Image and Vision Computing 38 (2015) 65–74

Contents lists available at ScienceDirect

Image and Vision Computing

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

Editor's Choice Article

How to use Bag-of-Words model better for image classification☆

Chong Wang [1], Kaiqi Huang *

National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, China

ARTICLE INFO

Article history:
Received 17 February 2014
Received in revised form 18 September 2014
Accepted 28 October 2014
Available online 4 December 2014

Keywords:
Image classification
Bag-of-Words
Visual vocabulary
Coding
Pooling

ABSTRACT

The Bag-of-Words (BoW) framework is well-known in image classification. In the framework, there are two essential steps: 1) coding, which encodes local features by a visual vocabulary, and 2) pooling, which pools over the response of all features into image representation. Many coding and pooling methods are proposed, and how to apply them better in different conditions has become a practical problem. In this paper, to better use BoW in different applications, we study the relation between many typical coding methods and two popular pooling methods. Specifically, complete combinations of coding and pooling are evaluated based on an extremely large range of vocabulary sizes (16 to 1*M*) on five primary and popular datasets. Three typical ones are 15 Scenes, Caltech 101 and PASCAL VOC 2007, while the other two large-scale ones are Caltech 256 and ImageNet. Based on the systematic evaluation, some interesting conclusions are drawn. Some conclusions are the extensions of previous viewpoints, while some are different but important to understand BoW model. Based on these conclusions, we provide detailed application criterions by evaluating coding and pooling based on precision, efficiency and memory requirements in different applications. We hope that this study can be helpful to evaluate different coding and pooling methods, the conclusions can be beneficial to better understand BoW, and the application criterions can be valuable to use BoW better in different applications.

1. Introduction

Image classification is a fundamental problem in computer vision. It plays a key role in many applications such as image analysis and visual surveillance. In recent years, the Bag-of-Words (BoW) model has been widely used on many popular datasets and competitions, e.g., 15 Scenes [1], Caltech 101 [2], Caltech 256 [3], PASCAL VOC [4] and ImageNet [5]. In BoW, local features are first extracted to construct image representation, which is then fed into a classifier, as shown in Fig. 1. Specifically, the representation is an essential part, which includes two steps:

Coding: Coding means that local features are encoded by a vocabulary and the response of the feature on the vocabulary is generated. The probabilistic strategies [6–9] describe the distribution of local features, while sparse coding methods [10–15] better reconstruct the features. Recently, superior performance has been obtained by some high-dimensional coding methods [16–19].

Pooling: Pooling transforms the response of all local features on a vocabulary into image representation, which is fed into a classifier. Average pooling [6] and maximum pooling [10] are widely used. Recently, weighted average pooling [17] and local pooling [20] have shown better results.

Although many coding and pooling methods have been proposed, there are limited guidelines about how to use them in different applications [21–24]. Boureau et al. [21,22] analyze theoretically how coding and pooling are related based on sample cardinality (the number of local features) under small vocabulary sizes; Chatfield et al. [23] and Huang et al. [24] evaluate typical coding methods under relatively larger vocabulary sizes, but without considering different pooling schemes. Besides, all studies do not evaluate coding and pooling on large-scale datasets for generalization, such as the ImageNet database [5]. Different from the previous studies, in this paper, we consider four aspects:

- To provide systematical user guidelines, the complete combinations of more popular coding methods [15,14] and two popular pooling methods (average, maximum) under an extremely large range of vocabulary sizes (16 to 1*M*) are considered. The maximum vocabulary size (1*M*) is 1000 and 40 times larger than 1024 in [21,22] and 25*k* in [23] respectively.
- Given the fact that large-scale image classification has become much more active in recent years [25–27], we consider two large-scale datasets, namely Caltech 256 [3] and ImageNet [5]. Furthermore, combined with three typical ones including 15 Scenes, Caltech 101 and PASCAL VOC 2007, the evaluation on these primary datasets can provide strong support and generalization for the conclusions and guidelines.

☆ Editor's Choice Articles are invited and handled by a select rotating 12 member Editorial Board committee. This paper has been recommended for acceptance by Sinisa Todorovic.

* Corresponding author at: Zhong Guan Cun East Road #95, Haidian District, Beijing, China.
E-mail addresses: cwang@nlpr.ia.ac.cn (C. Wang), kqhuang@nlpr.ia.ac.cn (K. Huang).

[1] Permanent address: Hai Dian Qu Zhong Guan Cun Dong Lu 95 Hao, Beijing, China.

http://dx.doi.org/10.1016/j.imavis.2014.10.013
0262-8856/© 2014 Elsevier B.V. All rights reserved.

- Based on experimental results, the relation between coding and pooling is analyzed from different regimes of classification performance. In different regimes, the combinations of coding and pooling have different influence on the classification performance. Besides, these conclusions and guidelines are validated on various vocabulary construction methods for their strong generalization.
- To use the BoW model conveniently in practical applications, we provide detailed application criterions by selecting the appropriate pairs of coding and pooling methods. These criterions are given based on precision, efficiency and memory requirements, and we summarize these three factors as guidelines for some typical applications.

There are three contributions in this paper:

- *Systematic evaluation*. In this paper, complete combinations of more coding and pooling methods, an extremely large range of vocabulary sizes, primarily typical and large-scale datasets constitute a systematic evaluation. This evaluation compares many coding and pooling methods on primary datasets, and it is convenient to use appropriate methods.
- *Interesting conclusions*. In this paper, we draw some conclusions about coding and pooling. Some of them are different from the previous viewpoints [21–23], while some have never been found before but have shown importance in better applying the BoW model in practice. Particularly, extremely large sizes and large-scale datasets are important to draw these conclusions and improve the generalization ability.
- *Application criterions*. In this paper, based on the conclusions, the detailed application criterions of the BoW model are provided based on precision, efficiency and memory requirements. These criterions can be helpful for researchers and industry community to use appropriate coding and pooling methods in different applications.

The rest of this paper is organized as follows. Section 2 first introduces the related work on coding and pooling. Then, detailed experimental setups are presented in Section 3, and conclusions are drawn in Section 4. Besides, application criterions are provided in Section 5. Finally, Section 6 gives conclusive remarks.

2. Related work

In this section, the related work on coding and pooling is presented. Let $\mathbf{X} = [\mathbf{x_1}, \mathbf{x_2}, \ldots, \mathbf{x_N}] \in \Re^{D \times N}$ be a set of N local features, and $\mathbf{C} = [\mathbf{c_1}, \mathbf{c_2}, \ldots, \mathbf{c_M}] \in \Re^{D \times M}$ be a visual vocabulary with M visual words. For a local feature $\mathbf{x_i}$, the response on $\mathbf{C}$ is $\mathbf{R_i} = [r_1, r_2, \ldots, r_M] \in \Re^{1 \times M}$. For a visual word $\mathbf{c_j}$, the cluster weight and covariance matrix are w_j and $\sigma_\mathbf{j}$ respectively. Besides, λ is a penalty term in sparse coding based methods. Table 1 summarizes some popular coding methods, and some other variables are explained in the footnote below Table 1.

2.1. Coding

In the past decade, many feature encoding methods have been proposed in the literature of image classification. Hard Quantization (HQ) [6] efficiently represents each local feature by the nearest visual word, but it obtains good performance only under large vocabulary sizes [16]. To overcome the limitation, Fisher Kernel (FK) [16] extends HQ by applying a Gaussian mixture model to approximate the distribution of local features, and it shows good results under small sizes. However, assigning continuous local features to discrete visual words causes ambiguity [8]. To model the ambiguity, Soft Quantization (SQ) [8] describes each local feature by applying a Gaussian kernel on the Euclidean distance between the feature and a vocabulary. Recently, Liu et al. have observed the locality of local features in underlying manifolds, so localized SQ [9] is proposed by only considering each feature's neighbor words. However, with average pooling, HQ and SQ may not reconstruct local features precisely, which can be important in feature encoding [10–12,17,18].

