Evaluation of Second-order Visual Features for Land-Use Classification
Romain Negrel, David Picard, Philippe-Henri Gosselin

To cite this version:
Romain Negrel, David Picard, Philippe-Henri Gosselin. Evaluation of Second-order Visual Features for Land-Use Classification. 12th International Workshop on Content-Based Multimedia Indexing, Jun 2014, France. 5 p. hal-01022971

HAL Id: hal-01022971
https://hal.archives-ouvertes.fr/hal-01022971
Submitted on 11 Jul 2014

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L’archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d’enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.
Evaluation of Second-order Visual Features for Land-Use Classification

Romain Negrel*, David Picard* and Philippe-Henri Gosselin†

*ETIS/ENSEA - UCP - CNRS, F95014 Cergy, France
†Texmex team, Inria, F35042 Rennes, France

Email: romain.negrel, picard, gosselin@ensea.fr
Email: philippe.gosselin@inria.fr

Abstract—This paper investigates the use of recent visual features based on second-order statistics, as well as new processing techniques to improve the quality of features. More specifically, we present and evaluate Fisher Vectors (FV), Vectors of Locally Aggregated Descriptors (VLAD), and Vectors of Locally Aggregated Tensors (VLAT). These techniques are combined with several normalization techniques, such as power law normalization and orthogonalisation/whitening of descriptor spaces. Results on the UC Merced land use dataset shows the relevance of these new methods for land-use classification, as well as a significant improvement over Bag-of-Words.

I. INTRODUCTION

In land-use classification of high-resolution overhead imagery, the most popular pipeline is composed of two parts: the extraction of image descriptors; and the use of statistical learning tools. To extract the descriptors, two types of methods are used: global descriptors; aggregated descriptors. The global descriptors commonly used are color descriptors [1] (e.g. histogram of RGB, HSV, CIE Lab) and homogeneous texture descriptors [1] (Gabor filter responses). The aggregated image descriptors are computed in two steps: the extraction of a set of local descriptors; and their aggregation in a single descriptor. The most popular local descriptors is the well known SIFT descriptor [2]. Many aggregation methods have been proposed and evaluated [1], [3–7]. Yang et al. [1], [3] evaluated numerous methods used in computer vision: the bag-of-visual-words [8] and its spatial extensions (Spatial Pyramid Match Kernel [9] and Spatial Co-occurrence Kernel [10]). Risojevic et al. [4] propose to use the cross-correlations between Gabor wavelet coefficient and quaternion framework for the representation of color images to compute the image descriptor. Recently, Chertiyadat et al. [5] propose to use a coding/pooling method [11] to aggregate the local descriptors into a single descriptor.

In this paper, we propose to evaluate recent methods of aggregated descriptors for visual representation of land-use classification in high-resolution overhead imagery [1]. This includes improved Fisher Vectors (FV) [12], Vectors of Locally Aggregated Descriptors (VLAD) [13], Vectors of Locally Aggregated Tensors (VLAT) [14], and many improvements that were recently proposed [15]. These methods are the extension of visual dictionaries approaches introduced by Bag-of-Words (BoW) [8], thanks to several key ideas. A first key idea is the introduction of the deviation approach, which was first motivated by statistical models from Fisher Kernels [12]. The idea is to consider the deviation between the local image model and a global model, rather than only the local image model. As a result, specific properties of an image are better emphasised. A second key idea is the introduction of second-order statistics, for instance using covariance data in addition to mean data. While the feature size is then significantly increased, recent techniques for high dimensionality reduction have solved this problem [13], [15]. Another key idea is the introduction of normalisation processing at the different levels of the tool-chain, like the power law [12] or cluster-wise component analysis [15]. A last key idea is to only consider features vectors compared with a linear similarity (e.g. dot product). These constraints allow the use of very efficient retrieval and learning techniques, like Stochastic Gradient Descent SVM [16]. Furthermore, in most cases this leads to methods with computational and memory complexity linear with the size of datasets, rather than quadratic complexity with non-linear features.

In this scope, we first propose a detailed presentation of these methods in Section II. Then, we present in Section III evaluation results on UC Merced land use dataset [1].

II. IMAGE FEATURES

Most image features are obtained by a two steps scheme. The first step is to extract a set of local visual descriptors from the images. The most commonly used visual descriptors are highly discriminant local descriptors (HOG, SIFT, SURF, ...). Regions of interest can be selected by uniform sampling, or by automatic point of interest detection. The set extracted from an image is called a bag. We denote by \( B_i = \{ b_{rij} \} \), the set of descriptors \( b_{rij} \in \mathbb{R}^D \) in image \( i \). The second step is to map the descriptors of the bag \( B_i \) into a single vector \( x_i \in \mathbb{R}^W \), known as the image feature.

