\mathbf{s}_k$, $\mathbf{r}_{\mathbf{s}_k}$ is a D-dimensional vector with all elements as 0.

SM first classifies the regions into different semantic classes via simple 1-NN, and then all regions are aligned in a fixed semantic order. Since SM matches regions according to their semantic content information, it can handle the rotation, flipping, and translation problem well. Besides, by controlling the number of semantic labels K, *i.e.* $|\mathcal{S}|$, we can get semantic spaces in different granularity. A small K leads to a coarse partition of the semantic space, while a large K leads to a fine partitioned one. With this granularity control, SM has strong tolerance to noise. For example, in our toy example, if we set $K = 4$, the resulting SM representation is the ideal case as illustrated in Fig. 4. If we set $K = 3$, the results are illustrated in Fig. 5. Here, the obtained semantic space $\mathcal{S}$ consists of three semantic labels {"ring", "cross", and "arow"} since the two similar "cross" objects are clustered into the same semantic class.

Semantic-Spatial Matching: Spatial matching methods [3, 7, 8] divide an image into a set of regions and align their feature histograms along fixed spatial order. Semantic matching aligns regions from an image along a pre-defined semantic order. It is natural to combine these two complementary models together and generate a Semantic-Spatial Matching (SSM). Semantic matching can be combined with any spa-

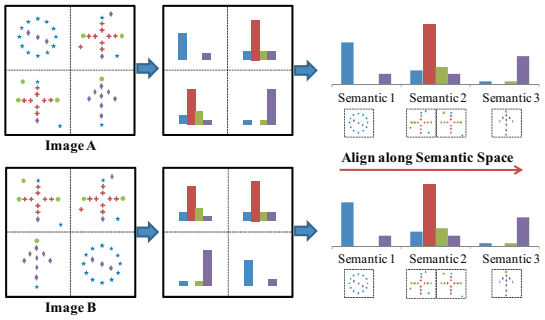

Fig. 5. Illustration of the Semantic Matching.

Fig. 6. Example images of the VOC2011 database.

tial matching method [3, 7, 8]. Here we take SPM [3] as an example for its popularity.

For SPM and SM combination, the most straightforward way is to concatenate the SPM and SM feature vectors into a long one,

$$V_I = [V_I(\mathbf{SPM})^T, V_I(\mathbf{SM})^T]^T \quad (3)$$

where $V_I(\mathbf{SPM})$ is the feature vector derived from SPM. Due to the popularity of kernel based methods in classification and in order to control the independent influence of SPM and SM, we adopt a general combination model, *i.e.*, linear kernel combination,

$$\mathbf{K}_{SSM}(I_i, I_j) = \alpha \mathbf{K}_{SM}(I_i, I_j) + (1-\alpha)\mathbf{K}_{SPM}(I_i, I_j) \quad (4)$$

where $\mathbf{K}_{SM}(I_i, I_j) = K(V_{I_i}(\mathbf{SM}), V_{I_j}(\mathbf{SM}))$, $\mathbf{K}_{SPM}(I_i, I_j) = K(V_{I_i}(\mathbf{SPM}), V_{I_j}(\mathbf{SPM}))$, and $\alpha \in [0, 1]$ is the combination coefficient for controlling their effects. Various kernels (Linear, Radial Basis Function, Polynomial) can be adopted here. Eq.(3) is a special case when the linear kernel is adopted with certain α.

Space and Time Complexity Analysis: The semantic space construction in SSM is conducted via *K*-means clustering. It has the time complexity of *O(LKND)* and space complexity of *O((N+K)D)*, where *K* is the number of clusters, *L* is number of iterations, *N* is the number of training regions, and *D* is the dimension of region's feature vectors. It should be noted that the semantic space can be constructed offline and only needs to be learned once. For the online SM feature generation, the only computational cost is to find the semantic label for each region, which is very fast (*O(KD)*). For the storage cost, once the semantic space is trained, we only need to record *K* clustering centers with space complexity of *O(KD)*. Therefore, SSM extends SPM with very little additional computational and storage cost.

