Using Spatial Pyramids with Compacted VLAT for Image Categorization

Romain Negrel, David Picard and Philippe-Henri Gosselin
ETIS/ENSEA - University of Cergy-Pontoise - CNRS, UMR 8051
6, avenue du Ponceau, BP44, F95014 Cergy-Pontoise, France
$\{$*romain.negrel,picard,gosselin*$\}$*@ensea.fr*

Abstract

In this paper, we propose a compact image signature based on VLAT. Our method integrates spatial information while significantly reducing the size of original VLAT by using two pojection steps. we carry out experiments showing our approach is competitive with state of the art signatures.

1. Introduction

Content Based Image Categorization is a research field that has known a rapid development in the last years. The typical scheme is composed of the following steps: extraction of local image descriptors, aggregation of the descriptors in an image signature, and learning a classifier in signatures space. The most recent and successful methods are Coding/pooling [9], Fisher Vector [6], VLAD [4], and VLAT [7].

In this paper we present a new method for aggregating local descriptors based on VLAT [7]. Our method improves significantly the classification results over of VLAT, while reducing its size.

This paper is organized as follows: the next section describes recent related work on image representation and similarities. Section 3 introduces our novel approach. Finally we present successful experiments on the well known VOC2007 dataset [3].

2. State of the art

In this section we present the most recent and efficient image signatures using local descriptors.

2.1. Coding/Polling Schemes

Coding/Polling methods proposed in [9] by J. Wang et al. are a generalization of BoW methods[8]. These methods are based on visual words dictionary. The visual words are computed using a clustering algorithm on a training set of descriptors. The Coding/Pooling schemes divide the problem into two steps. The first step maps each descriptor of the image on the codebook ("coding" step). The second step aggregates the mapped descriptors into a single signature ("pooling" step). For the Coding step in [9], they compute a mapped vector $\boldsymbol{\alpha}^\star$ as the result of the following reconstruction problem:

$$\boldsymbol{\alpha}^\star = \underset{\boldsymbol{\alpha}}{\operatorname{argmin}} ||\mathbf{b} - \mathbf{W}\boldsymbol{\alpha}||^2 + ||\mathbf{d} \circ \boldsymbol{\alpha}||^2, \qquad (1)$$

with $\mathbf{b}$ being a descriptor, $\mathbf{W}$ the codebook matrix, $\boldsymbol{\alpha}$ the projection coefficients and $\mathbf{d}$ a locality constraint. The output vector $\boldsymbol{\alpha}^\star$ is thus optimized with respect to the reconstruction error and constrained to the projection on few nearby codewords. For the Pooling step, they propose 2 methods: Sum Pooling (sum of all codes) or Max Pooling (maximum value for each coding coefficient). Compared to BoW, the coding/pooling schemes give good performance with linear similarities, while retaining a small signature size. However this method is more complex to implement and requires a fine tuning of the parameters to obtain good results [2].

2.2. Fisher Vector

Recently, Perronnin et al.[6] presented a new method called Fisher Vectors. The authors hypothesize the descriptors can be modeled by a probability density function denoted as u_λ of parameters λ. To describe the image, they use the derivative of the log-likelihood of all image descriptors to the model. The key idea is to describe the contribution of the descriptors to the deviation of the model parameters. The model used is a GMM of parameters $\boldsymbol{\mu}_c$ and σ_c. The elements of the Fisher Vector for each Gaussian c can be written as:

$$\mathcal{G}_{\boldsymbol{\mu},c}^{\mathbf{B}_i} = \frac{1}{T\sqrt{\omega_c}} \sum_r \gamma_c(\mathbf{b}_{ri}) \left(\frac{\mathbf{b}_{ri} - \boldsymbol{\mu}_c}{\boldsymbol{\sigma}_c} \right), \qquad (2)$$

$$\mathcal{G}_{\boldsymbol{\sigma},c}^{\mathbf{B}_i} = \frac{1}{T\sqrt{\omega_c}} \sum_r \gamma_c(\mathbf{b}_{ri}) \left[\frac{(\mathbf{b}_{ri} - \boldsymbol{\mu}_c)^2}{\boldsymbol{\sigma}_c^2} - 1 \right], \qquad (3)$$

