Efficient Bag of Scenes Analysis for Image Categorization

Sébastien Paris[1], Xanadu Halkias[2] and Hervé Glotin[2,3]
[1]*DYNI team, LSIS CNRS UMR 7296, Aix-Marseille University, Aix-en-Provence, France*
[2]*DYNI team, LSIS CNRS UMR 7296, Université Sud Toulon-Var, Toulon, France*
[3]*Institut Universitaire de France, Paris, France*

Keywords: Image Categorization, Scenes Categorization, Fine-grained Visual Categorization, Non-parametric Local Patterns, Multi-scale LBP/LTP, Dictionary Learning, Sparse Coding, LASSO, Max-pooling, SPM, Linear SVM.

Abstract: In this paper, we address the general problem of image/object categorization with a novel approach referred to as *Bag-of-Scenes* (BoS).Our approach is efficient for low semantic applications such as texture classification as well as for higher semantic tasks such as natural scenes recognition or fine-grained visual categorization (FGVC). It is based on the widely used combination of i) Sparse coding (Sc), ii) Max-pooling and iii) Spatial Pyramid Matching (SPM) techniques applied to histograms of multi-scale Local Binary/Ternary Patterns (LBP/LTP) and its improved variants. This approach can be considered as a two-layer hierarchical architecture: the first layer encodes the local spatial patch structure *via* histograms of LBP/LTP while the second encodes the relationships between pre-analyzed LBP/LTP-scenes/objects. Our method outperforms SIFT-based approaches using Sc techniques and can be trained efficiently with a simple linear SVM.

1 INTRODUCTION

Image categorization[1] consists of assigning a unique label with a generally high-level semantic value to an image while FGVC refers to the task of classifying objects that belong to the same basic-level class. Both have long been a challenging problem area in computer vision, biomonitoring and robotics and can mainly be viewed as belonging to the broader supervised classification framework. In scene categorization, the difficulty of the task can be partly explained by the high-dimensional input space of the images as well as the high-level semantic visual concepts that lead to large intra-class variation. For object recognition more specifically, the small aspect ratio (object'size *vs* image'size) can induce a high level of uninformative background pixels. A preliminary detection procedure is required to "home-in" the object in a Region of Interest (ROI) (Bosch et al., 2007; Larios et al., 2011).

The *direct* framework (see Fig.1) in vision systems consists of extracting directly from the images meaningful features (using shape/texture/similarity/ color information) in order to achieve the maximum generalization capacity during the classification stage. Examples of such popular features in computer vision and human cognition inspired models include GIST (Oliva and Torralba, 2001), HOG (Dalal and Triggs, 2005), Self-Similarity (Deselaers and Ferrari, 2010) and WLD (Chen et al., 2010).

Widely used in face detection (Fröba and Ernst, 2004; Wu et al., 2011), face recognition (Marcel et al., 2007; Zhang et al., 2007), texture classification (Sadat et al., 2011; Bianconi et al., 2012) and scene categorization (Wu and Rehg, 2008; Gao et al., 2010; Paris and Glotin, 2010; Zhang et al., 2010), Local Binary Pattern (LBP) (Ojala et al., 2002) and recent derivatives such as Local Ternary Pattern (LTP) (Zheng et al., 2010), Gabor-LBP (Zhang et al., 2009; Lee et al., 2010), Local Gradient Pattern (LGP) (Jun and Kim, 2012) or Local Quantized Pattern (LQP) (Hussain and Triggs, 2012) are efficient local micro-patterns that define competitive features achieving state-of-the-art performances.

LBP can be considered as a non-parametric local visual micro-pattern texture, encoding mainly contours and differential excitation information of the 8 neighbors surrounding a central pixel (Heikkilä et al., 2006; Huang et al., 2011). This process represents a contractive mapping from $\mathbb{R}^9 \mapsto \mathbb{N}_{2^8} \subset \mathbb{N}^+$ for

[1]Granded by COGNILEGO ANR 2010-CORD-013 and PEPS RUPTURE Scale Swarm Vision

Paris S., Halkias X. and Glotin H. (2013).
Efficient Bag of Scenes Analysis for Image Categorization.
In *Proceedings of the 2nd International Conference on Pattern Recognition Applications and Methods*, pages 335-344
DOI: 10.5220/0004198303350344

each local patch $p(\boldsymbol{x})$ centered in $\boldsymbol{x}$ ((Bianconi and Fernández, 2011) provide a theoretical study of LBP). The total number of different LBPs is relatively small and by construction is finite: from 256 up to 512 different patterns (if improved LBP is used).

LTP (Tan and Triggs, 2010) have been extended from LBP as a parametric approximation of a ternary pattern. Instead of mapping $\mathbb{R}^9 \mapsto \mathbb{N}_{38} \subset \mathbb{N}^+$, they proposed to split the ternary pattern into two binary patterns and concatenating the two associated histograms. In (Hussain and Triggs, 2012), they generalize local pattern with LQP by both increasing neighborhood range, number of neighbors and pattern cardinality leading to map $\mathbb{R}^9 \mapsto \mathbb{N}_{b^N} \subset \mathbb{N}^+$.

Histograms of LBP (HLBP) (respectively HLTP), which count the occurrence of each LBP (respectively LTP) in the scene, can easily capture general structures in the visual scene by integrating information in a ROI, while being less sensitive to local high frequency details. This property is important when the desire is to generalize visual concepts. As depicted in this work, it is advantageous to extend this analysis for several sizes of local ROIs using a spatial pyramid denoted by $\boldsymbol{\Lambda}$.

Recently, the alternative scheme of *Bag-of-Features* (BoF) has been employed in several computer vision tasks with wide success. It offers a deeper extraction of visual concepts and improves accuracy of computer vision systems. BoF image representation (Willamowski et al., 2004) and its SPM extension (Lazebnik et al., 2006) share the same idea as HLBP: counting the presence (or combination) of visual patterns in the scene. BoF contains at least three modules prior to the classification stage: (i) region selection for patch extraction; (ii) codebook/dictionary generation and feature quantization; (iii) frequency histogram based image representation with SPM. In general, SIFT/HOG patches (Lowe, 2009; Dalal and Triggs, 2005) are employed in the first module. These visual descriptors are then encoded, in an unsupervised manner, into a moderate sized dictionary using Vector Quantization (VQ) (Lazebnik et al., 2006) or sparse coding (Yang et al., 2009b). In (Wu and Rehg, 2009), Wu and *al* were first to introduce LBP (*via* CENTRIST) into BoF framework coupled with histogram intersection kernel (HIK).