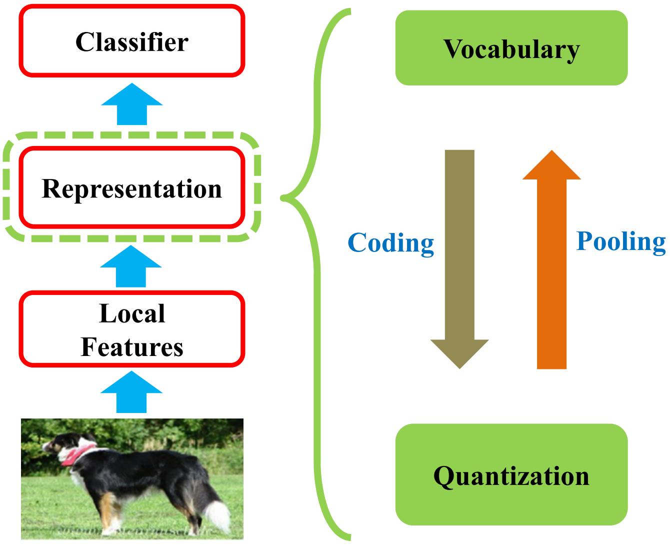

Fig. 1. The framework of the BoW model. Firstly, local features are extracted and clustered to obtain a vocabulary. Then, the features are encoded on the vocabulary to generate feature response. Finally, all the response is pooled over to construct image representation, which is fed into a classifier.

To reconstruct local features precisely, Sparse Coding (SC) [10] represents the features by a visual vocabulary sparsely. Combined with maximum pooling and the spatial pyramid matching (SPM) [1] model, SC can work well with the efficient linear SVM. Recently, empirical studies have shown that the high-dimensional representation constructed by SC can obtain superior performance [14], thus Over-complete Sparse Coding (OSC) [14] enhances the efficiency of SC by softly partitioning the feature space into some sub-manifolds. Based on SC, Yu et al. observe that locality is essential, so Local Coordinate Coding (LCC) [11,28] considers feature encoding in a local manifold, but it has high computational complexity. To implement LCC efficiently, Wang et al. propose the Local-constrained Linear Coding (LLC) [12], which has analytical solution. However, Gao et al. observe that SC, LCC and LLC do not consider the dependence of local features, thus Laplacian Sparse Coding (LSC) [13] enhances the robustness of feature encoding. Recently, some other sparse coding methods [29–36] have also shown good results.

Except for the above-mentioned methods, Huang et al. observe that the sparse coding based methods are saliency oriented, so they propose the Salient Coding (SaC) [15] which has shown competitive performance. To further reduce the reconstruction error of local features, Super Vector Coding (SV) [17] extends HQ to a much higher dimensional feature space, and Improved Fisher Kernel (IFK) [18] enhances FK by power normalization, and both SV and IFK have obtained superior performance [23]. To exploit these high-dimensional methods practically, Hervé Jégou et al. propose the Vector of Locally Aggregated Descriptors (VLAD) [19], which enjoys the high efficiency and low memory requirements jointly by the optimization of dimensionality reduction and an indexing algorithm. Based on VLAD, Picard and Gosselin [37] propose the Vectors of Locally Aggregated Tensors (VLAT) to improve image similarity, and it has shown better performance against VLAD. To analyze these high-dimensional coding methods in a general way, Zhao et al. [38] propose a unified framework to perform coding via vector difference.

Table 1[2] summaries some typical coding methods based on sparsity, locality and efficiency. Sparsity means that only a few words have large

[2] In SQ, β is the Gaussian smoothing factor, which is also used in LLC with the number of nearest words set to be K. In LLC, $\odot$ denotes the element-wise multiplication. In LSC, α penalizes the feature similarity, in which the similarity matrix $\mathbf{S}$ is included. In OSC, p_j is the posterior probability that $\mathbf{x_i}$ is assigned to $\mathbf{c_j}$ in GMM, and L denotes the number of primary clusters. Besides, $\mathbf{C_j}$ is the jth secondary cluster and $\mathbf{R_i^j}$ is the corresponding response. Finally, in SVC, s is a small constant determined by cross-validation.

Table 1
This table summarizes the widely used coding methods with respect to formulation, sparsity, locality and efficiency.

Coding	Formulation	Sparsity	Locality	Efficiency
Hard Quantization (HQ) [6]	$\mathbf{R_i} = \underset{\mathbf{R_i}}{\arg\min} \left\lVert \mathbf{x_i} - \mathbf{C}\mathbf{R_i^T} \right\rVert_2^2 \quad s.t.\ Card(\mathbf{R_i}) = 1,\ \lVert \mathbf{R_i} \rVert_2 = 1,\ \mathbf{R_i} > 0$	✓	✓	$O(M)$
Soft Quantization (SQ) [7–9]	$\mathbf{R_i} = \frac{\exp\left(-\beta \lVert \mathbf{x_i} - \mathbf{c_j} \rVert_2^2\right)}{\sum_{t=1}^{M} \exp\left(-\beta \lVert \mathbf{x_i} - \mathbf{c_t} \rVert_2^2\right),} \quad \forall j = 1, \dots, M$	✓	✓	$O(M)$
Sparse Coding (SC) [10]	$\mathbf{R_i} = \underset{\mathbf{R_i},\mathbf{C}}{\arg\min} \left\lVert \mathbf{x_i} - \mathbf{C}\mathbf{R_i^T} \right\rVert_2^2 + \lambda \lVert \mathbf{R_i} \rVert_1 \quad s.t.\ \lVert \mathbf{c_j} \rVert_2^2 \le 1,\ \forall j = 1, 2, \dots, M$	✓	×	$O(M^2)$
Local Coordinate Coding (LCC) [11]	$\mathbf{R_i} = \underset{\mathbf{R_i},\ \mathbf{C}}{\arg\min} \left\lVert \mathbf{x_i} - \mathbf{C}\mathbf{R_i^T} \right\rVert_2^2 + \lambda \sum_{j=1}^{M} \left\lvert (\mathbf{R_i})_j \right\rvert \lVert \mathbf{x_i} - \mathbf{c_j} \rVert_2^2 \quad s.t.\ \lVert \mathbf{c_j} \rVert_2^2 \le 1,\ \forall j = 1, 2, \dots, M$	✓	✓	$O(M^2)$
Local-constrained Linear Coding (LLC) [12]	$\mathbf{R_i} = \underset{\mathbf{R_i},\ \mathbf{C}}{\arg\min} \left\lVert \mathbf{x_i} - \mathbf{C}\mathbf{R_i^T} \right\rVert_2^2 + \lambda \lVert \mathbf{d_i} \odot \mathbf{R_i} \rVert_2^2 \quad \mathbf{d_i} = exp\left[-\beta\left(\lVert \mathbf{x_i} - \mathbf{c_1} \rVert_2^2, \dots, \lVert \mathbf{x_i} - \mathbf{c_M} \rVert_2^2\right)\right] \quad s.t.\ \lVert \mathbf{R_i} \rVert_1 = 1$	✓	✓	$O(M + K^2)$
Laplacian Sparse Coding (LSC) [13]	$\mathbf{R_i} = \underset{\mathbf{R_i},\ \mathbf{C}}{\arg\min} \left\lVert \mathbf{x_i} - \mathbf{C}\mathbf{R_i^T} \right\rVert_2^2 + \lambda \lVert \mathbf{R_i} \rVert_1 + \alpha/2 \sum_{ij} \lVert \mathbf{R_i} - \mathbf{R_j} \rVert_2^2 \mathbf{S_{ij}} \quad s.t.\ \lVert \mathbf{c_j} \rVert_2^2 \le 1,\ \forall j = 1, 2, \dots, M$	✓	×	$O(M^2)$
Over-complete Sparse Coding (OSC) [14]	$\mathbf{R_i} = p_j(\mathbf{x_i}) \mathbf{R_i}^{\mathbf{j}} / \sum_{t=1}^{M} p_t(\mathbf{x_i}) \quad \mathbf{R_i}^{\mathbf{j}} = \underset{\mathbf{R_i}^{\mathbf{j}},\ \mathbf{C_j}}{\arg\min} \left\lVert \mathbf{x_i} - \mathbf{C_j} \left(\mathbf{R_i}^{\mathbf{j}}\right)^T \right\rVert_2^2 + \lambda \left\lVert \mathbf{R_i}^{\mathbf{j}} \right\rVert_1 \quad s.t.\ \lVert \mathbf{c_j} \rVert_2^2 \le 1,\ \forall j = 1, 2, \dots, M$	✓	×	$O(L \times M^2)$
Saliency Coding (SaC) [15]	$\mathbf{R_i} = \Phi\left(\frac{\lVert \mathbf{x_i} - \mathbf{c_j} \rVert_2^2}{\frac{1}{K-1} \sum_{t \ne j}^{K} \lVert \mathbf{x_i} - \mathbf{c_t} \rVert_2^2}\right)$ $s.t.\ \mathbf{c_j} = \underset{\mathbf{c_t}}{\arg\min} \lVert \mathbf{x_i} - \mathbf{c_t} \rVert_2^2$	✓	✓	$O(M)$
Super-vector Coding (SV) [17]	$\mathbf{R_i} = [s; \mathbf{x_i} - \mathbf{c_j}]$ $s.t.\ \mathbf{c_j} = \underset{\mathbf{c_t}}{\arg\min} \lVert \mathbf{x_i} - \mathbf{c_t} \rVert_2^2$	✓	✓	$O(M)$
Improved Fisher Kernel (IFK) [16,18]	$\mathbf{R_i} = \gamma(j) \left[\frac{\mathbf{x_i} - \mathbf{c_j}}{\sigma_{\mathbf{j}}}, \frac{(\mathbf{x_i} - \mathbf{c_j})^2}{\sigma_{\mathbf{j}}^2} - 1\right]$ $\gamma(j) = \frac{w_j p_j(\mathbf{x_i})}{\sum_{t=1}^{M} w_t p_t(\mathbf{x_i}),} \forall j = 1, \dots, M$	✓	✓	$O(M)$