Where $\mathbf{b}_{ri}$ are the descriptors of image i, $(\omega_c, \boldsymbol{\mu}_c, \sigma_c)$ are the weight, mean and standard deviation of Gaussian c, and $\gamma_c(\mathbf{b}_{ri})$ the normalized likelihood of $\mathbf{b}_{ri}$ to

Gaussian c. The final descriptor is obtained by concatenation of $\mathcal{G}^{\mathbf{B}_i}_{\boldsymbol{\mu},c}$ and $\mathcal{G}^{\mathbf{B}_i}_{\boldsymbol{\sigma},c}$ for all Gaussians. The Fisher Vector method achieve very good results [2]. However, Fisher Vectors are limited to the simple model of mixtures of Gaussians with diagonal matrices. Moreover, the GMM algorithm is computationally very intensive.

2.3. Vector of Locally Aggregated Tensors

In [7], Picard et al. proposed an extension of Fisher vectors, called VLAT for Vector of Locally Aggregated Tensors. This method uses a codebook of visual words, computed with a clustering algorithm (typically: k-mean). They compute the 1st and 2nd order moments for each visual word (cluster):

$$\boldsymbol{\mu}_c = \frac{1}{|c|} \sum_i \sum_r \mathbf{b}_{rci}, \tag{4}$$

$$\mathcal{T}_c = \frac{1}{|c|} \sum_i \sum_r (\mathbf{b}_{rci} - \boldsymbol{\mu}_c)(\mathbf{b}_{rci} - \boldsymbol{\mu}_c)^\top, \tag{5}$$

with $|c|$ the number of descriptors in cluster c and $\mathbf{b}_{rci}$ the descriptors of image i belonging to cluster c. For each image, the difference between image 2nd order moment and the cluster 2nd order moment is computed for each cluster c:

$$\mathcal{T}_{i,c} = \sum_r (\mathbf{b}_{rci} - \boldsymbol{\mu}_c)(\mathbf{b}_{rci} - \boldsymbol{\mu}_c)^\top - \mathcal{T}_c. \tag{6}$$

Each $\mathcal{T}_{i,c}$ is flattened into a vector $\mathbf{v}_{i,c}$. The VLAT signature $\mathbf{v}_i$ for image i consists in the concatenation of $\mathbf{v}_{i,c}$ for all clusters c:

$$\mathbf{v}_i = (\mathbf{v}_{i,1} \dots \mathbf{v}_{i,C}). \tag{7}$$

It is advisable to perform a normalization step for best performance.

$$\forall j, \quad \mathbf{v}'_i[j] = sign(\mathbf{v}_i[j])|\mathbf{v}_i[j]|^\alpha, \tag{8}$$

$$\mathbf{x}_i = \frac{\mathbf{v}'_i}{||\mathbf{v}'_i||}, \tag{9}$$

with $\alpha = 0.5$ typically. VLAT gives very good results in similarity search and automatic indexing of images with linear metric, but leads to large feature vectors. The size of the VLAT signature is $C \times D \times D$, with C the number of clusters and D the size of descriptors.

2.4. Spatial pyramid binning

The spatial pyramid method [5] allows the introduction of spatial information in image signatures. The process consists in computing a set of p signatures for different regions of the image. To compute the similarity between images, the weighted sum of similarities for all regions is performed. It can be written as the following kernel function:

$$K(X_i, X_j) = \sum_{m=1}^{p} \alpha_m k(X_i^{(m)}, X_j^{(m)}), \tag{10}$$

with $k(\cdot,\cdot)$ the minor kernel function and $X_i^{(m)}$,$X_j^{(m)}$ the signatures of image i and j for region m.

