PASCAL VOC 2012 Image Classification,
Amir Rahimi
June, 2013

1 Image classification

The methods which use local features for image classification have attracted the researchers in recent years. The main idea behind these methods is inspired by the bag-of-words methods in text classification. Csurka *et al* [1] first introduced these methods for image classification in 2004. All of these kind of feature extraction methods can be represented in a single framework with the following steps:

1. Selecting local patches on each image and extracting their features.
2. Constructing bases which are key features (dictionary learning).
3. Representing each feature by constructed bases (coding).
4. Mixing local patch codes to obtain the final feature vector (pooling).

The first step in these methods is to select a number of points along with their neighbourhoods. These points are called local points and their neighbourhoods are called local patches of images. Then, features such as SIFT[2] are extracted from these patches. The feature extraction method is chosen to be invariant to different kind of transformations such as rotation and scale. The output of this step is a D-dimensional (D = 128 for SIFT descriptors) vector for each patch. These vectors are called image descriptors and are shown by s_i. The selection method of local patches is an open problem. They are usually selected randomly, on edges, or uniformly from an image. There have been a lot of experiments on comparison of different selection strategies[3]. However, the results show that non of these strategies is completely better than the others.

Since the number of all of the descriptors in datasets such as PascalVOC is extremely large, the clustering is applied in the second step to reduce the computational cost of the problem. The centroid of each cluster, which is called a key-point or a basis, is a key feature which best represents the descriptors in the corresponding cluster. The set of the determined key-points is called a vocabulary $\mathbf{B} = [b_1, b_2, \ldots, b_M]$. The centroid of l-th cluster is shown by b_l.

In the coding step, a code a_i is assigned to each descriptor $s_i \in S$. The dimensionality of the codes is equal to the number of clusters. The simplest way to encode each descriptor is to set all of the codes to zero except the one which corresponds to the nearest centroid to the descriptor (hard coding or 1-hot encoding). This algorithm has a high rate of reconstruction error. To overcome this problem, describing each descriptor with a few nearest bases has been proposed with sufficient theoretical reasons. The tradition bag-of-words method with hard coding solves the following problem:

$$arg \min_{\mathbf{A}} \sum_{i=1}^{N} ||s_i - \mathbf{B}a_i||^2 \tag{1}$$

s.t. only one element of a_i is one and the others are zero.

where $\mathbf{A} = [a_1, a_2, \ldots, a_N]$ is the set of codes for S and $\mathbf{B} \in \mathbb{R}^{D \times M}$ is matrix which its columns are the bases of the dictionary.

Using only one basis is a hard constraint. It can be relaxed by using a sparsity regularization term. This method which is proposed in [4] solves the following problem:

$$arg \min_{\mathbf{A}} \sum_{i=1}^{N} ||s_i - \mathbf{B}a_i||^2 + \lambda ||a_i||_1 \tag{2}$$

This method yields a much less reconstruction error than hard coding and can be learnt efficiently with a linear SVM classifier.

Practical experiments in [5] shows that using local bases for feature construction achieves better results than sparse coding. Locality-constraint Linear Coding (LLC) [6] is local coding scheme which solves the following problem:

$$\begin{aligned} arg \min_{\mathbf{A}} \sum_{i=1}^{N} ||s_i - \mathbf{B}a_i||^2 + \lambda ||\mathbf{d}_i \odot a_i||^2 \\ s.t.\, \mathbf{1}^T a_i = 1, \forall i \end{aligned} \tag{3}$$

where $\odot$ denotes element-wise multiplication, and $\mathbf{d}_i \in \mathbb{R}^M$ is represented by the following formula:

$$\mathbf{d}_i = exp\left(\frac{dist(s_i, \mathbf{B})}{\sigma}\right). \tag{4}$$

where $dist(s_i, \mathbf{B}) = [dist(s_i, b_1), \ldots, dist(s_i, b_M)]^T$, and $dist(s_i, b_j)$ is the Euclidean distance between s_i and b_j. σ is an adjustable constant. Local smooth sparsity, analytical solution and less reconstruction error than hard coding are the main features of LLC coding. To further speeding up the process of coding, an approximate LLC coding is also proposed in [6]. Instead of solving 3, $K (K < D < M)$ nearest neighbours of s_i are considered as its local bases $\mathbf{B}_i$. So, a much smaller linear system can be solved for coding:

$$\begin{aligned} arg \min_{\mathbf{A}} \sum_{i=1}^{N} ||s_i - \tilde{a}_i \mathbf{B}||^2 \\ s.t.\, \mathbf{1}^T \tilde{a}_i = 1, \forall i \end{aligned} \tag{5}$$

This reduces the computational complexity from $\mathcal{O}(M^2)$ to $\mathcal{O}(M + K^2)$, where $K \ll M$.