responses ($\mathbf{R_i}$), while locality indicates that the words far away from a local feature should be given weak responses. The above methods partially share the sparsity and locality. In detail, all the coding methods satisfy sparsity, which embodies the importance of sparse representation. However, most sparse coding based methods are not local and have the computational complexity of $O(M^2)$, while most local methods are much more efficient with the complexity of $O(M)$.

2.2. Pooling

The idea of feature pooling [22] originates in Hubel and Wiesel's seminal work on complex cells in the visual cortex [39], and is related to Koenderink's concept of locally orderless images [40]. Pooling is an operation which can transform feature responses into new and more usable image representation that preserves important information [21,22]. Let $\mathbf{U}$ be a matrix defined as

$$\mathbf{U} = \left[\mathbf{R_1^T}, \mathbf{R}_2^{\mathbf{T}}, \dots, \mathbf{R_N^T}\right]^{\mathbf{T}} = [\mathbf{u_1}, \mathbf{u_2}, \dots, \mathbf{u_M}], \tag{1}$$

wherein $\mathbf{u_i}$ contains the responses of all local features on the visual word $\mathbf{c_i}$. After the pooling step, the image representation $\mathbf{Z}$ is constructed. In the literature, two classic pooling methods have been widely used as follows.

Average pooling is widely accepted in convolutional networks [41], some models of the primary visual cortex [42] and the BoW model [6]. As shown in Eq. (2), it computes the average response of all local features on a vocabulary, and preserves frequency information [8].

$$\mathbf{Z} = \frac{1}{N}\left[\lVert \mathbf{u_1} \rVert_1, \lVert \mathbf{u_2} \rVert_1, \dots, \lVert \mathbf{u_M} \rVert_1\right]. \tag{2}$$

Maximum pooling is popular in convolutional networks [41], biological inspired model [43] and the BoW model [6]. It is proposed for its robustness to local spatial transformation and noise [10]. As shown in Eq. (3), it keeps the maximum response of all local features on a vocabulary, and preserves reconstruction information [10,15].

$$\mathbf{Z} = \left[\lVert \mathbf{u_1} \rVert_\infty, \lVert \mathbf{u_2} \rVert_\infty, \dots, \lVert \mathbf{u_M} \rVert_\infty\right]. \tag{3}$$

Besides, researchers observe that the frequency [8,21] and reconstruction [10,15] information can be complementary to each other, thus Zhou et al. propose the weighted average pooling [17]. Recently, Boureau et al. [20] have observed that pooling should be local in feature space, thus a multi-way local pooling strategy is proposed to leverage locality based on average or maximum pooling.

2.3. Relation

Recently, some researchers have explored the relation between coding and pooling in the BoW model [21–23]. With several classic coding methods, Boureau et al. [21,22] analyze theoretically how coding and pooling are related based on sample cardinality (the number of local features) under small vocabulary sizes; while Chatfield et al. [23] and Huang et al. [24] evaluate typical coding methods under relatively larger vocabulary sizes, but without considering different pooling schemes. Their main conclusions are listed as follows:

- The best performance of Sparse Coding (SC) improves over the ones of Hard Quantization (HQ) and Soft Quantization (SQ) [21].
- The best performance of maximum pooling is superior to the one of average pooling [21].
- Under maximum pooling, larger vocabularies lead to higher performance [23].

The first two are obtained under the maximum vocabulary size of 1024. In our experiments, we will extend these two conclusions by using a large range of vocabulary size (up to 1*M*). Finally, the third one is obtained under the maximum size of 25*k*. However, in this paper, it is observed that maximum pooling will remain stable or decrease under larger sizes, e.g., 260*k*. The main conclusions in this paper will be given later in Section 4.

3. Experimental setup

Before giving our conclusions in Section 4, in this section, detailed experimental setups are first introduced. Specifically, based on an extremely large range of vocabulary sizes, we use the complete combinations of more popular coding methods and two popular pooling methods under five typical and large-scale datasets for evaluation.

3.1. Datasets and evaluation

Five popular datasets are used in this paper. Three of them are the typical small-scale datasets, including 15 Scenes [1], Caltech 101 [2] and PASCAL VOC 2007 [44]; while another two are the large-scales ones, which include Caltech 256 [3] and ImageNet [5].

15 Scenes and Caltech 101 include 15 scene and 101 object categories. For each category, 100 images are used for training and the rest for testing on 15 Scenes, while 30 images for training and at most 50 for testing on Caltech 101. Experiments are implemented over 10 random splits of the data, and the mean accuracy and standard deviation are reported. PASCAL VOC 2007 has 20 object categories, including 5011 training and validation images, and 4952 testing images. The mean average precision is reported.

Caltech 256 and ImageNet include 256 and 1000 object categories. On Caltech 256, 60 images are used for training and the rest for testing, and the evaluation is the same to Caltech 101. ImageNet consists of 1.2 million training images, and 150k validation and testing images. The error rate based on the first confident label is reported.

3.2. Local feature

We use SIFT local features which are generated by the VLFeat toolbox [45]. They are extracted from local patches densely located by every 4 pixels on an image under three feature scales, defined by setting the width of the SIFT spatial bins to 4, 6 and 8 pixels. For other options, based on the average of gradient magnitude, the low contrast SIFT features are detected and dropped when the magnitude is below a certain threshold. Besides, the '*fast*' option is selected for a faster extraction algorithm [23].

3.3. Vocabulary

The vocabulary sizes are set to be $[2^4, 2^5, \ldots 2^{19}, 2^{20}]$, seventeen scales in total. Particularly, for SV and IFK, the sizes are set to be [16,32, 64,128,256] because of large memory requirement. The vocabularies are constructed by the standard k-means, in which 40 million random local features are used to guarantee the effectiveness of the clustering. In order to accelerate k-means on large vocabularies, the two-layer hierarchical k-means is adopted and the number of nodes for each layer is set in Table 2. Besides, using the k-d tree random forest Approximate Nearest Neighbor (ANN) to speed up is also available, while we will show in Section 4.3 that the hierarchical k-means and ANN are comparable. Furthermore, k-means is the initialization of the GMM clustering, in which the diagonal covariance matrix is used. To make fair comparison, for the methods incorporating vocabulary learning [12,14], the learning process is not adopted.