A. Statistical Approaches

The first methods to map the descriptors in a feature are based on the statistical study of the distribution of descriptors in the bag. These approaches have been inspired by text retrieval methods. To study the distribution of descriptors, we use a visual codebook composed by \( C \) visual words. The visual codebook is generally computed by a clustering algorithm (e.g., \( k \)-means) on a large sample of descriptors. A bag can then be described by a statistical analysis of occurrences of visual words.

The first method of this kind, named Bag of Words (BoW) [8] counts the number of descriptors belonging to
each cluster. The dimension of the feature is then \( C \). Many extension of BoW has been proposed [11], for example in the classification of urban scenes in geo-referenced images [17]. These approaches obtain good results in similarity search and in images classification. However, to obtain good results with these methods it is necessary to use visual codebooks with very large dictionaries (about 100k visual word), and the use of a non-linear metric.

B. Model Deviation Approaches

Perronnin et al. [12] proposed a successful method called Fisher Vectors. This method uses a probability density function denoted by \( u_\lambda \) of parameters \( \lambda \) as model of the descriptors space. To describe the image, they compute the derivative of the log-likelihood of image descriptors to the model:

\[
G^B_\lambda = \frac{1}{T} \nabla \lambda \log u_\lambda (B_i). \tag{1}
\]

The authors propose to use a Gaussian Mixture Model (GMM) of parameters \( \mu_c \) and \( \sigma_c \). Elements of the Fisher Vector for each Gaussian \( c \) can be written as:

\[
G^{B}_{\mu,c} = \frac{1}{T \sqrt{\omega_c}} \sum_r \gamma_c(b_{ri}) \left( \frac{b_{ri} - \mu_c}{\sigma_c} \right), \tag{2}
\]

\[
G^{B}_{\sigma,c} = \frac{1}{T \sqrt{\omega_c}} \sum_r \gamma_c(b_{ri}) \left[ \frac{(b_{ri} - \mu_c)^2}{\sigma_c^2} - 1 \right]. \tag{3}
\]

Where \((\omega_c, \mu_c, \sigma_c)\) are the weight, mean and standard deviation of Gaussian \( c \), and \( \gamma_c(b_{ri}) \) the normalized likelihood of \( b_{ri} \) to Gaussian \( c \). The final feature is obtained by concatenation of \( G^{B}_{\mu,c} \) and \( G^{B}_{\sigma,c} \) for all Gaussians. Fisher Vectors achieve very good results [12]. However, Fisher Vectors are limited to the simple model of mixtures of Gaussians with diagonal covariance matrices. Moreover, the GMM algorithm is computationally very intensive.

Jegou et al. [13] proposed a simplified version of Fisher Vector by aggregating local descriptors, called Vectors of Locally Aggregated Tensors (VLAT). They proposed to model the descriptors space by a small codebook obtained by clustering a large set of descriptors. The model is simply the sum of all centered descriptors \( B_{ci} = \{ b_{rci} \} \subseteq B_i \) from image \( i \) and cluster \( c \):

\[
\nu_{ci} = \sum_{b_{rci}} b_{rci} - \mu_c, \tag{4}
\]

with \( \mu_c \) the center of cluster \( c \). The final feature is obtained by a concatenation of \( \nu_{ci} \) for all \( c \). The feature size is \( D \times C \).

Picard et al. [14] proposed an extension of VLAT by aggregating tensor products of local descriptors, called Vector of Locally Aggregated Tensors (VLAT). They proposed to use the covariance matrix of the descriptors of each cluster. Let us denote by “\( \mu_c \)” the mean of cluster \( c \) and “\( T_c \)” the covariance matrix of cluster \( c \) with \( b_{rci} \) descriptors belonging to cluster \( c \):

\[
\mu_c = \frac{1}{|c|} \sum_i \sum_r b_{rci} \tag{5}
\]

\[
T_c = \frac{1}{|c|} \sum_i \sum_r (b_{rci} - \mu_c)(b_{rci} - \mu_c)^\top, \tag{6}
\]

with \(|c|\) being the total number of descriptors in cluster \( c \).

