Exploring Self-Similarities of Bag-of-Features for Image Classification[*]

Chih-Fan Chen
Research Center for IT Innovation
Academia Sinica, Taipei, Taiwan
ryanchen@citi.sinica.edu.tw

Yu-Chiang Frank Wang
Research Center for IT Innovation
Academia Sinica, Taipei, Taiwan
ycwang@citi.sinica.edu.tw

ABSTRACT

The use of bag-of-features (BOF) models has been a popular technique for image classification and retrieval. In order to better represent and discriminate images from different classes, we advance BOF and explore the self-similarities of visual words for improved performance. The proposed *self-similarity hypercubes* (SSH) model, which observes the concurrent occurrences of visual words in an image, is able to describe the structural information of the BOF in an image. Our experiments confirm that our SSH provides additional and complementary information to BOF and thus results in improved classification performance. Unlike most prior methods requiring extraction or integration of multiple types of features for similar improvements, our SSH works in the same domain as the BOF does. Moreover, we do not limit the use of our SSH to any particular type of image descriptors, and its generalization is also verified.

Categories and Subject Descriptors

H.3.1 [**Information Storage & Retrieval**]: Content Analysis and Indexing; I.4.7 [**Image Processing & Computer Vision**]: Feature Measurement—*feature representation*

General Terms

Algorithms, Experimentation, Performance

Keywords

Image classification, bag-of-features, self-similarity

1. INTRODUCTION

Due to the growth of Web-based image applications such as Flickr and Google Images, lots of attention has been directed to image classification and retrieval for researchers in multimedia communities. Among existing methods, the *bag-of-features* (BOF) model [1, 16, 10] has been a popular technique due to its simplicity and effectiveness. BOF quantizes local image descriptors into distinct visual words, and it uses a histogram representation to show the number of occurrences of each word for each image. Several works have been proposed to improve the BOF model [9, 10]. Some proposed to preserve the spatial information of descriptors, while others combined different features to boost the performance [6, 12]. Although promising results were reported, the increase of computation complexity in extracting and fusing different features, plus high feature dimensionality, would prohibit the use of these methods in large-scale problems.

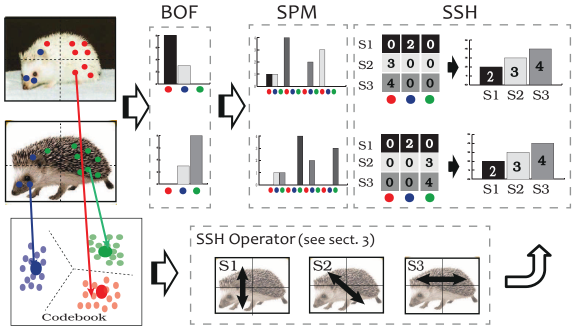

Figure 1: Advantages of our self-similarity hypercube (SSH) model over standard BOF and SPM (see Sect. 2).

In this paper, we utilize the *self-similarities* of visual words and propose the *self-similarity hypercubes* (SSH) feature model (see Section 3). We aim at utilizing the *related* spatial information of visual words present in images (see Figure 1 for example), and such structural information will provide additional representation ability to the standard BOF. As a results, improved classification performance can be expected. While the self-similarities of image *patches* or *descriptors* have been investigated in literature (e.g., [15, 3], these prior approaches typically required extraction and processing of self-similarity features in the associated feature domains. In our work, we explore the self-similarity of existing visual words; in other words, we do not require additional feature extraction or selection processes. Our experiments will verify that our method is not limited to the use of appearance features (e.g., SIFT [11]), and we can extend our SSH to HoG [2] or other image descriptors.

2. RELATED WORK

2.1 BOF and SPM for Image Classification

Among approaches for image classification, the *bag-of-features* (BOF) model [1, 16, 10] has demonstrated its suc-

[*] Area chair: Lexing Xie

MM'11, November 28–December 1, 2011, Scottsdale, Arizona, USA.

cess in recent years. BOF considers each image as a collection of *unordered* local image descriptors. To represent an image, BOF quantizes these descriptors into distinct *visual words*, and it uses a histogram to indicate the number of occurrences of each visual word. To determine the visual words (i.e., *dictionary* or *codebook*) from images, one typically performs k-means, etc. clustering algorithms and divide the image descriptors into distinct groups, and the representative of each group (e.g., cluster mean) is thus considered as a visual word. Once the BOF is obtained, one can design classifiers such as SVM for classification.