Table 2
The setup of visual vocabularies in hierarchical k-means.

Size	8*k*	16*k*	32*k*	65*k*	131*k*	262*k*	524*k*	1*M*
Layer 1	32	64	64	64	128	128	256	256
Layer 2	256	256	512	1024	1024	2048	2048	4096

3.4. Coding and pooling

Table 3 shows the detailed settings of coding and pooling methods for evaluation, and it includes the coding methods, the number of nearest words (K), the vocabulary construction, the maximum vocabulary size, and the pooling methods. For HQ and SQ, the source code released by Chatfield et al. [23] is used; while for LLC, the source code released by Wang et al. [12] is used. For other coding methods, they are implemented by ourself and a combined *MATLAB*/*C*++ implementation is used. Especially, there are some points to be noted as follows:

- OSC and LLC are used instead of SC and LCC. OSC is an accelerated version of SC, while LLC is an efficient implementation of LCC.
- For the number of nearest words (K) in coding, it is set to be 5 in SQ, LLC and SaC for the best performance [9,12,15], and the same setting is used for IFK [16,18,23]. In OSC, two primary clusters are used practically in the first layer [14].
- In IFK, power normalization [18] is used before L_2 normalization in constructing representation ($\mathbf{Z}$); while for others, only L_2 normalization is used.

3.5. Image partition

For the three typical small-scale datasets, the spatial pyramid matching (SPM) [1] is used under the partition of 1 × 1, 2 × 2 and 3 × 1; while for the two large-scale ones, we do not use the SPM partition. Particularly, the partition of 4 × 4 at the third level for typical datasets is not adopted because the dimensionality of image representation will be larger than 21 million at the vocabulary size of 1*M*, which can become extremely challenging for training, and this is the same reason we do not use the SPM partition in large-scale datasets. However, it is shown that we can achieve comparable or better precision, and the same baseline can guarantee the correctness of the experimental results and the reliability of our conclusions.

3.6. Training

In the training phase, liblinear, libsvm and averaging stochastic gradient descent (ASGD) [46–49] are used, and the corresponding penalty terms are determined by a five-fold cross validation. Except for ImageNet, liblinear is used for small and medium vocabulary sizes because it is much faster in this case, and the libsvm with the ' *precomputed_kernel* ' option is adopted to reduce the memory cost in the training of extremely large vocabularies. Besides, we adopt the ASGD algorithm in ImageNet to deal with large-scale training problems. All the classifiers are trained by the rule of one-vs-all, and only linear SVM is adopted for its high efficiency and low memory requirements.

3.7. Comparison

All the coding methods are evaluated based on the equal dimensionality of the image representation ($\mathbf{Z}$) for fair comparison. For the top five

Table 3
The detailed settings of coding and pooling methods.

Coding	*K*	Vocabulary	Maximum size	Pooling
HQ	1	K-means	256 × 4096	AP, MP
SQ	5	K-means	256 × 4096	AP, MP
LLC	5	K-means	256 × 4096	AP, MP
OSC	–	K-means + GMM	256 × 4096	AP, MP
SaC	5	K-means	256 × 4096	AP, MP
SV	1	K-means	256	AP
IFK	5	K-means + GMM	256	AP

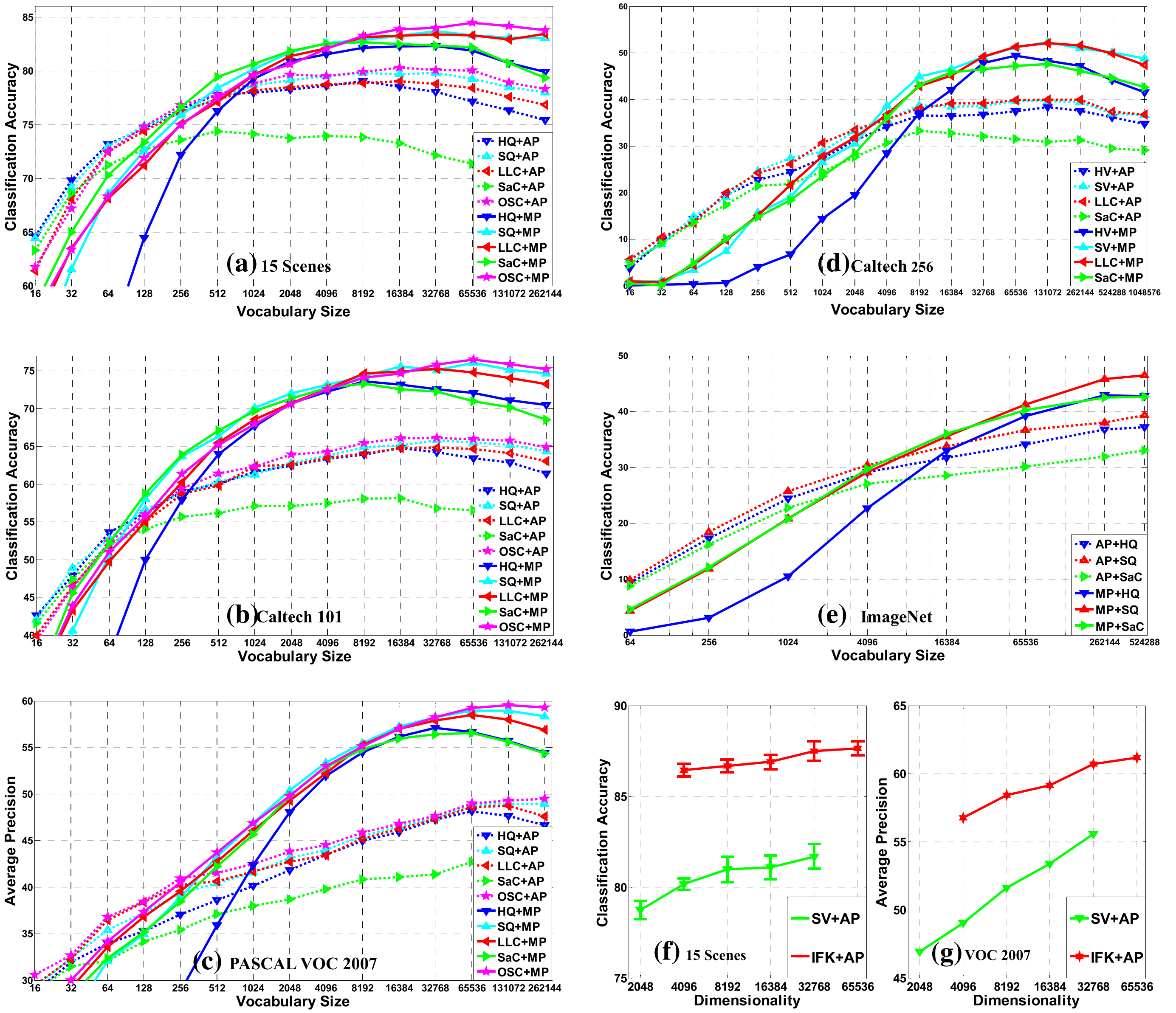

Fig. 2. The classification precision of the combinations of coding and pooling on the five datasets: (a) 15 Scenes; (b) Caltech 101; (c) PASCAL VOC 2007; (d) Caltech 256; (e) ImageNet; (f) SV and IFK on 15 Scenes; (g) SV and IFK on PASCAL VOC 2007. Best viewed in color.

coding methods in Table 3, the dimensionality is M, wherein M denotes the vocabulary size. For SV and IFK, according to their implementations [17,18], Principal Component Analysis (PCA) is used to reduce the dimensionality of SIFT features from 128 to 80. Therefore, the dimensionalities for SV and IFK are actually 80 and 160 × [16,32,64,128,256] respectively.

4. Interesting conclusions

In this section, based on the experimental results of coding and pooling methods, we provide deep understanding to the intrinsic nature of the BoW model. In the following experiments, we obtain three conclusions:

- Combined with average pooling, most coding methods are comparable.
- The best performance of max pooling is better than the one of average pooling.
- A saturant point exists in maximum pooling.