For each cluster \( c \), the feature of image \( i \) is the sum of centered tensors of centered descriptors belonging to cluster \( c \):

\[
T_{ic} = \sum_r (b_{rci} - \mu_c)(b_{rci} - \mu_c)^\top - T_c. \tag{7}
\]

Each \( T_{ic} \) is flattened into a vector \( v_{ic} \). The VLAT feature \( v_i \) for image \( i \) consists of the concatenation of \( v_{ic} \) for all clusters:

\[
v_i = (v_{i1} \ldots v_{ic}). \tag{8}
\]

As the \( T_{ic} \) matrices are symmetric, only the diagonal and the upper part are kept while flattening \( T_{ic} \) into a vector \( v_{ic} \). The size of the feature is then \( C \times \frac{D(D+1)}{2} \).

C. Normalization

Several normalization processing are proposed in the literature to enhance the quality of visual features. The most popular one is the power law normalization, which first raises each value to a power \( \alpha \), and then \( \ell_2 \)-normalize the resulting vector:

\[
x_i = \frac{v_i'}{|v_i'|}, \quad \forall j, v'_j[i] = \text{sign}(v_i[j])|v_i[j]|^\alpha, \tag{9}
\]

with \( \alpha \) typically set to 0.5. This normalization was first introduced for the final visual features [12], and more recently for the normalization of low-level descriptors, such as Root-SIFT [18].

Another common improvement is the orthogonalization and/or whitening of vector spaces, using a Principal Component Analysis. This can be performed at different levels. For instance, this is a required pre-processing on low-level descriptors for Fisher Vectors [12]. It can also be used to normalize each cluster of the dictionary, as it is done for VLAD [19] and for VLAT [15].

In the case of VLAT, the processing is the following one. First, we compute the eigendecomposition of the covariance matrix of each cluster \( c \):

\[
T_c = V_c D_c V_c^\top, \tag{10}
\]

where \( D_c \) is a real non-negative diagonal matrix (eigenvalues), and \( V_c \) is unitary (eigenvectors). Then we project the centered descriptors belonging to \( c \) on the eigenvectors:

\[
b'_{rci} = V_c^\top (b_{rci} - \mu_c). \tag{11}
\]

Combining eq.(11) and eq.(7), we get:

\[
T_{ic} = V_c^\top \left( \sum_r (b_{rci} - \mu_c)(b_{rci} - \mu_c)^\top - T_c \right) V_c
\]

\[
= \sum_r V_c^\top ((b_{rci} - \mu_c)(b_{rci} - \mu_c)^\top) V_c - D_c
\]

\[
= \sum_r (V_c^\top (b_{rci} - \mu_c)) (V_c^\top (b_{rci} - \mu_c))^\top - D_c.
\]

The cluster-wise normalized VLAT feature of image \( i \) in cluster \( c \) is the sum of tensors of projected descriptors \( b'_{rci} \) belonging to cluster \( c \), centered by \( D_c \):

\[
T_{ic} = \sum_r b'_{rci} b'_{rci}^\top - D_c. \tag{12}
\]
In this section, we present the result using FV, VLAD and VLAT features on UC Merced land use dataset [1].

A. Dataset

This dataset is composed of 256 × 256 pixels RGB images, with pixel resolution of one foot. They are manually classified into 21 classes, corresponding to various land cover and land use types: agricultural, airplane, baseball diamond, beach, buildings, chaparral, dense residential, forest, freeway, golf course, harbor, intersection, medium density residential, mobile home park, overpass, parking lot, river, runway, sparse residential, storage tanks, and tennis courts. Each class contains 100 images. Examples are presented in Fig. 1.

We evaluate methods using the same protocol as in [1]. This is a five-fold multi-classification protocol. We randomly split the dataset into five subsets, train using four subsets and test using the remaining one. The results presented in the following sections are then average classification accuracy over the five runs. We follow a one-versus-all classification strategy: for each class, we train a linear SVM classifier, and label each test image with the classifier that returns the highest classification score.

For all the following experiments, we $\ell_2$ normalize low-level descriptors (no power law at this stage), and normalize final visual features using a power law with $\alpha = 0.5$.

B. Features comparison

We first evaluated the scale parameter of low-level descriptors. For this purpose, we used HOG descriptors at different scales: cells of 4 pixels, 6 pixels, 8 pixels and 10 pixels. Then, we carried out experiment with VLAD features for different dictionary sizes $D$, from 32 to 256 keywords. As presented in Table I, individual scales presents similar performance. Furthermore, this behaviour is the same with different dictionary size: in all cases, performance is improved by the size of the dictionary, but is similar from one scale to another. However, when combining all scales, the performance is improved. As a result, we always consider the combination of these four scales in the following experiments.