At least two disadvantages can be addressed against the BoF framework, mainly concerning the second stage. Firstly, and more specifically for FGVC, the trained dictionaries don't have enough representative basis vectors for some (rare and detailed) local patches that are crucial for discriminativity. Secondly, during quantification/encoding a lot of important information can be lost (Boiman et al., 2008). For these reasons, dictionary-free approaches have been recently introduced. In (Yao and Bradski, 2012), they performed an efficient template matching coupled with a bagging classification procedure. In (Bo et al., 2010; Bo et al., 2011a), they bypass BoF with efficient but computationally expensive hierarchical kernel descriptors. In (Larios et al., 2011; Choi et al., 2012), they proposed patche's supervised learning (respectively supervised projection) with random forest (respectively with PLS).

In order to improve the encoding scheme, it has been shown that localized soft-assignement (Avila et al., 2011), local-constrained linear coding (LLC) (Oliveira et al., 2012), Fisher vectors (FV) (Perronnin et al., ; Krapac et al., 2011), orthogonal matching pursuit (OMP) (Bo et al., 2011b) or Sparse coding (Sc) (Yang et al., 2009b; Gao et al., 2010) can easily be plugged into the BoF framework as a replacement for VQ. Moreover, pooling techniques coupled with SPM (Lazebnik et al., 2006) can be effectively used as a replacement for the global histogram based image representation.

Our contributions in this paper are two-fold. We first re-introduce two multi-scale variants of the LBP operators and extend two novel multi-scale variants of the LTP operators (Tan and Triggs, 2010). Secondly, we propose to plug HLBP/HLTP into the Sc framework as a second analyzing layer and call this procedure *Bag-of-Scenes* (BoS). This new approach is efficient as well as for scene categorization, object recognition or FGVC. The novel features can be trained efficiently with simple large-scale linear SVM solver such as *Pegasos* (Shalev-Shwartz et al., 2007) or *LIBLINEAR* (Hsieh et al., 2008). BoS can be seen as a two layer Hierarchical BoF analysis: a first fast contractive low-dimension manifold encoder *via* HLBP/HLTP and a second inflating high-dimension encoder *via* Sc.

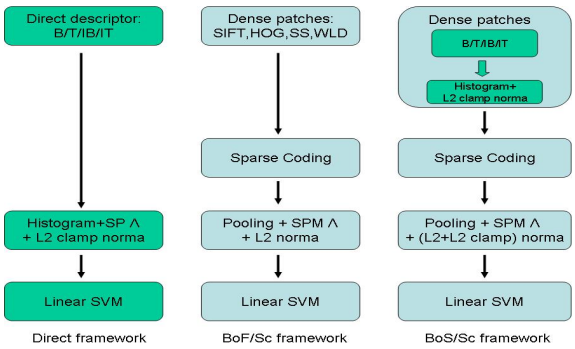

Figure 1: Comparison of the different frameworks. Left: *direct* framework, Middle: BoF/Sc framework, Right: Our proposed BoS/Sc framework.

2 HISTOGRAM OF MULTI-SCALE LOCAL PATTERNS

For an image/patch $\boldsymbol{I}$ $(n_y \times n_x)$, we present two existing multi-scale versions of the LBP operator, denoted by the B operator and for its *improved* variant by the IB operator. We also introduce two novel multi-scale versions of the LTP, denoted by the T operator and for its *improved* variant by the IT operator.

2.1 Multi-scale LBP/ILBP

Basically, operator B encodes the relationship between a central block of $(s \times s)$ pixels located in (y_c, x_c) with its 8 neighboring blocks (Liao et al., 2007), whereas operator IB adds a ninth bit encoding a term homogeneous to the differential excitation (see left Fig. 2). Both can be considered as a non-parametric local texture encoder for scale s. In order to capture information at different scales, the range analysis $s \in \mathcal{S}$, is typically set at $\mathcal{S} = [1,2,3,4]$ for this paper, where $S = Card(\mathcal{S})$. These two micro-codes are defined as follows[2]:

$$\begin{cases} B(y_c,x_c,s) &= \sum_{i=0}^{i=7} 2^i \mathbb{1}_{\{A_i \geq A_c\}} \\ IB(y_c,x_c,s) &= B(y_c,x_c,s) + 2^8 \mathbb{1}_{\left\{\sum_{i=0}^{7} A_i \geq 8Ac\right\}}. \end{cases} \quad (1)$$

For $\forall (y_c,x_c) \in \boldsymbol{R} \subset \boldsymbol{I}$, $B(y_c,x_c,s) \in \mathbb{N}_{2^8}$ and $IB(y_c,x_c,s) \in \mathbb{N}_{2^9}$ respectively.

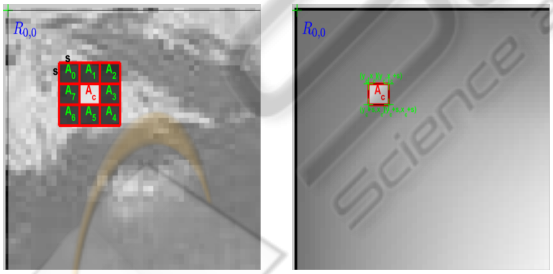

Figure 2: Left: $\boldsymbol{I}$ and $B(y_c,x_c,4)$ overlaid. Right: corresponding image integral $\boldsymbol{II}$ and the central block A_c. A_c can be efficiently computed with the 4 corner points.

2.2 Multi-scale LTP/ILTP

We introduce the multi-scale version of LTP and its improved variant. The idea behind LTP is to extend the LBP for $b = 3$ with the help of a single threshold parameter $t \in \mathbb{N}_{2^8}$. With the same neighborhood configuration with $N = 8$ (see left Fig. 2), a direct extension would conduct to have $3^8 = 6561$ different patterns. In (Tan and Triggs, 2010), they proposed to break the high dimensionality of the code by splitting the ternary code into two binary operators T_p and T_n such as:

$$\begin{cases} T_p(y_c,x_c,s;t) &= \sum_{i=0}^{i=7} 2^i \mathbb{1}_{\{\frac{1}{s^2}(A_i - A_c) \geq t\}} \\ T_n(y_c,x_c,s;t) &= \sum_{i=0}^{i=7} 2^i \mathbb{1}_{\{\frac{1}{s^2}(A_i - A_c) \leq -t\}}. \end{cases} \quad (2)$$

The improved multi-scale LTP operators (denoted IT_p and IT_n) are derived similarly from MSLBP by:

$$\begin{cases} IT_p(y_c,x_c,s;t) = T_p(y_c,x_c,s;t) + 2^8 \mathbb{1}_{\left\{\frac{1}{s^2}\left(\sum_{i=0}^{7} A_i - 8Ac\right) \geq t\right\}} \\ IT_n(y_c,x_c,s;t) = T_n(y_c,x_c,s;t) + 2^8 \mathbb{1}_{\left\{\frac{1}{s^2}\left(\sum_{i=0}^{7} A_i - 8Ac\right) \leq -t\right\}}. \end{cases} \quad (3)$$

Now, for $\forall (y_c,x_c) \in \boldsymbol{R} \subset \boldsymbol{I}$, both codes $\{T_p(y_c,x_c,s;t), T_n(y_c,x_c,s;t)\} \in \mathbb{N}_{2^8}$ while the improved version $\{IT_p(y_c,x_c,s;t), IT_n(y_c,x_c,s;t)\} \in \mathbb{N}_{2^9}$ respectively.