In this section, we will elaborate these conclusions based on average and maximum pooling. The first one about average pooling is explained in Section 4.1, while the other two about maximum pooling will be analyzed in Section 4.2. Besides, to guarantee the generalization of these conclusions, the effect of vocabulary construction will be evaluated in Section 4.3. Finally, Section 4.4 summarizes the main results and conclusions.

4.1. Average pooling

In this subsection, the relation between average pooling and coding is studied. Fig. 2 shows the precision of average pooling on the five datasets. It can be observed that most coding methods are comparable. With the increasing vocabulary size, the precision gradually achieves the highest, e.g., 8192 on 15 Scenes and 16,384 on Caltech 101, while it begins to decrease after the highest value. Besides, it can be observed in Fig. 2(f) and (g) that SV and IFK perform better than other coding methods, and IFK is superior to SV.

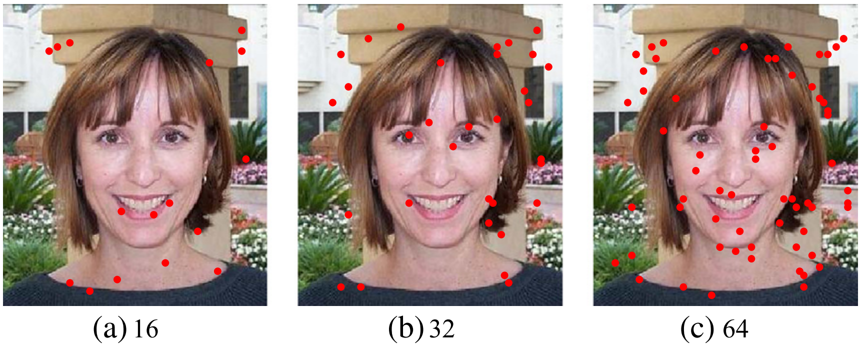

Fig. 3. An illustration of the reconstructed local features by maximum pooling. The red dots in (a), (b) and (c) denote these reconstructed features under the vocabulary sizes of 16, 32 and 64 respectively. The image is from the Face-easy category on the Caltech 101 database.

Assume each visual word is a histogram bin, then the average pooling defined in Eq. (2) has the same form to the probability density estimation (PDE) problem [50]. Based on the PDE theory, it can be concluded that average pooling describes the PDE of all local features on a vocabulary. For different coding methods, they model the same PDE problem under average pooling, thus their precision is comparable. Therefore, the first conclusion is that combined with average pooling, most coding methods are comparable.

One exception here is the Salient Coding (SaC), which performs worse than other coding methods. Assume we only consider the 2 nearest visual words (c_1,c_2) of a local feature (x) in feature encoding, and the response (r_1) on the nearest word (c_1) will be $r_1 = 1 - (d_1/d_2)^2$, in which $d_i = ||x - c_i||^2$. If $d_1 \ll d_2$, r_1 is similar to the response of Hard Quantization; while if d_1 and d_2 are comparable, r_1 should be around 0.5, but it becomes zero in this case. As the vocabulary size increases, visual words are denser in the feature space and the distance among visual words become much smaller. Therefore, the response of most features will be zero and the PDE problem will fail for SaC.

4.2. Maximum pooling

In this subsection, the relation between maximum pooling and coding is studied. Based on the observation in Fig. 2, the relation is elaborated from two regimes of classification precision in Section 4.2—1 and —2: *Increase* and *saturation*, which correspond to our last two conclusions.

1) *Increase*: It can be observed from Fig. 2(a) to (e) that the precision of maximum pooling improves largely at the beginning, e.g., the vocabulary size of 16 to 65,536 on PASCAL VOC 2007 and 64 to 524,288 on ImageNet.
According to the definition of maximum pooling in Eq. (3), it preserves the maximum response of all local features, and in turn the maximum response must belong to some features, which are called the reconstructed features. Therefore, it can be inferred that maximum pooling can reconstruct local features, and Fig. 3 illustrates the feature reconstruction under the sizes of 16, 32 and 64. It can be observed that as the size increases, more features are reconstructed and focus on the face, which demonstrates that these reconstructed local features are discriminative to objects. However, under very small vocabulary size such as 16 and 32, the performance of average pooling is higher because only a small number of discriminative features can be reconstructed, which are not discriminative enough to represent the object. With more discriminative features discovered, the objects can be represented more discriminatively, and the performance will exceed the one of average pooling because many noisy features will affect the probability density estimation in average pooling. Therefore, the best performance of maximum pooling is better than the one of average pooling.
2) *Saturation*: It can be observed from Fig. 2(a) to (e) that the precision of maximum pooling achieves saturant, e.g., 32,768 on 15 Scenes, 131*k* on Caltech 256. However, the precision decreases after the saturant point, which implies that a saturant point exists in maximum pooling.

Fig. 5 shows the intra-class variance of the image representation on the Face-easy category. The variance is defined as $V = \sum_{i=1}^{Q}\sum_{j=1}^{Q}(||\mathbf{Z_i} - \mathbf{Z_j}||_2^2)/Q^2$, wherein $\mathbf{Z_i}$ represents the image representation of the *i*th image, and *Q* is the number of images in this category. According to Fig. 2(b), the precision of HQ and SaC decreases at the size of 8192, which corresponds to the variance of 1.13 in Fig. 5. Meanwhile, 1.13 is almost the variance for SQ, LLC and OSC at the size of 65,536, at which their precision begins to decrease in Fig. 2(b). Based on these observations, we hypothesize that there may be a strong relation between vocabulary size and intra-class variance of image representation. As the vocabulary size increases to extremely large, the increasing intra-class variance will largely differentiate the representation of similar images, and this over-fitting probably causes the saturation of maximum pooling. Therefore, a saturant point exists in maximum pooling (see Fig. 4).

4.3. Vocabulary generalization

Vocabulary construction is important in image representation. In the above experiments, we have used the standard and the two-layer hierarchical k-means clustering. In order to generalize these conclusions to vocabulary construction, in this subsection, we validate the conclusions on different construction methods. In recent years, two primary clustering methods have been widely used in the BoW model: k-means and

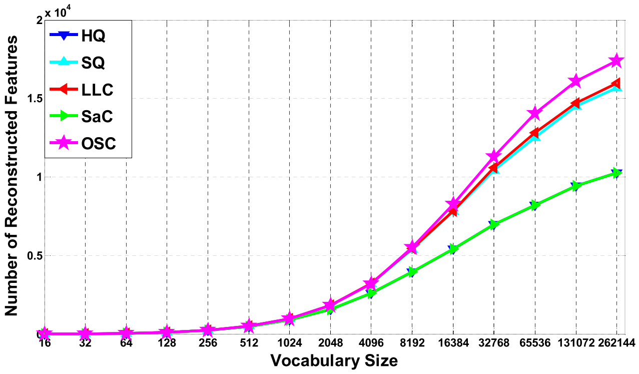

Fig. 4. An illustration of the number of reconstructed local features for different coding methods under an extremely large range of vocabulary sizes. The results are obtained based on all the images from the Face-easy category on the Caltech 101 database.

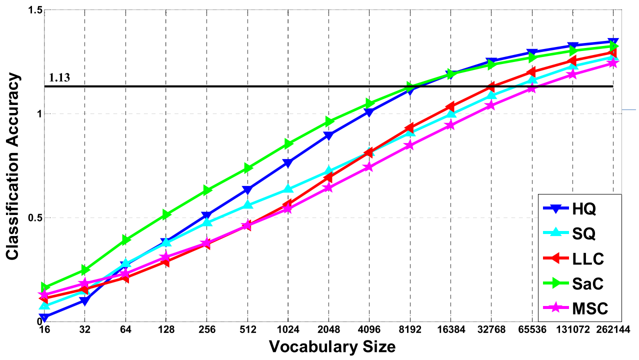

Fig. 5. An illustration of the intra-class variance for different coding methods under a large range of vocabulary sizes. The results are obtained based on all the images from the Face-easy category on the Caltech 101 database.