As mentioned in [9], BOF discards the spatial order of image descriptors and thus limits the classification performance. To address this problem, spatial pyramid matching (SPM) is proposed to partition an image into several grids in different scales. The BOF models pooled from different grids are concatenated as the final SPM model to preserve the spatial order of local descriptors (and the associated visual words) [9]. However, take a SPM with three image scales for example, the dimension of a SPM feature vector will be $(1+2^2+4^2) = 21$ times larger than that of the BOF, which inevitably increases the computation time.

2.2 Self-Similarity

Figure 1 shows an example in which standard BOF/SPM models are not sufficient to represent different images from the same object category. Comparing the two BOF (or SPM) models, it is obvious that they are not similar to each other due to texture, etc. appearance variations. Shechtman and Irani [15] first proposed to utilize the *local self-similarity* (LSS) of patches present in an image. By exploring the neighborhood of each pixel, LSS uses such information to detect object images containing similar local structures as the query input does. Later, Deselaers *et al.* [3] advocated to construct BOF models on LSS. They learned a codebook based on LSS descriptors, and they utilized the *global* self-similarity to describe the structural information of the entire image. In order to achieve improved performance, their method needs to determine and extract self-similarity features prior to the integration of their resulting features and the standard BOF (e.g., from SIFT, HoG [2], or GIST [14] descriptors). In this paper, we advance the self-similarity of the existing BOF models. Unlike [15, 3], our method need not extract different types of features or learn extra codebooks. We aim at exploring the self-similarity of existing visual words, and we do not limit our method to the use of any particular image descriptors. Therefore, our proposed feature model is computationally more preferable than prior works requiring combination of different types of features (e.g., [6]) for improved performance.

3. OUR PROPOSED METHOD

3.1 Learning of BOF via Sparse Coding

Since our approach extracts the self-similarities of visual words from the existing BOF model, we briefly discuss the construction of BOF in this subsection. Take the codebook constructed from SIFT features for example (see Figure 2(a)), we first extract dense SIFT descriptors from images, and we quantize these descriptors into distinct visual words. Figure 2(b) shows example quantization results, in which each colored number indicates the visual word assigned to the associated descriptor. Instead of using vector quantization (VQ), we apply sparse coding (SC) [13] to construct an over-complete codebook and encode the descriptors accordingly. The final BOF model is produced by the *max-pooling* operation on the encoded descriptors. It has been verified in [17] that such SC-based BOF models outperform those constructed by VQ.

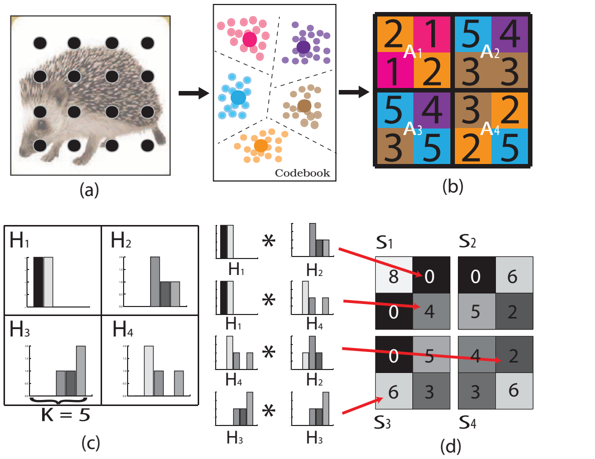

Figure 2: Our proposed SSH model. (a) Extracting dense image descriptor for codebook learning. (b) Encoding the descriptors with the associated visual words. (c) Construction of BOF for each grid. (d) Performing inner products S_1, S_2, etc. between BOF from different grids to obtain visual word self-similarities. The final SSH model is determined by Eq. (2).

Since the detailed discussion of SC is beyond the scope of this paper, we only highlight this BOF learning process. Suppose that we have N SIFT descriptors extracted from images, SC learns an over-complete codebook with size K, which solves the following optimization problem:

$$\min_{D,\alpha} \sum_{i=1}^{N} \frac{1}{2} \|x_i - A\alpha_i\|_2^2 + \lambda \|\alpha_i\|_1, \qquad (1)$$

where x_i is the image descriptor, A is the codebook with K visual words, α_i is the associated sparse coefficient vector, and λ controls the sparsity. To produce the final K-dimensional BOF model for an image, all encoded α_i vectors will be summed up by a dimension-wise max-pooling operation. Once the codebook A is learned during training, we only need to encode the descriptors of the test image and calculate its BOF accordingly. If a SPM model is of interest, a K-dimensional BOF will be pooled from each grid in each image scale. For example, a SPM model with three image scales will result in a $21K$ dimensional feature vector.