2.3 Integral Image for Fast Areas Computation

The different areas $\{A_i\}$ and A_c in eq.(1), eq.(2) and eq.(3) can be computed efficiently using the image integral technique (Viola and Jones, 2004). Let's define $\boldsymbol{II}$ the image integral of $\boldsymbol{I}$ by:

$$\boldsymbol{II}(y,x) \triangleq \sum_{y'=0}^{y'<y} \sum_{x'=0}^{x'<x} \boldsymbol{I}(y',x'). \quad (4)$$

Any square area $A(y,x,s) \in \boldsymbol{R}$ (see right Fig. 2) with upper-left corner located in (y,x) and side length s is the addition of only 4 values:

$$\begin{aligned} A(y,x,s) &= \boldsymbol{II}(y+s,x+s) + \boldsymbol{II}(y,x) \\ &\quad -(\boldsymbol{II}(y,x+s) + \boldsymbol{II}(y+s,x)). \end{aligned} \quad (5)$$

2.4 Histogram of Local Patterns

For all previously defined operators $op \in \{B, IB, T_p, T_n, IT_n, IT_p\}$, efficient features are obtained by counting occurrences of the j^{th} visual LBP/LTP at scale s in a ROI $\boldsymbol{R} \subseteq \boldsymbol{I}$:

$$z_{op}(\boldsymbol{R}, j, s) = \sum_{(x_c,y_c) \in \boldsymbol{R}} \mathbb{1}_{\{op(y_c,x_c,s)=j\}},$$

where $j = 0, \ldots, b-1$ is the j^{th} bin of the histogram and $b = \{256, 512, 256, 256, 512, 512\}$ for $op \in \{B, IB, T_p, T_n, IT_n, IT_p\}$ respectively.

[2] $\mathbb{1}_{\{x\}} = 1$ if event x is true, 0 otherwise.

Full histogram of LBP and variant its ILBP, denoted $\boldsymbol{z}_B$, $\boldsymbol{z}_{IB}$, are computed by:

$$\boldsymbol{z}_{op}(\boldsymbol{R},s) \triangleq [z_{op}(\boldsymbol{R},0,s),\ldots,z_{op}(\boldsymbol{R},b-1,s)], \quad (6)$$

with a total size for patches $d = b = \{256,512\}$ respectively.

For LTP, full histograms, denoted $\boldsymbol{z}_T$, $\boldsymbol{z}_{IT}$ are defined by:

$$\begin{aligned}\boldsymbol{z}_{op}(\boldsymbol{R},s) \quad \triangleq & \left[z_{op_p}(\boldsymbol{R},0,s),\ldots,z_{op_p}(\boldsymbol{R},b-1,s),\ldots,\right. \\ & \left.,\ldots,z_{op_n}(\boldsymbol{R},0,s),\ldots,z_{op_n}(\boldsymbol{R},b-1,s)\right],\end{aligned} \quad (7)$$

with a total size for patches $d = 2.b = \{512,1024\}$ respectively.

To end the patch extraction stage, regardless the type of histogram of local patterns used, a ℓ_2 clamped normalization procedure (ℓ_2 normalization followed by a saturation with the clamp value and again a ℓ_2 normalization) is performed on each histogram (clamp value = 0.2).

3 SPARSE CODING ON PATCHES OF MULTI-SCALE LOCAL PATTERNS

Following the same framework as in (Lazebnik et al., 2006; Yang et al., 2009b; Boureau et al., 2010a; Chatfield et al., 2011), we show here that the traditional BoF approach can be advantageously replaced by i) Sc, ii) max-pooling technique and iii) a simple linear SVM as a classifier since the produced features are mostly linearly separable (see Fig. 1 for synopsis).

3.1 Patches of HB/HIB/HT/HIT

Here, we replace the collection of usual SIFT patches densely sampled on a grid by our HB/HIB/HT/HIT patches $\boldsymbol{z}$ seen previously. Specifically, F patches of size $(m \times m)$ associated with ROI's $\{\boldsymbol{O}_k\}$ (possibly overlapping) are extracted for $k = 0,\ldots,F-1$ and $\forall s \in \mathcal{S}$ (see Fig. 3). For a faster computation for each scale s, the integral image $\boldsymbol{II}$ is first computed from $\boldsymbol{I}$.

For a complete dataset containing N images and $\forall s \in \mathcal{S}$, we obtain a collection of $P = TS$ patches $\boldsymbol{Z} \triangleq \{\boldsymbol{z}_i\}$, $i = 1,\ldots,P$, where $T = NF$. We define, the subset of patches $\boldsymbol{z}_i$ at scale s by $\boldsymbol{Z}(s) \subseteq \boldsymbol{Z}$ with T elements.

3.2 Sparse Coding Overview

In order to obtain highly discriminative visual features, a common procedure consists of encoding each patch $\boldsymbol{z}_i \in \boldsymbol{Z}(s)$ at scale s through an unsupervised trained dictionary $\boldsymbol{D} \triangleq [\boldsymbol{d}_1,\ldots,\boldsymbol{d}_K] \in \mathbb{R}^{b\times K}$, where K denotes the number of dictionary elements, and its corresponding weight vector $\boldsymbol{c}_i \in \mathbb{R}^K$. In the BoF framework, the vector $\boldsymbol{c}_i$ is assumed to have only one non-zero element:

$$\arg\min_{\boldsymbol{D},\boldsymbol{C}} \sum_{i=1}^{T} \|\boldsymbol{z}_i - \boldsymbol{D}\boldsymbol{c}_i\|_2^2 \quad s.t. \quad \|\boldsymbol{c}_i\|_{\ell_0} = 1, \quad (8)$$

where $\boldsymbol{C} \triangleq [\boldsymbol{c}_1,\ldots,\boldsymbol{c}_K]$ and $\|\bullet\|_{\ell_0}$ defines the pseudo zero-norm, where here only one element of $\boldsymbol{c}_i$ is non-zero. In eq. (8), under these constraints, $(\boldsymbol{D},\boldsymbol{C})$ can be optimized jointly by a Kmeans algorithm for example.