The last step for generating features is to mix them to construct a single feature vector for each image (pooling). Computing the normalized histogram along the coding vector is the most common method in this step. However, it has been reported (without theoretical reasons) that using a maximum operator yields better results [4].

One of the problems in bag-of-words methods is that the final features are independent of the location of local points. It means that changing the location of the patches across the image has no impact on the final results. Spatial pyramid matching (SPM)[7] is proposed to overcome this problem. The main idea of this method is inspired by the pyramid matching kernel [8] method. In SPM the image is partitioned into different sub-regions at different levels. Pooling is applied on each region at each level. Then, all of the

Table 1: Image classification results on PASCAL VOC 2012 validation set

object class	aero	bicyc	bird	boat	bottle	bus	car	cat	chair	cow
HC	46.52	11.15	20.40	29.27	12.09	31.03	33.10	26.69	28.18	7.86
LLC	**78.91**	41.80	39.85	**51.64**	13.62	**73.59**	46.33	54.44	43.34	**19.43**
HIC	71.39	**43.84**	**44.05**	48.39	**16.75**	72.68	**50.91**	**55.28**	**48.26**	18.11

object class	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv
HC	12.29	28.14	14.62	25.24	52.18	9.72	18.51	15.88	21.69	29.13
LLC	33.05	40.27	**39.55**	**54.05**	67.80	14.00	**38.33**	30.45	60.01	46.46
HIC	**37.18**	**44.70**	38.90	52.87	**75.55**	**14.60**	35.20	38.30	**64.40**	**53.57**

object class	Average
HC	23.68
LLC	44.35
HIC	**46.25**

vectors obtained from each region at all levels are concatenated to create the final feature vector. Hence, the location of patches have their own influence on the final feature vector. Although good results have been reported on SPM based methods, these methods need non-linear kernels in order to have good results. So, they are computationally complex.

2 Implementation and results

I used the Spatial Pyramid code provided by Prof. Lazebnik, VLFeat package, LLC code, practical image classification guide, and libsvm in order to extract features and classify images. I used SIFT descriptors for feature extraction on each patch. The patches are selected on a uniform grid on each image. The size of window is 16 pixels. The step between each patch is 6 pixels. The descriptor vectors are normalized to have unit length. Sift descriptor generation is done in parallel using MATLAB. I used k-means clustering method on 100,000 randomly selected descriptors to build a $M = 1024$ dimensional dictionary. It should be noted that selecting more descriptors and building a bigger dictionary will improve the accuracy of the overall classification. However, they need more RAM, CPU, and running time. These values are set so that my algorithm can be runnable with my limited computational resources.

I used hard coding (HC) and LLC with spatial pyramid to construct the feature vectors for each image. Then a linear SVM is used for training/testing the performance of these methods. I also used the histogram intersection kernel method on Hard coded k-means clusters with spatial pyarmid (HIK). Since there may be multiple objects in one picture, the training and testing is in 1-vs-all method. The PASCAL VOC 2012 dataset consists 5717 training and 5823 validation images in 20 different categories. There are some difficult objects on this dataset where their label are set to zero. I ignored these pictures for training and testing. The classification performance is evaluated using the Average Precision measure. It is equal to the area under the Precision/Recall curve, and the higher the score, the better is the performance. The results of these methods for all classes are shown in table 1.

References

[1] G. Csurka, C. Dance, L. Fan, J. Willamowski, C. Bray, "Visual categorization with bags of keypoints," *In ECCV Workshop on statistical learning in computer vision*, Vol. 1, p.22, 2004.

[2] D. Lowe, "Object recognition from local scale-invariant features," *In Computer Vision,1999. The Proceedings of the Seventh IEEE International Conference on*, Vol. 2, pp.1150-1157, IEEE, 1999.

[3] E. Nowak, F. Jurie, and B. Triggs, "Sampling strategies for bag-of-features image classication," *In European Conference on Computer Vision, ECCV*, pp.490-503, 2006.

[4] J. Yang, K. Yu, Y. Gong, and T. Huang, "Linear spatial pyramid matching using sparse coding for image classication," *In Proc. of IEEE Computer Vision and Pattern Recognition, CVPR*, 2009.

[5] K. Yu, T. Zhang, and Y. Gong, "Nonlinear learning using local coordinate coding," *Advances in Neural Information Processing Systems*, Vol. 22, pp. 2223-2231, 2009.

[6] J. Wang, J. Yang, K. Yu,F. Lv, T. Huang, and Y. Gong, "Locality-constrained linear coding for image classication," *In Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pp. 3360-3367, 2010.

[7] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories," *In Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, Vol. 2, pp. 2169-2178, IEEE, 2006.

[8] K. Grauman, and T. Darrell, "The pyramid match kernel: Discriminative classication with sets of image features," *In Computer Vision (ICCV), 2005 International Conference on*, Vol. 2, pp.1458-1465, IEEE, 2005.