3. Proposed method

We propose a new VLAT based signature that significantly reduces the size of the signature, while increasing their discriminative power. Our scheme is as follows:
1. We perform a Principal Component Analysis (PCA) within each cluster (We project the descriptors in the space of the eigenvectors of their cluster).
2. We compute VLAT signatures for different part of the spatial pyramid.
3. For some training set of images, we then compute the Gram matrix using Eq. (10) with a linear minor kernel.
4. We then perform a low rank approximation of the gram matrix and compute the set of projection vectors associated with the generated subspace.
The final signatures are the projections of normalized pyramid VLAT using the obtained projection vectors.

3.1. Pre-projection: descriptors PCA by cluster

As for standard VLAT, we compute the 1st and 2nd order moments (Eq. 4 and 5) of each cluster. We perform a Takagi factorization of $\mathcal{T}_c$ matrix:

$$\mathcal{T}_c = \mathbf{V}_c \mathbf{D}_c \mathbf{V}_c^\top, \tag{11}$$

where $\mathbf{D}_c$ is a diagonal matrix formed from the eigenvalues of $\mathcal{T}_c$ and $\mathbf{V}_c$ is a matrix of eigenvectors of $\mathcal{T}_c$. We denote by $\mathbf{D}_{c,p_c}$ the matrix with the p_c largest eigenvalues on the diagonal:

$$\mathbf{D}_{c,p_c} = diag(\lambda_{c,1} \dots \lambda_{c,p_c}), \tag{12}$$

with $\lambda_{1,p_c} \geq \lambda_{2,p_c} \geq \dots \geq \lambda_{c,p_c}$ and we denote by $\mathbf{V}_{c,p_c}$ the matrix of the first p_c eigenvectors:

$$\mathbf{V}_{c,p_c} = [\mathbf{v}_{c,1} \dots \mathbf{v}_{c,p_c}]. \tag{13}$$

We perform an approximation of the image descriptors by projection on the subspace spanned by the p_c largest eigenvectors of $\mathcal{T}_c$:

$$\mathbf{b}'_{rci} = \mathbf{V}_{c,p_c}^\top (\mathbf{b}_{rci} - \boldsymbol{\mu}_c). \tag{14}$$

We can thus rewrite Eq. (6):

$$\mathcal{T}'_{i,c} = \mathbf{V}_{c,p_c}^\top \mathcal{T}_{i,c} \mathbf{V}_{c,p_c}. \tag{15}$$

Figure 1. Images from PASCAL Visual Object Classes Challenge 2007.

Like in VLAT, we concatenate each cluster signature and do proper normalization. Due to the tensor product, the size of the new signature depends on the square of the number of eigenvector selected in each cluster $\left(\sum_{c=1}^{C} p_c^2\right)$. We propose 2 strategies for selecting the eigenvectors in each cluster. The *fixed dimension* method is to arbitrarily set the number of eigenvalue that is kept for each cluster : $p_1 = \cdots = p_C$. The *error threshold* method is to set a threshold ε on the error introduced by the subspace projection:

$$p_c = \min_p p \quad \text{s.t.} \quad 1 - \frac{\sum_{i=1}^{p} \lambda_{c,i}}{\sum_{i=1}^{D} \lambda_{c,i}} \leq \varepsilon. \qquad (16)$$

3.2. Spatial pyramid

To add spatial information, we compute the signature of Eq.(15) for different region of spatial pyramid. The spatial regions are obtained by dividing image using 1×1, 3×1, and 2×2 grids, for a total of 8 regions like in most image categorization setups [3, 2]. We use a linear metric for classification (e.g. the minor kernel of Eq. (10) is the dot product). Hence, the global signature for the image is the concatenation of weighted signatures for each region of the pyramid:

$$\mathbf{x}_i = \left(\sqrt{\alpha_1}\mathbf{x}_{i,1} \ldots \sqrt{\alpha_m}\mathbf{x}_{i,m} \ldots \sqrt{\alpha_8}\mathbf{x}_{i,8}\right), \qquad (17)$$

with $\mathbf{x}_{i,m}$ is the signature of the image i computed in the area m of the pyramid.