In the Sc approach, in order to i) reduce the quantization error and ii) to have a more accurate representation of the patches, each vector $\boldsymbol{z}_i$ is now expressed as a linear combination of a few vectors of the dictionary $\boldsymbol{D}$ and not only by a single one. Imposing the exact number of non-zero elements in $\boldsymbol{c}_i$ (sparsity level) involves a non-convex optimization (Mairal et al., 2009). In general, it is preferred to relax this constraint and to use instead an ℓ_1 penalty which also involves sparsity. The problem is then reformulated using the following equation:

$$\arg\min_{\boldsymbol{D},\boldsymbol{C}} \sum_{i=1}^{T} \|\boldsymbol{z}_i - \boldsymbol{D}\boldsymbol{c}_i\|_2^2 + \beta\|\boldsymbol{c}_i\|_{\ell_1} \quad s.t. \quad \|\boldsymbol{c}_i\|_{\ell_1} = 1, \quad (9)$$

where the sparsity in controlled by the parameter β. The last equation is not jointly convex in $(\boldsymbol{D},\boldsymbol{C})$ and a common procedure consists of optimizing alternatively $\boldsymbol{D}$ given $\boldsymbol{C}$ by a block coordinate descent and then $\boldsymbol{C}$ given $\boldsymbol{D}$ by a LASSO procedure (Tibshirani, 1996). At the end of the process, for each scale $s \in \mathcal{S}$, a trained dictionary $\widehat{\boldsymbol{D}}(s)$ is obtained.

3.3 Spatial Pyramidal Matching and Max Pooling

For an image $\boldsymbol{I}$ and given a trained dictionary $\widehat{\boldsymbol{D}}(s)$ for a type of code at scale s, F sparse vectors $\{\boldsymbol{c}_k(s)\}$ are computed by a LASSO algorithm. The final efficient descriptor $\boldsymbol{x}(s) \triangleq [x^0(s),\ldots,x^{K-1}(s)] \in \mathbb{R}^K$ is obtained by the following max-pooling procedure (Yang et al., 2009b; Boureau et al., 2010b):

$$x^j(s) \triangleq \max_{k|\boldsymbol{O}_k\in\boldsymbol{R}} (|c_k^j(s)|),\ j = 0,\ldots,K-1, \quad (10)$$

where each element of $\boldsymbol{x}(s)$ represents the max-response of the absolute value of sparse codes belonging to the ROI $\boldsymbol{R}$. In order to improve accuracy, a spatial pyramidal matching procedure helps to perform a more robust local analysis. The spatial pyramid $\boldsymbol{\Lambda}$ has $V = \sum_{l=0}^{L-1} V_l$ ROIs $\{\boldsymbol{R}_{l,v}\}$ with $l = 0,\ldots,L-1$,

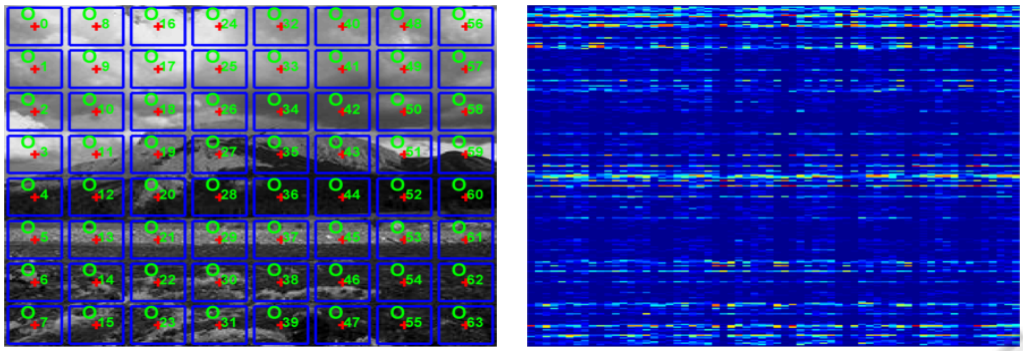

Figure 3: Example Left: ROI's $\{\boldsymbol{O}_k\}, k = 0,\ldots,F-1$ of extracted patches used to compute HB. Right: associated normalized histograms $\{\boldsymbol{z}_B(\boldsymbol{O}_k)\}$, one per column.

$v = 0,\ldots,V_l - 1$ (see Fig. 4 for an example). The quantity $\boldsymbol{z}^j_{l,v}(s)$ for each ROI $\boldsymbol{R}_{l,v}$ is computed by:

$$x^j_{l,v}(s) \triangleq \max_{k|\boldsymbol{O}_k \in \boldsymbol{R}_{l,v}} (|c^j_k(s)|),\ j = 0,\ldots,K-1. \quad (11)$$

We reinforce our model by an important normalization step, improving considerably accuracy, consists of the ℓ_2 normalization of all vectors $\{\boldsymbol{x}_{l,v}(s)\}$, $v = 0,\ldots,V_l - 1, s \in \mathcal{S}$, *i.e.* belonging to the same pyramidal layer l. This step is also very important and often hidden in the existing literature.

The final descriptor $\boldsymbol{x}(\boldsymbol{\Lambda})$ will be defined by the weighted concatenation of all the $\boldsymbol{x}_{l,v}(s)$ vectors, *i.e.* $\boldsymbol{x}(\boldsymbol{\Lambda}) \triangleq \{\lambda_l \boldsymbol{x}_{l,v}(s)\}$, $l = 0,\ldots,L-1, v = 0,\ldots,V_l - 1$ and $\forall s \in \mathcal{S}$. The total size of the feature vector $\boldsymbol{x}(\boldsymbol{\Lambda})$ is $d = K.V.S$, where typically in our simulations, we fixed $K = \{1024, 2048\}$, $V = \{10, 21, 26\}$ and $S = 4$. A final ℓ_2 clamped normalization step is performed on the full vector $\boldsymbol{x}(\boldsymbol{\Lambda})$.

4 LINEAR SVM FOR SUPERVISED TRAINING

Let's assume available a training data set $\{\boldsymbol{x}_i(\boldsymbol{\Lambda}), y_i\}_{i=1}^N$, where $\boldsymbol{x}_i(\boldsymbol{\Lambda}) \in \mathbb{R}^d$ is one of four previously defined features and $y_i \in \{1,\ldots,M\}$, where M is the number of classes. As in (Yang et al., 2009b; Boureau et al., 2010a), we will use a simple large-scale linear SVM such as LIBLINEAR (Hsieh et al., 2008) with the 1-vs-all multi-class strategy. The associated binary unconstrained convex optimization problem to solve is:

$$\min_{\boldsymbol{w}} \left\{ \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} + C\sum_{i=1}^{N} \max\left(1 - y_i \boldsymbol{w}^T \boldsymbol{x}_i, 0\right)^2 \right\}, \quad (12)$$

where the parameter C controls the generalization error and is tuned on a specific validation set. LIBLINEAR converges to a solution linearly in $O(dN)$ compared to $O(dN_{sv}^2)$ in the worst case for classic SVM where $N_{sv} \leq N$ defines the number of support vectors.