GMM clustering, thus we will evaluate these two methods. For k-means, we use three typical variants: *k-d ANN* for the k-d tree random forest ANN k-means [51,52], *k-medoids* [53] and *fuzzy-kmeans* [54]; while for GMM, similar to k-means, three typical variants are considered: *gmm-full*, *gmm-diag* and *gmm-cons*, which denote the GMMs with the full, diagonal and constant covariance matrix respectively [14,18].

Fig. 6 shows the classification precision of four coding and two pooling methods on the 15 Scenes dataset under the above six vocabulary construction methods. Specifically, compared to the hierarchical k-means in Fig. 2(a), it can be observed that *k-d ANN* is comparable in speeding up k-means on large vocabularies. Besides, the experiments also show that the conclusions can be easily observed in the variants of both k-means and GMMs, which can demonstrate that the conclusions do not depend on specific vocabulary construction methods. Therefore, these conclusions have strong generalization to vocabulary construction.

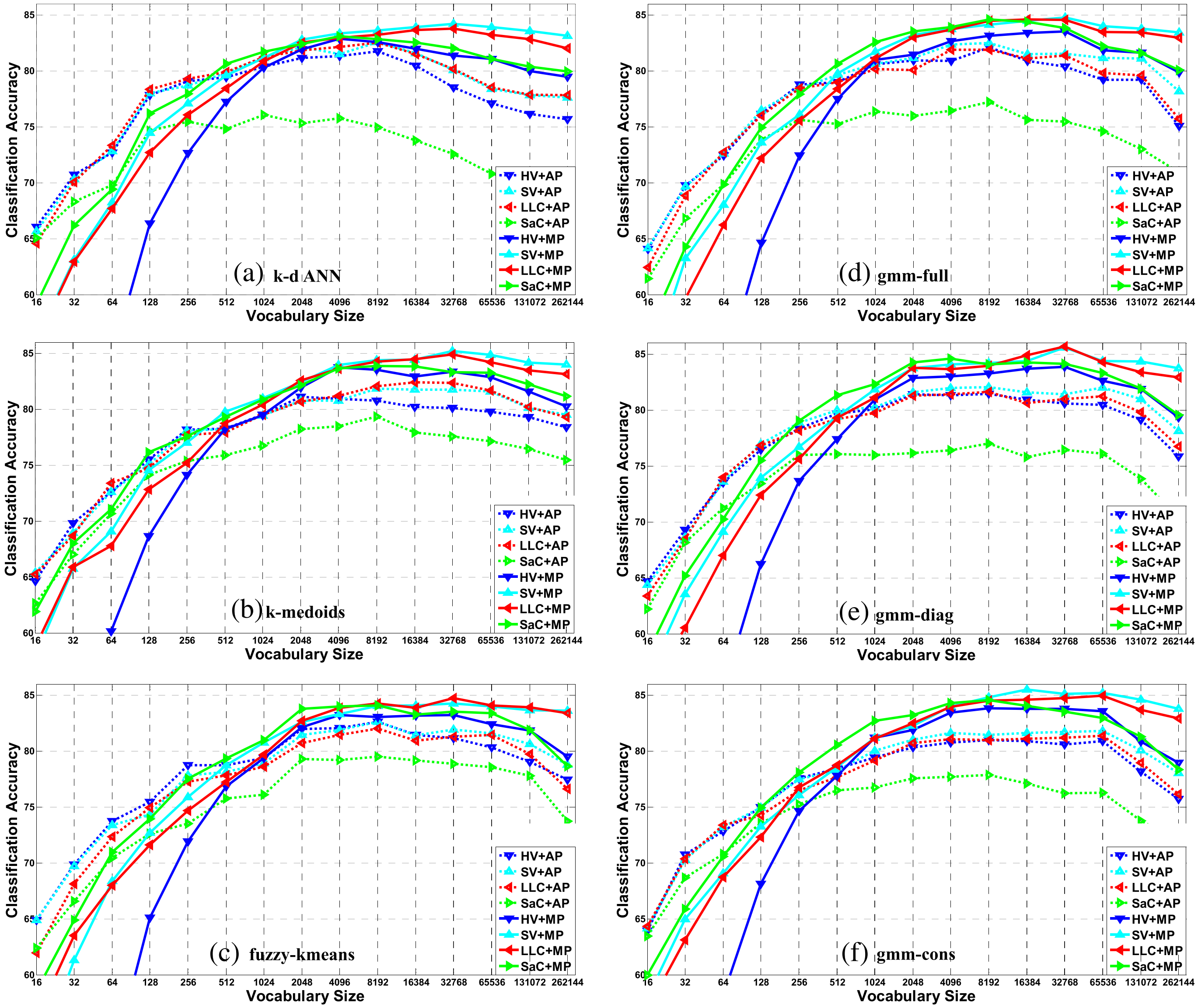

Fig. 6. The evaluation of six vocabulary construction methods on the 15 Scenes dataset: (a) *k-d ANN*; (b) *k-medoids*; (c) *fuzzy-kmeans*; (d) *gmm-full*; (e) *gmm-diag*; (f) *gmm-cons*. Best viewed in color.

4.4. Summary

In this section, we have empirically explored the relation between coding and pooling under an extremely large range of vocabulary sizes on five primary datasets. Based on the experimental results, we have drawn some conclusions, which are summarized as follows:

1) *Average pooling*: Combined with average pooling, most coding methods are comparable. The combinations of average pooling and different coding methods model the same probability density estimation (PDE) problem, which will result in comparable precision.
2) *Maximum pooling — increase*: The best performance of max pooling is better than the one of average pooling. With the increasing vocabulary size, maximum pooling reconstructs more discriminative local features, which are discriminative to represent objects, while average pooling will be affected by many noisy local features.
3) *Maximum pooling — saturation*: A saturant point exists in maximum pooling. As the vocabulary size increases to extremely large, the intra-class variance of image representation will become much larger, thus over-fitting will happen and saturate maximum pooling.

Besides, in order to validate the generalization of these conclusions, we further evaluate them on different vocabulary construction methods, in which all the conclusions are clearly observed. Therefore, these conclusions have strong generalization to vocabulary construction in BoW.

5. Application criterions

With the increasing number of coding and pooling methods have been proposed recently, how to apply them in specific applications has become an important problem. In this section, we select the appropriate pairs of coding and pooling methods in practical applications. Based on the experimental results of the above section, we observe that our conclusions can be understood from three regimes:

- *Regime 1*: maximum pooling is below average pooling.
- *Regime 2*: maximum pooling reaches the saturant point.
- *Regime 3*: maximum pooling begins to decrease.

Based on these regimes, we give the detailed application criterions of the BoW model. Firstly, the performance analysis is given based on accuracy, efficiency and memory requirements, which correspond to Sections 5.1 to 5.3 respectively. Then, in Section 5.4, a summary is given for researchers and industry community to use the BoW model conveniently in different applications.

5.1. Precision

Table 4 summarizes the appropriate pairs of coding and pooling methods based on classification precision, and the conclusion is given based on *regime 1* and *regime 2*. The reason we do not include *regime 3* is that the precision begins to decrease in this regime, thus it is not necessary to consider it in practical applications. Besides, for SV and IFK, their feature dimensionality ranges from *regime 1* to *regime 2* in Fig. 2, thus SV and IFK are also considered in these three regimes. Based on the conclusions in Section 4, the appropriate pairs of coding and pooling can be easily selected at *regime 1*. However, in *regime 2*, only SQ, LLC and MSC are recommended for their more stable precision. For SV and IFK, no matter in which regime, they are always appropriate to combine with average pooling.

Table 4
Application criterions based on classification precision.

Regime	*Regime 1*		*Regime 2*	
Coding	AP	MP	AP	MP
HQ	✓	–	–	✓
SQ	✓	–	–	✓
LLC	✓	–	–	✓
SaC	–	–	–	✓
OSC	✓	–	–	✓
SV	✓	–	✓	–
IFK	✓	–	✓	–

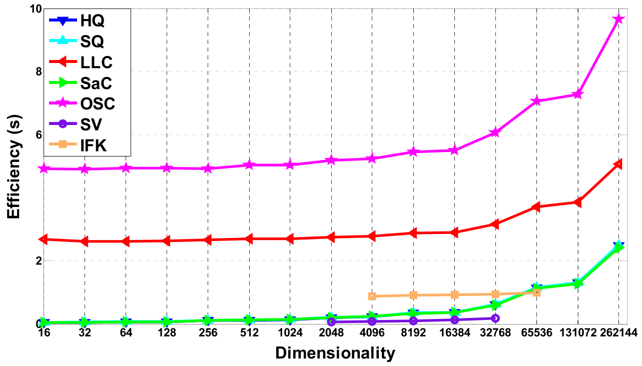

Fig. 7. The time cost(second) of different coding methods under an extremely large range of feature dimensionality. The figure is obtained by processing a fixed image on the PASCAL VOC 2007 database. Best viewed in color.