3.2 Learning of Self-Similarity Hypercubes

In this paper, we advocate to explore the *self-similarity* of visual words within the BOF model. We propose to construct the *self-similarity hypercubes* (SSH) features for each image. The purpose of our SSH is to preserve the structural information of visual words present in images, and our experiments will confirm that such information results in improved classification performance. While the term SSH has been used by Deselaers and Ferrari [3], they focus on extracting the global structure of local self-similarity features in *pixel* domain. Our work is very unique, since our SSH extends the existing BOF and does not require to determine/extract additional self-similarity features in other domains as [15, 3] did.

To construct our SSH model, we first divide an image into several sub-regions. For example, we segment the input image in Figure 2(a) into $D \times D = 2 \times 2 = 4$ grids, which are denoted as A_1 to A_4 in Figure 2(b). Recall that the numbers in each grid indicate the visual words assigned to the corresponding image descriptors. As suggest by [17], we construct the BOF models for each grid by max-pooling and produce the four histogram representations $\{H_1, H_2, H_3, H_4\}$ accordingly (as shown in Figure 2(c)).

Instead of directly concatenating these representations (like SPM), we exploit the self-similarity of visual words present in different grids. More precisely, we start from H_1 and calculate *inner products* between H_1 and different H_i. The resulting vector is denoted as $S_1 = [s_{11}, s_{12}, s_{13}, s_{14}]^T$, which contains structural information of visual words present in this image. Recall that each attribute in the BOF model H_i indicates the number of occurrences of each visual word, and the inner product between different H_i implies the similarity between those BOF models (and the associated visual words) present in an image.

Take s_{14} for example, $s_{14} = H_1^T H_4 = 4$, which calculates the visual word similarity between H_1 and H_4 in Figure 2(c). More precisely, s_{14} sums up the numbers of occurrences of each visual word *concurrently* present in grids A_1 and A_4. Therefore, we have $s_{14} = 2 \times 2 + 0 = 4$ (i.e., visual word 2 occurs twice, while none of the remaining words are present in both grids). Similarly, we have $s_{11} = H_1^T H_1 = 2 \times 2 + 2 \times 2 = 8$ representing the visual word self-similarity of grid H_1. After the calculation for the inner product vector S_1 is complete, we repeat this process for S_2, S_3, etc. We note that, since $s_{ij} = H_i^T H_j = s_{ji}$, the final SSH feature can be calcualted as follows:

$$SSH = [s_{11}, s_{12}, \ldots, s_{1D^2}, \ldots, s_{(D^2-1)D^2}]^T, \qquad (2)$$

where $s_{ij} = H_i^T H_j$, and D is the number of grids in each direction. As a result, if we divide an image into $D \times D = 4 \times 4 = 16$ grids, the dimension of SSH is $d_{SSH} = C_2^{D^2} + D^2 = D^2(D^2+1)/2 = 16 \times 17/2 = 136$. It is worth noting that, although our SSH is based on the existing BOF model, the dimensionality of SSH is a function of the number of grids D^2 and is *independent* of the dictionary size K. Once the SSH model is obtained, we simply concatenate the BOF/SPM and SSH models and produce the final feature representation for training and testing.

4. EXPERIMENTS AND RESULTS

4.1 Datasets

We consider two datasets for object recognition tasks: Caltech-101 [5] and Caltech-256 [7]. We randomly select 15 or 30 images per class for training (i.e., N_T =15 or 30). For the remaining images, no more than 50 per class are used for testing. All tests are repeated 10 times, and we present the averaged results with standard deviations. The longer side of each image is resized to 300 pixels ([17] also did this). For dense SIFT extraction, we extract SIFT descriptors from 16 × 16 pixel patches of each image, and the spacing between adjacent patches is 6 pixels. We set λ in (1) as 0.2 for codebook learning. Since we only consider linear SVMs (trained by *liblinear* [4]) for our SSH, our method provides excellent scalability for larger-scale classification problems.