5 EXPERIMENTAL RESULTS

We test our BoS framework on Scene-15 (Lazebnik et al., 2006), UIUC-Sport (Li, 2007), Caltech101 (Fei-Fei et al., 2007), USCD-Birds200 (Welinder et al., 2010) and Stanford-Dogs120 datasets(Khosla et al., 2011a).

We define our SPM matrix $\boldsymbol{\Lambda}$ with L levels such as $\boldsymbol{\Lambda} \triangleq [\boldsymbol{r}_y, \boldsymbol{r}_x, \boldsymbol{d}_y, \boldsymbol{d}_x, \boldsymbol{\lambda}]$. $\boldsymbol{\Lambda}$ is matrix of size $(L \times 5)$. For a level $l \in \{0,\ldots,L-1\}$, the image $\boldsymbol{I}$, with size $(n_y \times n_x)$, is divided into potentially overlapping sub-windows $\boldsymbol{R}_{l,v}$ of size $(h_l \times w_l)$. All these windows are sharing the same associated weight λ_l. In our implementation, $h_l \triangleq \lfloor n_y.r_{y,l} \rfloor$ and $w_l \triangleq \lfloor n_x.r_{x,l} \rfloor$ where $r_{y,l}$, $r_{x,l}$ and λ_l are the l^{th} element of vectors $\boldsymbol{r}_y$, $\boldsymbol{r}_x$ and $\boldsymbol{\lambda}$ respectively. Sub-window shifts in $x-y$ axis are defined by integers $\delta_{y,l} \triangleq \lfloor n_y.d_{y,l} \rfloor$ and $\delta_{x,l} \triangleq \lfloor n_x.d_{x,l} \rfloor$ where $d_{y,l}$ and $d_{x,l}$ are elements of $\boldsymbol{d}_y$ and $\boldsymbol{d}_x$ respectively. Overlapping can be performed if $d_{y,l} \leq r_{y,l}$ and/or $d_{x,l} \leq r_{x,l}$. The total number of sub-windows is equal to

$$V = \sum_{l=0}^{L-1} V_l = \sum_{l=0}^{L-1} \lfloor \frac{(1 - r_{y,l})}{d_{y,l}} + 1 \rfloor . \lfloor \frac{(1 - r_{x,l})}{d_{x,l}} + 1 \rfloor . \quad (13)$$

For all dataset used, we used SIFT patches with block size (16×16) pixels and (26×26) pixels for ours HB/HIB/HT/HIT respectively. For SIFT/HB/HIB/HT/HIT, we extract $F = 35.35 = 1225$ patches per scale. For both dictionary learning and sparse codes computation, we fix $\beta = 0.2$ and $N_{ite} = 50$ iterations to train dictionaries. We uses our own modified version of the SPAMS toolbox (Mairal et al., 2009). Finally, we performed 10 cross-validation to

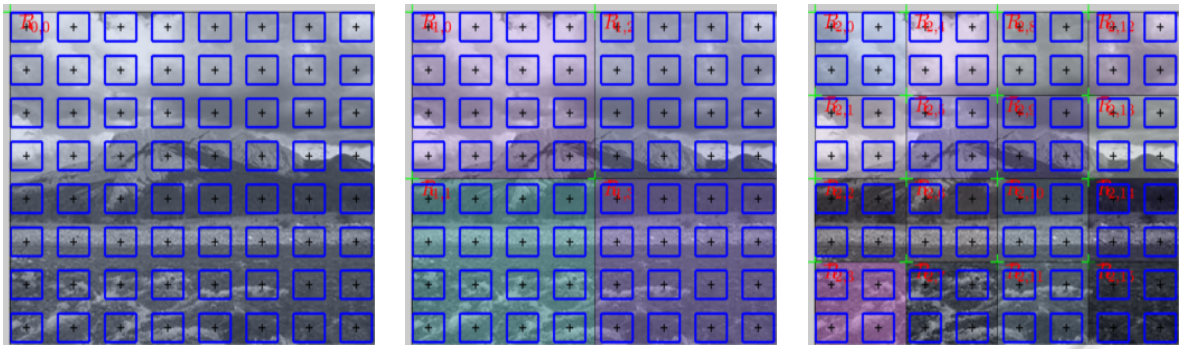

Figure 4: Example of SPM $\mathbf{\Lambda}$ with $L=3$, $F=8\times 8$ and $V=1+4+16$. The F ROIs $\{\boldsymbol{O}_k\}$, $k=0,\ldots,F-1$ associated with each patch $\boldsymbol{z}_k$ are represented by blue squares. Sparse codes $\boldsymbol{c}_k$ are computed for each ROI $\boldsymbol{O}_k$. Upper-left corner of each max-pooling window $\boldsymbol{R}_{l,v}$ taking $\{64,16,4\}$ $\boldsymbol{c}_k$ is indicated with a green cross. Left: $\boldsymbol{R}_{0,0}=\boldsymbol{I}$ for $l=0$. Middle: $\{\boldsymbol{R}_{1,v}\}$, $v=0,\ldots,3$ for $l=1$. Right $\{\boldsymbol{R}_{2,v}\}$, $v=0,\ldots,15$ for $l=2$.

compute the average overall accuracy and its standard deviation using the LIBLINEAR solver and fixing parameter $C=15$.

5.1 Scene-15 Dataset

The Scene-15 dataset contains a total of 4485 images in grey color assigned to $M=15$ categories. The number of images in each category is ranging from 200 to 400. 100 images per class are used to train, the rest for testing. For this dataset, we define $\mathbf{\Lambda}=\begin{bmatrix}1 & 1 & 1 & 1 & 1\\ \frac{1}{3} & \frac{1}{3} & \frac{1}{6} & \frac{1}{6} & 1\end{bmatrix}$, *i.e.* a two layer spatial pyramid dividing image in third and an overlapping of 50% representing a total of $1+25=26$ ROIs. For HT and HIT patches, we fix $t=1$. We select 15000 patches per class (a total of 225000 patches) to train dictionaries *via* Sc. In Fig. 5, we plot accuracy versus the number of words K in the dictionary training. With our particular choice of $\mathbf{\Lambda}$ and for one unique scale, we retrieved results comparable to (Yang et al., 2009b), *i.e.* 80.28% *vs.* 81.24% for our implementation. Whatever, the number of scale used and the type of patch, our BoS framework outperforms the SIFT-ScSPM approach. In Tab. 1, we compare our results with the state-of-the-art for this dataset (with $S=4$ scales). The best performance is actually obtained with the SIFT-LScSPM involving a more sophisticated dictionary training through the Laplacian sparse coding. The latter is very time and memory consuming[3] but is improving results with normal SIFT patches from $80.28\%\pm 0.93$ with simple Sc to $89.75\%\pm 0.5$ with LSc. The second best result is obtained with spatial FV following by the kernel descriptors. For FV, they reduced SIFT to 64 dimension (total size equal to $K(1+2.d)=12800$) and used a multi-class logistic regression. It is also worth noting that KDES-EKM uses a concatenation of 3 descriptors coupled with an efficient feature mapping (KDES-A+LSVM got $81.9\%\pm 0.60$ for a fair comparison). However, our results with a single HIT patch and a simple linear SVM are very close. More, if FV or LSc would be used, one can expect better results.