5.2. Efficiency

Fig. 7 shows the time cost of different coding methods under an extremely large range of feature dimensionality, and the computational time (s) includes the distance computation and feature encoding. Particularly, we only use average pooling for evaluation. Among these methods, it is observed that HQ, SQ, SaC, SV and IFK are much more efficient than others. Furthermore, under most feature dimensionality, these methods cost less than 1s, e.g., SV and SQ only cost 0.17*s* and 0.62*s* respectively at the dimensionality of 32*k*, which satisfies the high efficiency and high precision jointly for practical applications. However, for LLC and OSC, they are time-consuming because they have to solve the iterative optimization problem in Sparse Coding.

5.3. Memory

Fig. 8 shows the memory requirement (*KB*) of the image representation (**Z**) for different coding methods under an extremely large range of feature dimensionality. Due to the little difference between average and

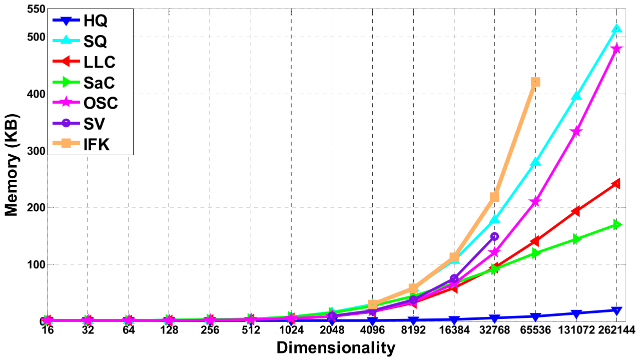

Fig. 8. The memory requirement (KB) of the image representation (Z) for different coding methods under an extremely large range of feature dimensionality. The figure is drawn by considering a fixed image on the PASCAL VOC 2007 database. Best viewed in color.

Table 5
Application criterions of the appropriate pairs of coding and pooling methods in different applications.

Application	*Regime 1*								*Regime 2*						
		HQ	SQ	LLC	SaC	OSC	SV	IFK		HQ	SQ	LLC	SaC	OSC	SV	IFK
High precision	AP	✓	✓	✓	–	✓	✓	✓	AP	–	–	–	–	–	✓	✓
	MP	–	–	–	–	–	–	–	MP	–	✓	✓	–	✓	–	–
High efficiency	AP	✓	✓	–	✓	–	✓	✓	AP	–	–	–	–	–	✓	✓
	MP	–	–	–	–	–	–	–	MP	✓	✓	–	✓	–	–	–
Low memory	AP	✓	✓	✓	✓	✓	✓	✓	AP	–	–	–	–	–	–	–
	MP	–	–	–	–	–	–	–	MP	✓	–	–	–	–	–	–

maximum pooling in memory requirement, we only use average pooling for evaluation. The final image representation is saved as *mat* file in double style without L_2 normalization. Among these methods, HQ is the most memory-saving one and it only occupies 20*KB* even at the feature dimensionality of 262*k*, which has shown its potential in large scale image classification. Meanwhile, LLC and SaC are also acceptable for their intermediate memory requirement, while SQ, OSC and IFK occupy much larger memory at large dimensionality.

5.4. Applications

Finally, based on the above evaluation of precision, efficiency and memory requirements, Table 5 gives the detailed application criterions of coding and pooling methods. We hope that these criterions can be helpful for researchers and industry community to select appropriate methods based on their own application demands.

5.4.1. High precision

In this case, the hardware is powerful enough to run *regime 2* regardless of efficiency and memory limitations, e.g., computer servers or a cluster of computers. Based on Fig. 2 and Table 4, it can be concluded that: 1) at *regime 1*, all coding methods except SaC are appropriate with average pooling; 2) at *regime 2*, SV and IFK with average pooling while SQ, LLC and MSC with maximum pooling are better.

5.4.2. High efficiency

In this case, the efficiency requirement is very high and it is expected to see the classification results as soon as possible, e.g., the image searching engine or the image indexing system. Based on Fig. 7 and Table 4, it can be concluded that: 1) at *regime 1*, except for LLC and MSC, all others are appropriate to combine with average pooling; 2) at *regime 2*, SV and IFK with average pooling while HQ, SQ, and SaC with maximum pooling are recommended.

5.4.3. Low memory

In this case, it is expected that the whole classification task occupies memory as small as possible, e.g., common desktops or embedded systems. Based on Fig. 8 and Table 4, it can be concluded that: 1) at *regime 1*, all coding methods are appropriate to combine with average pooling; 2) at *regime 2*, HQ with maximum pooling is available.

6. Conclusion

In this paper, we use the BoW model better in image classification by empirically studying the relation of coding and pooling methods. Specifically, we consider more typical coding methods, an extremely large range of vocabulary sizes and the primarily typical and large-scale datasets. Based on the experimental results, we have drawn three conclusions. 1) Combined with average pooling, most coding methods are comparable; 2) the best performance of maximum pooling is better than the one of average pooling; 3) a convergent point exists in maximum pooling. Besides, for the strong generalization of these conclusions, we have validated them on some widely used vocabulary construction methods. Finally, in order to use BoW in practice, detailed application criterions have been provided to select the appropriate pairs of coding and pooling methods. We hope that the evaluation provides exhaustive baselines to use appropriate coding and pooling in different datasets, the conclusions provide valuable understanding to the intrinsic nature of BoW, and the application criterions provide useful application guidelines for researchers and industry community to use the BoW model in different practical applications.

Acknowledgments

This work is supported by the National Basic Research Program of China (Grant No. 2012CB316302) and the National Natural Science Foundation of China (Grant No. 61322209, Grant No. 61175007, and Grant No. 2012BAH07B01).

References

[1] S. Lazebnik, C. Schmid, J. Ponce, Beyond bags of features: spatial pyramid matching for recognizing natural scene categories, Computer Vision and, Pattern Recognition 2006, pp. 2169–2178.
[2] L. Fei-Fei, R. Fergus, P. Perona, Learning generative visual models from few training examples: an incremental Bayesian approach tested on 101 object categories, Computer Vision and Pattern Recognition Workshop on Generative-Model Based Vision, 2004.
[3] G. Gregory, H. Alex, P. Pietro, Caltech-256 object category dataset, Calif. Inst. Technol. 54 (7694) (2007) 1–20.
[4] M. Everingham, L.V. Gool, C.K.I. Williams, J. Winn, A. Zisserman, The pascal visual object classes (voc) challenge, Int. J. Comput. Vis. (2010) 303–338.
[5] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, L. Fei-Fei, Learning generative visual models from few training examples: an incremental bayesian approach tested on 101 object categories, Computer Vision and, Pattern Recognition 2007, pp. 248–255.
[6] G. Csurka, C.R. Dance, L. Fan, J. Willamowski, C. Bray, Visual categorization with bags of keypoints, European Conference on Computer Vision International Workshop on Statistical Learning in Computer Vision, 2004.
[7] J.C. van Gemert, J.-M. Geusebroek, C.J. Veenman, A.W.M. Smeulders, Kernel codebooks for scene categorization, European Conference on Computer Vision 2008, pp. 696–709.
[8] J.C. van Gemert, C. Veenman, A.W.M. Smeulders, J. Geusebroek, Visual word ambiguity, IEEE Trans. Pattern Anal. Mach. Intell. 32 (7) (2010) 1271–1283.
[9] L. Liu, L. Wang, X. Liu, In defense of soft-assignment coding, International Conference on Computer Vision, 2011.
[10] J. Yang, K. Yu, Y. Gong, T. Huang, Linear spatial pyramid matching using sparse coding for image classification, Computer Vision and Pattern Recognition 2009, pp. 1794–1801.
[11] K. Yu, T. Zhang, Y. Gong, Nonlinear Learning Using Local Coordinate Coding, Neural Information Processing Systems, vol. 22, 2009.
[12] J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, Y. Gong, Locality-constrained linear coding for image classification, Computer Vision and, Pattern Recognition 2010, pp. 3360–3367.
[13] S. Gao, I.W.-H. Tsang, L.-T. Chia, P. Zhao, Local features are not lonely — Laplacian sparse coding for image classification, Computer Vision and, Pattern Recognition 2010, pp. 3555–3561.
[14] J. Yang, K. Yu, T. Huang, Efficient highly over-complete sparse coding using a mixture model, European Conference on Computer Vision 2010, pp. 1–14.
[15] Y. Huang, K. Huang, T. Tan, Salient coding for image classification, Computer Vision and, Pattern Recognition 2011, pp. 1691–1753.
[16] F. Perronnin, C. Dance, Fisher kernels on visual vocabularies for image categorization, Computer Vision and, Pattern Recognition 2007, pp. 1–8.
[17] X. Zhou, K. Yu, T. Zhang, T.S. Huang, Image classification using super-vector coding of local image descriptors, European Conference on Computer Vision 2010, pp. 141–154.
[18] F. Perronnin, J. Sánchez, T. Mensink, Improving the fisher kernel for large-scale image classification, European Conference on Computer Vision 2010, pp. 143–156.