4. EXPERIMENTS

4.1. Experiments on VOC 2011

4.1.1. Experimental Setting

We conduct extensive experiments to test our proposed SSM method on two benchmark image classification datasets. The first testing dataset is VOC 2011 [13]. It contains 14961 images from 20 categories and the average image size is 500×375. Fig. 6 shows some example images of this dataset. This dataset is quite challenging. It covers diverse object categories and the images have very complex content, as shown in Fig. 2 of "potted plant".

We follow the standard experiment setup for VOC2011, *i.e.,* 5717 images for training and 5823 images for testing. For the local feature, scale-invariant feature transform (SIFT) [5] is extracted from each image on a dense grid. The codebook size is 600. We adopt the linear kernel SVM [3] due to its efficiency. We train SVM [2, 16] classification models for each category on the training set and report the classification performance on the testing set in terms of the non-interpolated average precision (AP) [13, 17, 18, 19]. We compare the proposed model with spatial pyramid matching (SPM) [3] and recently proposed reordered SPM (R-SPM) [7].

Table 1. mAP comparison on VOC2011

Algorithm	mAP(%)
BoVW	28.93
R-SPM	30.42
SPM	37.69
SSM	**40.79**

4.1.2. Experimental Results

The mAP, average of AP over all 20 categories, is reported in Table 1. It shows that both spatial matching methods, R-SPM and SPM, can improve the baseline BoVW to some extent, that validates the advantage of taking spatial layout information into consideration. The SSM method outperforms both R-SPM and SPM. It demonstrates the effectiveness of the combination of semantic and spatial information. We also investigate the effectiveness of SSM on each category. Fig. 7 shows the AP on each category as well as mAP. From Fig. 7, we can see that SSM performs the best on 19 categories, and only suffers a slight AP decrease from SPM in the category "Sheep". Overall, it outperforms R-SPM and SPM stably.

In SSM, there are two important parameters, *i.e.*, the combination coefficient α and the semantic space size *K*. We have conducted a series of experiments to investigate the sensitivity of SSM to them. The combination coefficient α in Eq. (4)

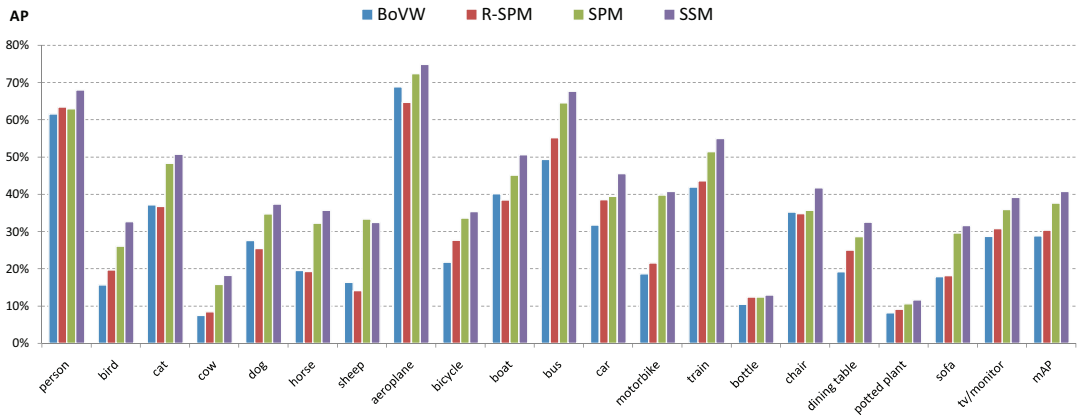

Fig. 7. The performance comparison of BoVW, R-SPM, SPM, and SSM on each object category.

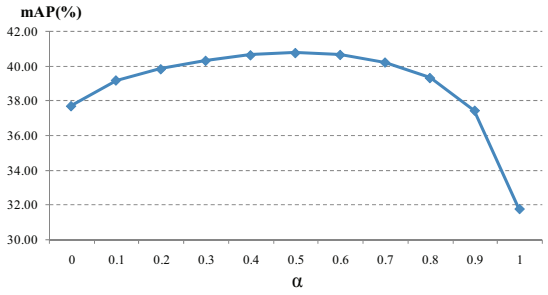

Fig. 8. The performance of SSM with different α.