[3]LSc requiers to store sparse codes of the template set, *i.e*, a sparse matrix $(K\times N_{template})$.

Table 1: Recognition rate (and standard deviation) for Scene-15 dataset.

Algorithms	Accuracy ± Std
SIFT-ScSPM ($K=1024$) (Yang et al., 2009b)	$80.28\%\pm 0.93$
SIFT-MidLevel ($K=2048$) (Boureau et al., 2010a)	$84.20\%\pm 0.30$
SIFT-LScSPM ($K=1024$) (Gao et al., 2010)	$\mathbf{89.75\%\pm 0.50}$
KDES-EKM ($K=1000$) (Bo et al., 2010)	*86.70%*
PCASIFT-SFV ($K=100$) (Krapac et al., 2011)	**88.20%**
SIFT-DITC ($K=1000$) (Elfiky et al., 2012)	85.4%
SIFT-ScSPM ($K=1024$, our implementation)	$81.24\%\pm 0.73$
HB-ScSPM ($K=2048$, our work)	$86.04\%\pm 0.36$
HIB-ScSPM ($K=2048$, our work)	$86.45\%\pm 0.44$
HT-ScSPM ($K=2048$, our work)	$86.24\%\pm 0.43$
HIT-ScSPM ($K=2048$, our work)	$\mathbf{86.53\%\pm 0.37}$

5.2 UIUC-sport Dataset

The UIUC-sport dataset contains a total of 1579 images assigned to $M=8$ categories. 60 images per class are used to train, 70 for testing. For this dataset, we define $\mathbf{\Lambda}=\begin{bmatrix}1 & 1 & 1 & 1 & 1\\ \frac{1}{2} & \frac{1}{2} & \frac{1}{4} & \frac{1}{4} & 1\end{bmatrix}$ representing a total of $1+9=10$ ROIs for SPM. Color (R,G,B) information channels are used, sampling patches and training dictionaries on each of them. For HT and HIT patches, we fix $t=5$. We select 30000 patches per class (a total of 240000 patches) to train dictionaries *via* Sc. In Fig. 6, we plot accuracy *vs.* K. No-

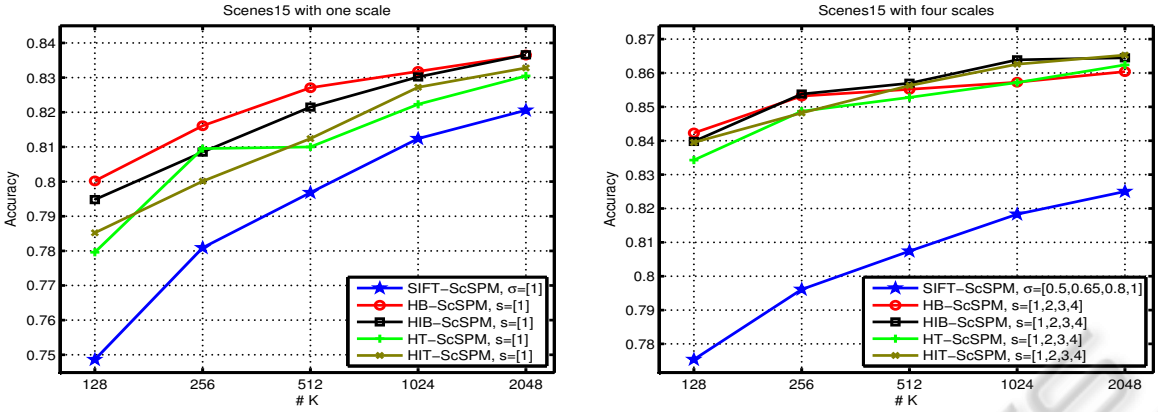

Figure 5: Results for Scenes 15. Left: one scale are used for all kind of patches. Right: four scales are used for all kind of patches.

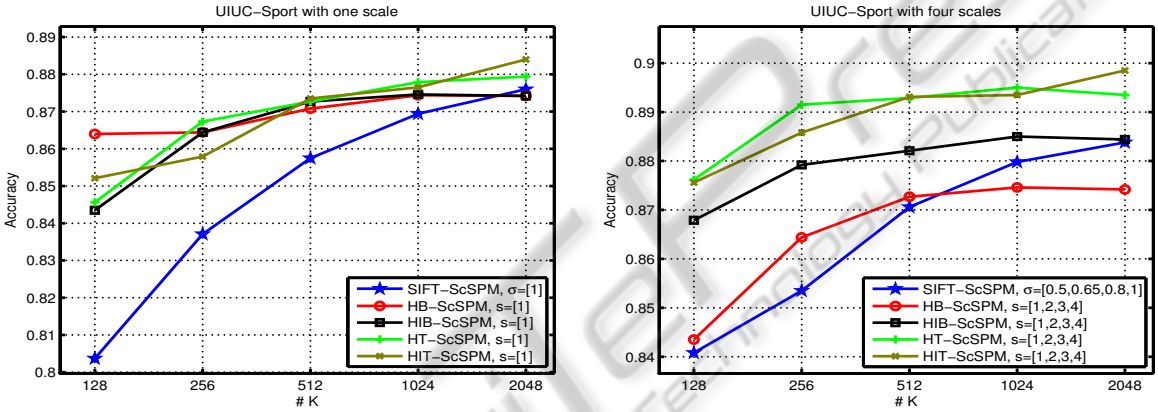

Figure 6: Results for UIUC-Sport. Left: one scale are used for all kind of patches. Right: four scales are used for all kind of patches.

tice, that our implementation of SIFT-ScSPM outperforms results from (Yang et al., 2009b). Our choice of $\mathbf{\Lambda}$, color information used in training and our specific normalization procedure may explain these improved results. We can also notice, especially for a small dictionary size, that our BoS framework is far superior to SIFT-ScSPM. In Tab. 2, we compare our results with the state-of-the-art (with $S = 4$ scales). To our best of knowledge, our BoS framework, with HIT patch, obtains the state-of-the-art performances with **89.85**% of overall accuracy.