The second set of experiments compares FV, VLAD and VLAT using HOG descriptors and RGB descriptors [12], for different dictionary sizes $D$. Results are presented in Table II. If we compare HOG and RGB descriptors, HOG are more effective. Let note that we did not combine HOG and RGB descriptors because no normalization is performed on descriptor spaces in these experiments. Focusing on the dictionary size, improvement can be observed with VLAD, however, this is less significant for FV and VLAT. Finally, the best feature in these experiment is the VLAT feature.

<table>
<thead>
<tr>
<th>$D$</th>
<th>32</th>
<th>64</th>
<th>128</th>
<th>256</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>84.5</td>
<td>86.7</td>
<td>88.5</td>
<td>89.0</td>
</tr>
<tr>
<td>6</td>
<td>84.8</td>
<td>86.0</td>
<td>87.6</td>
<td>89.0</td>
</tr>
<tr>
<td>8</td>
<td>83.2</td>
<td>86.7</td>
<td>87.7</td>
<td>88.6</td>
</tr>
<tr>
<td>10</td>
<td>83.8</td>
<td>86.5</td>
<td>87.8</td>
<td>88.7</td>
</tr>
<tr>
<td>4+6+8+10</td>
<td>85.2</td>
<td>87.5</td>
<td>89.6</td>
<td>90.9</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>$D$</th>
<th>32</th>
<th>64</th>
<th>128</th>
<th>256</th>
</tr>
</thead>
<tbody>
<tr>
<td>FV</td>
<td>88.3</td>
<td>90.0</td>
<td>91.2</td>
<td>91.8</td>
</tr>
<tr>
<td>VLAD</td>
<td>85.2</td>
<td>87.5</td>
<td>89.6</td>
<td>90.9</td>
</tr>
<tr>
<td>VLAT</td>
<td>81.5</td>
<td>83.7</td>
<td>85.8</td>
<td>87.2</td>
</tr>
<tr>
<td>FV</td>
<td>84.3</td>
<td>85.6</td>
<td>87.2</td>
<td>88.2</td>
</tr>
<tr>
<td>VLAD</td>
<td>80.0</td>
<td>82.5</td>
<td>83.7</td>
<td>85.4</td>
</tr>
<tr>
<td>VLAT</td>
<td>87.5</td>
<td>88.4</td>
<td>88.1</td>
<td>88.9</td>
</tr>
</tbody>
</table>
We carry out experiments to analyze the benefit of each new visual feature, as well as related normalization techniques. We present in Fig. 2 these results according to the size of features, using HOG and RGB descriptors, and dictionary sizes \( D = 32 \) and \( D = 64 \). If we focus on features with 1,000 dimensions, we can see that performance over 92% can be obtained with FV and VLAD methods. This can be compared to the results using BoW [1], which are from 77% to 82% with the same number of dimensions. Finally, we compare our results to the state of the art in Table IV. As we can see, model deviation approaches clearly outperform their methods by a fair margin of at least 9%.

### IV. Conclusion

In this paper, we presented recent methods for computing visual features, as well as related normalization techniques. We carried out experiments to analyse the benefit of each new component on the UC Merced land use dataset [1]. Overall, all considered method (FV, VLAD and VLAT) showed high classification accuracies. Furthermore, a high accuracy can be obtained with small visual dictionaries (64 words), especially for FV and VLAT features. Descriptor space normalization using PCA-based methods improve the results, but also allows effective combination of different descriptors types (HOG and RGB in our experiments). Finally, when compared to BoW methods, these methods lead to large performance improvement, as it is seen for generalist image categorization. A new component that will be worth investigating in the future is the compression of visual features for large-scale indexing, and their evaluation on very large land-use datasets.

### References


### TABLE III. CLASSIFICATION ACCURACY (%) FOR HOG AND RGB DESCRIPTORS AND FOR VARIOUS COMPRESSION RATIOS OF DESCRIPTORS WITH FV, VLAD, VLAT FEATURES (\( D = 64 \)).

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Accuracy (%)</td>
<td>76.81</td>
<td>72.52</td>
<td>77.71</td>
<td>76.71</td>
<td>81.19</td>
<td>76.91</td>
<td>94.3</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Accuracy (%)</td>
<td>73.14</td>
<td>76.05</td>
<td>77.38</td>
<td>81.10</td>
<td>84.86</td>
<td>85.48</td>
<td>93.8</td>
</tr>
</tbody>
</table>

### TABLE IV. COMPARISON TO STATE-OF-THE-ART RESULTS.

![Classification accuracy (%) as function of dimensionality reduction](image-url)