[19] H. Jégou, F. Perronnin, M. Douze, J. Sánchez, P. Pérez, C. Schmid, Aggregating local image descriptors into compact codes, IEEE Trans. Pattern Anal. Mach. Intell. 34 (9) (2011) 1704–1716.

[20] Y.-L. Boureau, N.L. Roux, F. Bach, J. Ponce, Y. LeCun, Ask the locals: multi-way local pooling for image recognition, International Conference on Computer Vision 2011, pp. 2651–2658.

[21] Y.-L. Boureau, F. Bach, Y. LeCun, J. Ponce, Learning mid-level features for recognition, Computer Vision and, Pattern Recognition 2010, pp. 2559–2566.

[22] Y.-L. Boureau, J. Ponce, Y. LeCun, A theoretical analysis of feature pooling in visual recognition, International Conference on, Machine Learning 2010, pp. 111–118.

[23] K. Chatfield, V. Lempitsky, A. Vedaldi, A. Zisserman, The devil is in the details: an evaluation of recent feature encoding methods, British Machine Vision Conference, 2011.

[24] Y. Huang, Z. Wu, L. Wang, T. Tan, Feature coding in image classification: a comprehensive study, IEEE Trans. Pattern Anal. Mach. Intell. 36 (3) (2013) 493–506.

[25] J. Sánchez, F. Perronnin, High-dimensional signature compression for large-scale image classification, Computer Vision and, Pattern Recognition 2011, pp. 1665–1672.

[26] G.E. Hinton, R.R. Salakhutdinov, Reducing the dimensionality of data with neural networks, Science 313 (5786) (2006) 504–507.

[27] A. Krizhevsky, I. Sutskever, G.E. Hinton, ImageNet classification with deep convolutional neural networks, Advances in Neural Information Processing, 2012.

[28] K. Yu, T. Zhang, Improved local coordinate coding using local tangents, International Conference on, Machine Learning 2010, pp. 1215–1222.

[29] K. Yu, Y. Lin, J. Lafferty, Learning image representations from the pixel level via hierarchical sparse coding, Computer Vision and Pattern Recognition 2011, pp. 1713–1720.

[30] D. Cai, H. Bao, X. He, Sparse concept coding for visual analysis, Computer Vision and, Pattern Recognition 2011, pp. 2905–2910.

[31] K. Naveen, B. Li, Discriminative affine sparse codes for image classification, Computer Vision and, Pattern Recognition 2011, pp. 1609–1616.

[32] J. Yang, K. Yu, T. Huang, Supervised translation invariant sparse coding, Computer Vision and, Pattern Recognition 2010, pp. 3517–3524.

[33] A. Coates, A.Y. Ng, The importance of encoding versus training with sparse coding and vector quantization, Proceedings of the 28th International Conference on, Machine Learning 2011, pp. 921–928.

[34] A. Shabou, H.L. Borgne, Locality-constrained and spatially regularized coding for scene categorization, Computer Vision and Pattern Recognition, 2012.

[35] N. Morioka, S. Satoh, Compact correlation coding for visual object categorization, International Conference on Computer Vision 2011, pp. 1639–1646.

[36] Y. Huang, K. Huang, C. Wang, T. Tan, Exploring relations of visual codes for image classification, Computer Vision and, Pattern Recognition 2011, pp. 1649–1656.

[37] D. Picard, P.-H. Gosselin, Improving image similarity with vectors of locally aggregated tensors, International Conference on Image Processing 2011, pp. 669–672.

[38] X. Zhao, Y. Yu, Y. Huang, K. Huang, T. Tan, Feature coding via vector difference for image classification, International Conference on Image Processing, 2012.

[39] D.H. Hubel, T.N. Wiesel, Receptive fields, binocular interaction and functional architecture in the cat's visual cortex, J. Physiol. 160 (1) (1962) 106–154.

[40] J.J. Koenderink, A.J.V. Doorn, The structure of locally orderless images, Int. J. Comput. Vis. 31 (2–3) (1999) 159–168.

[41] Y.L. Cun, B. Boser, J.S. Denker, D. Henderson, R.E. Howard, W. Hubbard, L.D. Jackel, Handwritten digit recognition with a back-propagation network, Advances in Neural Information Processing Systems 1989, pp. 396–404.

[42] N. Pinto, D.D. Cox, J.J. Dicarlo, Why is real-world visual object recognition hard, PLoS Comput. Biol. 1 (4) (2008) 151–156.

[43] T. Serre, L. Wolf, T. Poggio, Object recognition with features inspired by visual cortex, Computer Vision and, Pattern Recognition 2005, pp. 994–1000.

[44] M. Everingham, L.V. Gool, C. Williams, J. Winn, A. Zisserman, The Pascal Visual Object Classes Challenge 2007, 2007. (http://www.pascal-network.org/challenges/VOC/voc2007/workshop/index.html).

[45] A. Vedaldi, B. Fulkerson, VLFeat: An Open and Portable Library of Computer Vision Algorithms, 2008. (http://www.vlfeat.org/).

[46] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, C.-J. Lin, Liblinear: a library for large linear classification, J. Mach. Learn. Res. 9 (7694) (2008) 1871–1874.

[47] C.-C. Chang, C.-J. Lin, Libsvm: A Library for Support Vector Machines, 2001.

[48] Y. Lin, F. Lv, S. Zhu, M. Yang, T. Cour, K. Yu, Large-scale image classification: fast feature extraction and SVM training, International Conference on Computer Vision 2011, pp. 1689–1696.

[49] W. Xu, Towards optimal one pass large scale learning with averaged stochastic gradient descent, Technique Report, 2011.

[50] C.M. Bishop, Pattern Recognition and Machine Learning, Springer, 2007. (1em plus 0.5em minus 0.4em).

[51] J. Philbin, O. Chum, M. Isard, J. Sivic, A. Zisserman, Object retrieval with large vocabularies and fast spatial matching, Computer Vision and Patter Recognition 2007, pp. 1–8.

[52] M. Muja, D.G. Lowe, Fast approximate nearest neighbors with automatic algorithm configuration, VISAPP International Conference on Computer Vision Theory and Applications 2009, pp. 331–340.

[53] T. Velmurugan, T. Santhanam, Computational complexity between k-means and k-medoids clustering algorithms for normal and uniform distributions of data points, J. Comput. Sci. 6 (3) (2010) 363–368.

[54] R. Nock, F. Nielsen, On weighting clustering, IEEE Trans. Pattern Anal. Mach. Intell. 28 (8) (2006) 1223–1235.