5.3 Caltech101 Dataset

The Caltech101 dataset contains a total of 9144 images assigned to $M = 102$ categories. 30 images per class are used to train, the rest for testing. For this dataset, we define $\mathbf{\Lambda} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ \frac{1}{3} & \frac{1}{3} & \frac{1}{6} & \frac{1}{6} & 1 \end{bmatrix}$. We extract 2000 HIT patches per class (a total of 204000 patches) for $S = 4$ scales to train dictionaries *via* Sc.

Table 2: Recognition rate (and standard deviation) for UIUC-Sport dataset.

Algorithms	Accuracy ± Std
SIFT-ScSPM ($K = 1024$) (Yang et al., 2009b)	82.70% ± 1.50
SIFT-LScSPM ($K = 1024$) (Gao et al., 2010)	85.30% ± 0.31
SIFT-HOMP ($K = 2 \times 1024$) (Bo et al., 2011b)	85.70% ± 1.30
SIFT-ScSPM ($K = 1024$, our implementation)	87.98% ± 1.08
HB-ScSPM ($K = 2048$, our work)	87.42%±1.27
HIB-ScSPM ($K = 2048$, our work)	88.44%±1.25
HT-ScSPM ($K = 2048$, our work)	89.35%±1.42
HIT-ScSPM ($K = 2048$, our work)	**89.85**% ± **1.28**

In Tab. 3, we compare our results with the state-of-the-art. We separate methods using more sophisticated approaches such as prior detection to localize more precisely objects or using complex supervised segmentation with methods classifying directly images. To the best of our knowledge, we have the highest recognition rate (**81.05**%) for a unique feature coupled with a simple linear SVM. With a medium dictionary size ($K = 1024$), we are competitive with sophisticated and time-consuming methods using su-

Table 3: Recognition rate (and standard deviation) for Caltech101 dataset.

Methods	Algorithms	Accuracy ± Std (15 Train)	Accuracy ± Std (30 Train)
Graph Matching + SVM.	MLMRF+Curv. Expen. (Duchenne et al., 2011)	**75.30**% ± **0.70**	80.30% ± 1.20
Detec. + Mult Non-Lin Ker.	Multiway-SVM (Bosch et al., 2007)	-	81.30%
Superv. Segm+Classif	Subcat. Relevances (Todorovic and Ahuja, 2008)	72.00%	82.00%
Superv. Segm+Classif+Non-Lin Ker	SvcSegm (Li et al., 2010)	72.60%	79.20%
Superv. Segm+Regress+Non-Lin Ker	SvrSegm (Li et al., 2010)	74.70%	82.30%
Classif+MKL	GS-MKL (Yang et al., 2009a)	73.20%	**84.30**%
Classif+Lin Ker	SIFT-Multiway ($K = 1024$) (Boureau et al., 2011)	-	77.30% ± 0.60
Classif+Lin Ker	SIFT-CDBN ($K = 4096$) (Sohn et al., 2011)	71.30%	77.80%
Classif+Non-Lin Ker	SIFT-LaRank ($K = 4096$) (Oliveira et al., 2012)	73.09% ± 0.77	80.02% ± 0.36
Classif+Lin Ker	HT-ScSPM ($K = 1024$, our work)	**74.24**% ± **0.69**	**81.05**% ± **0.43**
Classif+Lin Ker	HT-ScSPM ($K = 2048$, our work)	73.92% ± 0.81	80.90% ± 0.38
Classif+Lin Ker	HIT-ScSPM ($K = 1024$, our work)	73.23% ± 0.69	80.51% ± 0.46
Classif+Lin Ker	HIT-ScSPM ($K = 2048$, our work)	72.54% ± 0.70	80.27% ± 0.44

pervised segmentation, graph matching or complex MKL.

5.4 USCD-Birds200 Dataset

The USCD-Birds200 dataset is containing a total of 6033 images assigned to $M = 200$ categories. We crop all images with the provided bounding-box ground-truth. 15 images per class are used to train, the rest for testing. This dataset represents a challenging FGVC task, where categorization must exploits details difference between species. We particularize $\mathbf{\Lambda} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ \frac{1}{3} & \frac{1}{3} & \frac{1}{6} & \frac{1}{6} & 1 \end{bmatrix}$. Color (R,G,B) information channels are used, sampling patches and training dictionaries on each of them. For the HIT patches, we fix $t = 5$. We select 2000 patches per class (a total of 400000 patches) for $S = 4$ scales to train dictionaries *via* Sc. In Tab. 4, we compare our results with the state-of-the-art. To our best of knowledge, our BoS framework, with HIT patch, obtains the state-of-the-art performances with **27.93**% of overall accuracy, outperforming dictionary-free methods.

Table 4: Recognition rate and standard deviation on the USCD-Birds200 dataset.

Algorithms	Accuracy ± Std
BiCOS-MT (Chai et al., 2011)	16.20%
Discri. Decision Trees + RF (Yao et al., 2011)	19.20%
Mult.-Cue+DITC ($K = 5000$) (Khan et al., 2011)	22.40%
HIT-ScSPM ($K = 1024$, our work)	**27.93**% ± **1.16**

5.5 Stanford-Dogs120 Dataset

The Stanford-Dogs120 dataset is containing a total of 20580 images assigned to $M = 120$ categories. We crop all images with the provided bounding-box ground-truth. 100 images per class are used to train, the rest for testing (we use the provided train/test set). This dataset represents also a challenging FGVC task. We particularize $\mathbf{\Lambda} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{4} & \frac{1}{4} & 1 \end{bmatrix}$. Color (R,G,B) information channels are used, sampling patches and training dictionaries on each of them. For the HIT patches, we fix $t = 5$. We select 2000 patches per class (a total of 240000 patches) for $S = 3$ scales ($\mathcal{S} = \{1,2,3\}$) to train dictionaries *via* Sc. In Tab. 5, we compare our results with the state-of-the-art. To our best of knowledge, our BoS framework, with HIT patch, obtains the state-of-the-art performances with **36.36**% of overall accuracy with a unique descriptor and linear SVM. A simple late fusion of SIFT-ScSPM with HIT-ScSPM (product of $p(y = 1|\boldsymbol{x})$) gives a score of **40.03**%.

Table 5: Recognition rate and standard deviation on the Stanford-Dogs120 dataset.