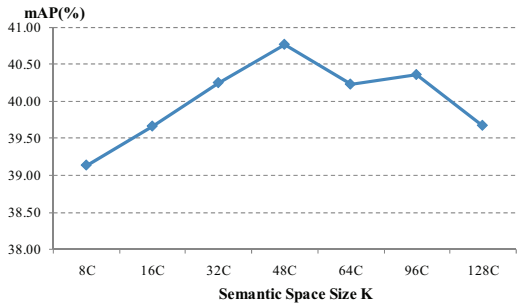

Fig. 9. The performance of SSM with different semantic space size K.

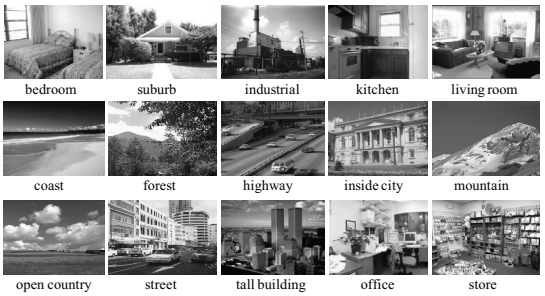

Fig. 10. Example images of the 15 Scene dataset

controls the influence of SM and SPM, reflecting the importance of semantic and spatial matching. In the special case, when $\alpha = 0$, SSM degrades to SPM, and when $\alpha = 1$, SSM degrades to SM. We vary α from 0 to 1 with interval 0.1, and the results are plotted in Fig. 8. It shows that SSM achieves the best performance at $\alpha = 0.5$ which implies that semantic matching and spatial matching are equally important.

To investigate the effects of semantic space size K, *i.e.*, the number of clusters in K-means for constructing the semantic space, we test various Ks, from 8 to 128, as shown in Fig. 9. As discussed in Section 3, K controls the granularity of the semantic space. When K is too small, the semantic space has low discriminative power of distinguishing different regions, causing regions with different content falling into the same semantic subspace. When K is too large, the semantic space is over-split and thus it has little robustness to noise, translations or other variances. From the experiments, we find that a moderate $K = 48$ is a good choice.

4.2. Experiments on 15 Scene

4.2.1. Experimental Setting

We also test our algorithm on the 15 Scene dataset [20, 21]. This dataset consists of 4485 images from fifteen scene categories, varying from bedroom and coast to store and mountain. The number of images in each category ranges from 200 to 400. The average image size is 300×250 pixels. Fig. 10 shows some example images of this dataset.

We follow the same experimental setting as in Lazebnik *et al.* [3] and Yang *et al.* [2]. We randomly select 100 images from each class for training and apply the linear kernel SVM for multi-class classification. The codebook size is 400. The random sampling process is repeated 10 times, and the average classification rate and standard deviation are reported.

4.2.2. Experimental Results

The experimental results on 15 scene dataset are given in Table 2. Classification rate shows the percentage of the images which are correctly classified. It is found that both SPM and SSM outperform the baseline BoVW significantly. Compared with SPM, SSM achieves limited classification accuracy improvement. This dataset has been well investigated in [3] and SPM is demonstrated working well on it, since the scene categories generally satisfy the similar spatial config-

Table 2. Classification rate (%) on 15 Scene

Algorithm	Classification Rate (%)
BoVW	43.51 ± 0.96
SPM	76.62 ± 0.78
SSM	$\mathbf{77.02 \pm 0.82}$

Table 3. Mean Average Precision (%) on 15 Scene

Algorithm	mAP (%)
BoVW	32.75 ± 0.55
SPM	78.86 ± 0.53
SSM	$\mathbf{80.11 \pm 0.57}$

urations assumption. Even though, the combined SSM still outperforms SPM. It demonstrates that the semantic matching also makes important contribution in scene classification. We further compare these state-of-the-arts in terms of AP on 15 scenes and report their mAPs in Table 3. We can find that SSM achieves 1.25% AP improvement over SPM.