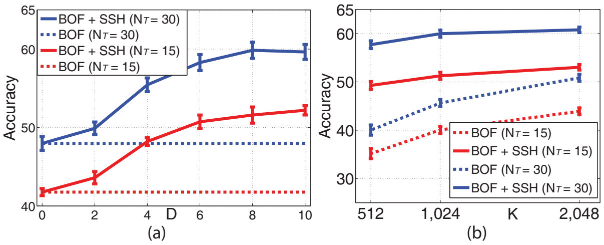

Figure 3: Performance comparisons on Caltech-101 with different SSH and codebook sizes. Note that $D = 0$ corresponds to the use of standard BOF.

4.2 Parameters of SSH

As discussed in Sect. 3.2, our SSH exploits visual word self-similarity, and the dimensionality of SSH is a function of the number of grids in an image, i.e., $d_{SSH} = D^2(D^2+1)/2$. Since the construction of our SSH is based on BOF, we also need to select its codebook size K. Once these parameters are determined (which can be done via cross-validation in practice), we concatenate both BOF and SSH models to form the final feature for classification.

Figure 3(a) shows the recognition performance on Caltech-101 using different D values. We note that $D = 0$ represents the standard BOF (we choose $K = 2048$), and the corresponding accuracy is considered as the baseline (shown in dotted lines). For $D > 0$, our method produces a feature vector BOF + SSH, which is of size $K + d_{SSH}$. From this figure, we observe the increase of accuracy when D goes up. However, for the use of finer grids ($D > 8$ or $D^2 > 64$) in SSH, we did not observe remarkable improvement in accuracy (while a much longer feature vector will be produced). Therefore, we choose $D = 8$ for our SSH, and the dimension of the final feature vector BOF + SSH is $K + d_{SSH} = 2048 + 64(64+1)/2 = 2048 + 2080 = 4128$.

Figure 3(b) shows the recognition performance using different codebook sizes K. It can be seen that, compared with the standard BOF, our approach consistently achieved improved recognition, and such improvements become more significant when smaller K values are of use. From the above results, we validate that additional structural information of visual words provided by our SSH indeed exhibits complementary classification ability, and thus improved performance can be obtained. Some example object categories which benefit from our SSH and thus achieve significant performance improvements are shown in Figure 4.

4.3 SSH of Different BOF models

While one of the advantages of our SSH is the extension of the current BOF model without further extraction or processing of features from other domains, we need to evaluate our SSH on BOF models constructed by different image descriptors. This is to show the generalization of our SSH. Besides the performance on Caltech-101 using BOF of SIFT descriptors, we also consider the use of dense HoG features [2]. To extract HoG for an image, we first divide each image patch into $4 \times 4 = 16$ grids, and each grid is of size 8×8, 6×6, or 4×4 pixels. In other words, the size of each patch is 32×32, 24×24, 16×16 pixels. For the 16 grids of each patch, we calculate 8 directional responses and concatenate the outputs into a 128-dimensional HoG descriptor. Table 1 shows the performances using standard BOF/SPM models using SIFT or HOG descriptors, and those with our method

Table 1: Classification performance using BOF, SPM and SSH with different image descriptors ($N_T = 30$).

Type	BOF	BOF+SSH	SPM	SPM+SSH
Dimension	K (2048)	$K + d_{SSH}$ (4128)	$21K$ (43008)	$21K + d_{SSH}$ (45088)
SIFT	51.5 ± 1.8	64.4 ± 0.8	73.3 ± 0.6	74.7 ± 0.9
HOG (8×8)	48.9 ± 0.6	59.4 ± 1.6	68.4 ± 1.0	69.4 ± 0.9
HOG (6×6)	39.5 ± 1.6	54.2 ± 1.3	64.3 ± 0.9	66.3 ± 0.6
HOG (4×4)	29.0 ± 0.8	47.8 ± 1.1	56.9 ± 0.7	58.7 ± 0.8

Table 2: Performance comparisons.

Database	Caltech 101		Caltech 256	
N_T	15	30	15	30
Griffin [7]	~ 59.0	67.6 ± 1.4	~ 28.10	34.1 ± 0.2
Germent [8]	$--$	66.2 ± 0.5	$--$	27.2 ± 0.5
Yang [17]	67.0 ± 0.5	73.2 ± 0.5	27.7 ± 0.5	34.0 ± 0.4
Ours	$\mathbf{68.2 \pm 0.7}$	$\mathbf{74.7 \pm 0.9}$	$\mathbf{33.0 \pm 0.3}$	$\mathbf{39.7 \pm 0.3}$

(denoted as BOF+SSH and SPM+SSH). From this table, we see that the use of our BOF+SSH or SPM+SSH consistently outperformed that of standard BOF or SPM, and this observation holds for both SIFT and HoG. Therefore, the robustness of our method is confirmed.