3.3. Post-projection: Low rank approximation

Given a training set $\mathcal{S}$ of N images ,we compute the Gram matrix $\mathbf{G}$ of signatures defined in Eq. (17):

$$\mathbf{G}_{i,j} = \mathbf{x}_i^\top \mathbf{x}_j, \quad \mathbf{x}_i \in \mathcal{S}, \mathbf{x}_j \in \mathcal{S}. \qquad (18)$$

Then, we perform the Takagi factorization of $\mathbf{G}$:

$$\begin{aligned} \mathbf{G} &= \mathbf{U}\mathbf{L}\mathbf{U}^\top, & (19)\\ \mathbf{L} &= diag(\lambda_1 \ldots \lambda_i \ldots \lambda_N), & (20)\\ \mathbf{U} &= (\mathbf{u}_1 \ldots \mathbf{u}_i \ldots \mathbf{u}_N), & (21) \end{aligned}$$

with $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_N$ and $\mathbf{u}_i$ the eigenvector associated with the eigenvalue λ_i. We want to compute a low rank approximation of the Gram matrix. We denote by $\mathbf{L}_t$ the matrix with the t largest eigenvalues on the diagonal:

$$\mathbf{L}_t = diag(\lambda_1 \ldots \lambda_t), \qquad (22)$$

and we denote by $\mathbf{U}_t$ the matrix of the first t eigenvectors:

$$\mathbf{U}_t = \left[\mathbf{u}_1 \ldots \mathbf{u}_t\right]. \qquad (23)$$

Then, we compute the projectors in the approximated subspace:

$$\mathbf{P}_t = \mathbf{X}\mathbf{U}_t\mathbf{L}_t^{-1/2}, \qquad (24)$$

with $\mathbf{X} = [\mathbf{x}_1 \ldots \mathbf{x}_N]$ is the matrix of signatures for the training set $\mathcal{S}$. This method akin to Kernel-PCA using the Spatial Pyramidal Kernel with a linear minor kernel.

3.4. Projection vectors

For some image i, we compute the projected signature in the approximated space:

$$\mathbf{y}_i = \mathbf{P}_t^\top \mathbf{x}_i, \qquad (25)$$

$\mathbf{y}_i$ contains an approximate and compressed version of $\mathbf{x}_i$ (from eq.(17)). This method selects the most energy directions of signature space. The size of $\mathbf{y}_i$ is very small compared to the size of $\mathbf{x}_i$, as it directly depends on t (varying from 1 to N). The final step is a ℓ_2 normalization of $\mathbf{y}_i$:

$$\mathbf{z}_i = \frac{\mathbf{y}_i}{||\mathbf{y}_i||}. \qquad (26)$$

4. Experiments

In this section we describe our experimental setup and some results on the PASCAL-VOC 2007 dataset [3]. This dataset consists in about 10,000 images and 20 categories, and is divided into 3 parts : *train*, *val* and *test*. We used an 1-vs-rest linear SVM classifier trained on *train* + *val* sets and tested on the *test* set. We used a fast stochastic gradient descent algorithm with C =

	mAP	aeroplane	bicycle	bird	boat	bottle	bus	car	cat	chair	cow
Our method	61.5	**80.3**	**72.2**	51.4	**71.4**	28.1	72.1	**81.6**	**63.1**	54.4	47.5
FV	**61.7**	79.0	67.4	**51.9**	70.9	**30.8**	**72.2**	79.9	61.4	**56.0**	**49.6**
LLC	57.6	71.1	62.9	47.4	67.7	25.2	62.7	77.0	59.6	54.2	45.3
	Size	table	dog	horse	bike	person	plant	sheep	sofa	train	tv
Our method	10k	57.8	**46.5**	**81.1**	70.3	**86.8**	30.8	41.2	54.0	**84.1**	54.9
FV	320k	**58.4**	44.8	78.8	**70.8**	85.0	**31.7**	**51.0**	**56.4**	80.2	**57.5**
LLC	200k	51.6	44.2	75.5	67.1	83.3	27.6	45.7	53.6	76.0	52.3