5. CONCLUSION

In this paper, we propose a new matching method, Semantic-Spatial Matching (SSM). SSM conducts region matching by considering both the spatial layout and the semantic content information. SSM has the advantage not only being robust to rotation, flipping and other variances, but also simple and easy for implementation. Experiments on two benchmark datasets demonstrate its effectiveness in object and scene classifications.

6. REFERENCES

[1] J. Sivic and A. Zisserman, "Video google: a text retrieval approach to object matching in videos," *ICCV*, pp. 1470–1477, 2003.

[2] J. Yang, K. Yu, Y. Gong, and T. Huang, "Linear spatial pyramid matching using sparse coding for image classification," *CVPR*, pp. 1794–1801, 2009.

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

[4] G. Csurka, J. Willamowski C.R. Dance, L. Fan, and C. Bray, "Visual categorization with bags of keypoints," *ECCV Workshop on SLCV*, pp. 1–22, 2004.

[5] D.G. LOWE, "Distinctive image features from scale-invariant keypoints," *IJCV*, vol. 60, no. 2, pp. 91–110, 2004.

[6] S. Savarese, J. Winn, and A. Criminisi, "Discriminative object class models of appearance and shape by correlations," *CVPR*, pp. 2033–2040, 2006.

[7] X. Li, Y. Song, Y. Lu, and Q. Tian, "Spatial pooling for transformation invariant image representation," *ACM Multimedia*, pp. 1509–1512, 2011.

[8] Y. Cao, C. Wang, Z. Li, L. Zhang, and L. Zhang, "Spatial-bag-of-features," *CVPR*, 2010.

[9] Z. Wu, Q. Ke, M. Isard, and J. Sun, "Bundling features for large-scale partial-duplicate web image search," *CVPR*, 2009.

[10] J. Matas, O. Chum, M. Urban, and T. Pajdla, "Robust wide baseline stereo from maximally stable extremal regions," *BMVC*, 2002.

[11] S. Zhang, Q. Tian, G. Hua, Q. Huang, and S. Li, "Descriptive visual words and visual phrases for image applications," *ACM Multimedia*, 2009.

[12] J. Yuan, Y. Wu, and M. Yang, "Discovery of collocation patterns: from visual words to visual phrases," *CVPR*, pp. 1–8, 2007.

[13] M. Everingham, L. Van-Gool, C. Williams, J. Winn, and A. Zisserman, "The Pascal Visual Object Classes Challenge 2011 (VOC2011) Results," *http://www.pascal-network.org/challenges/VOC/voc2011/workshop/index.html*.

[14] D. Xu, T.-J. Cham, S. Yan, and S.-F. Chang, "Near duplicate image identification with spatially aligned pyramid matching," *CVPR*, 2008.

[15] Y. Rubner, C. Tomasi, and L.J. Guibas, "The earth mover's distance as a metric for image retrieval," *IJCV*, vol. 40, no. 2, pp. 99–121, 2000.

[16] Zheng-Jun Zha, Xian-Sheng Hua, Tao Mei, Jingdong Wang, Guo-Jun Qi, and Zengfu Wang, "Joint multi-label multi-instance learning for image classification," *CVPR*, pp. 1–8, 2008.

[17] "Trecvid video retrieval evaluation," *http://www.nlpir.nist.gov/projects/trecvid/*.

[18] Meng Wang, Xian-Sheng Hua, Richang Hong, Jinhui Tang, Guo-Jun Qi, and Yan Song, "Unified video annotation via multigraph learning," *IEEE TCSVT*, vol. 19, no. 5, pp. 733–746, 2009.

[19] Jinhui Tang, Zheng-Jun Zha, Dacheng Tao, and Tat-Seng Chua, "Semantic-gap-oriented active learning for multilabel image annotation," *IEEE TIP*, vol. 21, no. 4, pp. 2354–2360, 2012.

[20] Li F.-F. and P. Pietro, "A bayesian hierachical model for learning natural scene categories," *CVPR*, 2005.

[21] A. Oliva and A. Torralba, "Modeling the shape of the scene: a holistic representation of the spatial envelope," *IJCV*, vol. 42, no. 3, pp. 145–175, 2001.