It is also worth noting that, although SPM is expected to outperform BOF (see [17]), the dimensionality of SPM is 21 times larger than that of BOF. For large-scale problems, K is typically very large (beyond thousands even millions) and thus the use of SPM is not preferable. From Table 1, one can see that our approach with BOF+SSH results in the feature model whose dimensionality is in *the same order* as that of BOF, while a remarkable improvement (64.4% vs. 51.5%) in accuracy was still achieved.

4.4 Comparisons

Table 2 compares our SSH with state-of-the-art methods on the two datasets. We only consider those using a single type of features (such as BOF and its extensions) for comparisons. Griffin *et al.* [7] first applied the SPM on both datasets, and we use their results as baselines. Although Germent *et al.* [8] proposed a kernel codebook to encoding the SIFT descriptors with more flexibility, negligible differences were observed from the results between [8] and [7]. Yang *et al.* [17] applied the sparse coding technique for BOF and SPM. While their approach achieved improved recognition on Caltech-101, marginal improvements were observed on Caltech-256. Using our SSH (SPM+SSH to be more precise), we obtained recognition rates at 74.7% and 39.7% on the two datasets, and the improvements over the aforementioned works are significant. We note that, although recent methods combining multiple types of features for improved performance exist (e.g., 77.7% and 45.8% on the same datasets were reported in [6]), these methods need to extract features from different domains and typically require more complicated or nonlinear classifiers to combine those features. Our method aims at utilizing the visual word self-similarity, which is not limited to any particular descriptor and does not require additional feature extraction or selection processes. From the above experiments, we successfully confirm the effectiveness of our SSH.

5. CONCLUSIONS

Figure 4: Selected object categories with improved performance by SSH.

In this paper, we advocated the visual word self-similarity for image classification. Based on the BOF model of interest, our proposed SSH model explores the structural information of visual words present in an images by observing concurrent occurrences of the associated visual words. Unlike most prior methods which typically integrate multiple types of features for improved classification performance, our SSH is constructed in the same feature domain as the BOF model does, and thus we do not need extra feature extraction, selection, or fusion processes. From the experimental results, we confirmed that our SSH provides additional and complementary representation and classification ability, and thus both the effectiveness and generalization of our SSH model were successfully verified.

Acknowledgements This work is supported in part by the National Science Council of Taiwan via NSC 100-2221-E-001-018-MY2 and NSC 100-2631-H-001-013.

6. REFERENCES

[1] G. Csurka et al. Visual categorization with bags of keypoints. In *ECCV Workshop*, 2004.

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

[3] T. Deselaers and V. Ferrari. Global and efficient self-similarity for object classification and detection. In *IEEE CVPR*, 2010.

[4] R.-F. Fan et al. Liblinear: A library for large linear classification. In *JMLR*, 9:1871-1874, 2008.

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

[6] P. Gehler and S. Nowozin. On feature combination for multiclass object classification. In *IEEE ICCV*, 2009.

[7] G. Griffin, A. Holub, and P. Perona. Caltech-256 object category dataset. Technical report, 2007.

[8] C. V. J. Gemert et al. Kernel codebooks for scene categorization. In *ECCV*, 2008.

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

[10] D. Li et al. Large-scale robust visual codebook construction. In *ACM MM*, 2010.

[11] D. Lowe. Object recognition from local scale-invariant features. In *IEEE ICCV*, 1999.

[12] H. Ma et al. Bridging the Semantic Gap Between Image Contents and Tags. In *IEEE Trans. Multimedia*, 2010.

[13] J. Mairal, F. Bach, J. Ponce, and G. Sapiro. Online dictionary learning for sparse coding. In *ICML*, 2009.

[14] A. Oliva and A. Torralba. Modeling the shape of the scene: A holistic representation of the spatial envelope. In *IJCV*, 42:145-175, 2001.

[15] E. Shechtman and M. Irani. Matching local self-similarities across images and videos. In *IEEE CVPR*, 2007.

[16] J. Yang et al. Evaluating bag-of-visual-words representations in scene classification. in *ACM MIR*, 2007.

[17] J. Yang et al. Linear spatial pyramid matching using sparse coding for image classification. In *IEEE CVPR*, 2009.