Table 1. Image classification results on Pascal VOC 2007 dataset compared to [2]

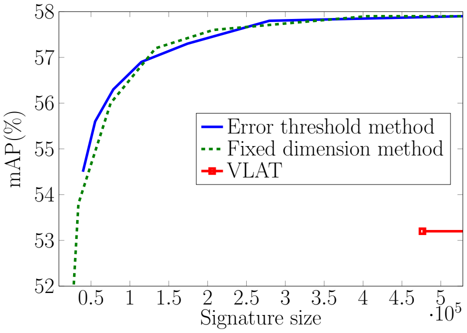

Figure 2. Effect of pre-projection

10 [1]. We used Mean Average Precision (mAP) over the 20 classes for performance measurement. We extracted HOG features with 4 different scales (4px, 6px, 8px and 10px cell size), every 3 or 6 pixels, depending on the test. We used a simple k-means clustering algorithm in order to obtain the codebook. For the low-order approximation of the post-projection step, we used the full dataset Gram matrix.

4.1. Pre-projection effect

In this section we study the action of the pre-projection (section 3.1). We used a 6 pixels step for extracting HOG features, a codebook of 64 visual words, only the first level of the spatial pyramid (1×1) and no post-projection. To compare the two strategies for estimating p_c, we measured the classification score against the signature size. For the *fixed dimension* strategy, we reduced the descriptor size to 32, 48, 64, 80, 112 and 128 dimensions. For the *error threshold* strategy, we tested 30%, 25%, 20%, 15%, 10%, 5% and 0% of reconstruction error rate ε. As we can see in Figure 2 the pre-projection significantly increases the performance relative to VLAT even if it greatly reduces the signatures size. We can also see that the thresholding strategy gives better results, which is consistent with the fact it balances the error among clusters.

4.2. Classification results

To compare our method to the state of the art, we used a similar setup to [2]. We used a 3 pixels step for extracting HOG features, a codebook of 256 visual words, and all levels of the spatial pyramid. We set the threshold ε to 10% of reconstruction error (about 74 dimensions). We used all projectors in post-projection step (about 10k signature size). We can see from Table 1, our signature gives comparative results to the of state of the art methods. However, our signature is much smaller and we used HOG descriptors (simpler than SIFT used in [2]).

5. Conclusion

In this paper, we proposed a new image signature based on VLAT. Our representation increases discriminative power of VLAT by introducing a spatial information while reducing its size significantly. Tests on VOC 2007 dataset show that our signature gives results similar to state of the art methods, with simpler descriptors and smaller signature. The results are very promising, and we are looking forward to a full investigation of the influence of two projection steps.

References

[1] L. Bottou. Large-scale machine learning with stochastic gradient descent. In Y. Lechevallier and G. Saporta, editors, *ICCS*, pages 177–187, Paris, France, August 2010.

[2] K. Chatfield, V. Lempitsky, A. Vedaldi, and A. Zisserman. The devil is in the details: an evaluation of recent feature encoding methods. In *BMVC*, volume 76, 2011.

[3] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2007 Results, 2007.

[4] H. Jégou, F. Perronnin, M. Douze, J. Sánchez, P. Pérez, and C. Schmid. Aggregating local image descriptors into compact codes. *IEEE PAMI*, 2012. QUAERO.

[5] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *IEEE CVPR*, pages 2169–2178, Washington, DC, USA, 2006. IEEE Computer Society.

[6] F. Perronnin, J. Sánchez, and T. Mensink. Improving the fisher kernel for large-scale image classification. In *ECCV*, pages 143–156, 2010.

[7] D. Picard and P. Gosselin. Improving image similarity with vectors of locally aggregated tensors. In *ICIP*, Brussels, Belgique, September 2011.

[8] J. Sivic and A. Zisserman. Video google: A text retrieval approach to object matching in videos. In *ICCV*, 2003.

[9] J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, and Y. Gong. Locality-constrained linear coding for image classification. In *IEEE CVPR*, pages 3360–3367, 2010.