Algorithms	Accuracy ± Std
SIFT-ScSPM (Khosla et al., 2011b)	26.10%
SIFT-ScSPM ($K = 2048$, our implementation)	32.05%
HIT-ScSPM ($K = 2048$, our work)	**36.36**%
SIFT-ScSPM+HIT-ScSPM ($K = 2048$, our work)	**40.03**%

6 CONCLUSIONS AND PERSPECTIVES

We have presented in this article the 2-layer BoS architecture mixing HB/HIB/HT/HIT as a fast local textures encoder for the first layer and Sc as scenes encoder for the second. This first hand-graft layer can advantageously replace complex hierarchical feature extractors such as Deep Belief Networks and the patch extraction are even faster than SIFT ones, thanks to the integral image technique. Achieved performances outperform state-of-the-art results with

a simple linear SVM as well for object recognition tasks as for FGVC ones.

As potential future works, many perspectives can be investigated. For example, complementary patch, multi-scale variants of LPQ could be coupled with our HB/HIB/HT/HIT approach, in order train a unique dictionary with these fused patches. Higher dimension local pattern can be also associated with the Sc framework such those proposed by (Hussain and Triggs, 2012). Finally, experimenting with LSc (Gao et al., 2010) or FV (Krapac et al., 2011) should improve the encoding part of the pipeline, while supervised pooling techniques (Jia et al., 2011) will surely also improve results.

REFERENCES

Avila, S. E. F., Thome, N., Cord, M., Valle, E., and de Albuquerque Araújo, A. (2011). Bossa: Extended bow formalism for image classification. In *ICIP' 11*.

Bianconi, F. and Fernández, A. (2011). On the occurrence probability of local binary patterns: A theoretical study. *Journal of Mathematical Imaging and Vision*, 40(3):259–268.

Bianconi, F., González, E., Fernández, A., and Saetta, S. A. (2012). Automatic classification of granite tiles through colour and texture features. *Expert Syst. Appl.*, 39(12):11212–11218.

Bo, L., Lai, K., Ren, X., and Fox, D. (2011a). Object recognition with hierarchical kernel descriptors. In *CVPR' 11*.

Bo, L., Ren, X., and Fox, D. (2010). Kernel descriptors for visual recognition. In *NIPS' 10*.

Bo, L., Ren, X., and Fox, D. (2011b). Hierarchical matching pursuit for image classification: Architecture and fast algorithms. In *NIPS' 11*, pages 2115–2123.

Boiman, O., Shechtman, E., and Irani, M. (2008). In defense of nearest-neighbor based image classification. In *CVPR' 08*.

Bosch, A., Zisserman, A., and Munoz, X. (2007). Image classification using random forests and ferns. In *ICCV' 07*.

Boureau, Y., Bach, F., LeCun, Y., and Ponce, J. (2010a). Learning mid-level features for recognition. In *CVPR' 10*.

Boureau, Y., Le Roux, N., Bach, F., Ponce, J., and LeCun, Y. (2011). Ask the locals: multi-way local pooling for image recognition. In *ICCV' 11*.

Boureau, Y., Ponce, J., and LeCun, Y. (2010b). A theoretical analysis of feature pooling in vision algorithms. In *ICML' 10*.

Chai, Y., Lempitsky, V. S., and Zisserman, A. (2011). Bicos: A bi-level co-segmentation method for image classification. In *ICCV' 11*.

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

Chen, J., Shan, S., He, C., Zhao, G., Pietikainen, M., Chen, X., and Gao, W. (2010). Wld: A robust local image descriptor. *IEEE Trans. PAMI*, 32(9).

Choi, J., Schwartz, W. R., Guo, H., and Davis, L. S. (2012). A complementary local feature descriptor for face identification. In *WACV' 12*.

Dalal, N. and Triggs, B. (2005). Histograms of oriented gradients for human detection. In *CVPR' 05*.

Deselaers, T. and Ferrari, V. (2010). Global and efficient self-similarity for object classification and detection. In *CVPR' 10*.

Duchenne, O., Joulin, A., and Ponce, J. (2011). A graph-matching kernel for object categorization. In *ICCV' 11*.

Elfiky, N. M., Khan, F. S., van de Weijer, J., and Gonzàlez, J. (2012). Discriminative compact pyramids for object and scene recognition. *Pattern Recognition*, 45(4):1627–1636.

Fei-Fei, L., Fergus, R., and Perona, P. (2007). Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. *Comput. Vis. Image Underst.*, 106(1):59–70.

Fröba, B. and Ernst, A. (2004). Face detection with the modified census transform. In *FGR' 04*.

Gao, S., Tsang, I. W.-H., Chia, L.-T., and Zhao, P. (2010). Local features are not lonely laplacian sparse coding for image classification. In *CVPR '10*.

Heikkilä, M., Pietikäinen, M., and Schmid, C. (2006). Description of interest regions with center-symmetric local binary patterns. In *CVGIP '06*.

Hsieh, C., Chang, K., Lin, C., and Keerthi, S. (2008). A dual coordinate descent method for large-scale linear svm.

Huang, D., Shan, C., Ardabilian, M., Wang, Y., and Chen, L. (2011). Local Binary Patterns and Its Application to Facial Image Analysis: A Survey. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, 41(4):1–17.

Hussain, S. u. and Triggs, W. (2012). Visual recognition using local quantized patterns. In *CVPR' 12*.

Jia, Y., Huang, C., and Darrell, T. (2011). Beyond Spatial Pyramids: Receptive Field Learning for Pooled Image Features. In *NIPS '11*.

Jun, B. and Kim, D. (2012). Robust face detection using local gradient patterns and evidence accumulation. *Pattern Recognition*, 45(9):3304–3316.

Khan, F. S., van de Weijer, J., Bagdanov, A. D., and Vanrell, M. (2011). Portmanteau vocabularies for multi-cue image representation. In *NIPS' 11*.

Khosla, A., Jayadevaprakash, N., Yao, B., and Fei-Fei, L. (2011a). Novel dataset for fine-grained image categorization. In *CVPR '11*.

Khosla, A., Jayadevaprakash, N., Yao, B., and Fei-Fei, L. (2011b). Novel dataset for fine-grained image categorization. In *First Workshop on Fine-Grained Visual Categorization, CVPR '11*.

Krapac, J., Verbeek, J., and Jurie, F. (2011). Modeling Spatial Layout with Fisher Vectors for Image Categorization. In *ICCV '11*.

Larios, N., Lin, J., Zhang, M., Lytle, D., Moldenke, a., Shapiro, L., and Dietterich, T. (2011). Stacked spatial-pyramid kernel: An object-class recognition method to combine scores from random trees. In *WACV' 11*.

Lazebnik, S., Schmid, C., and Ponce, J. (2006). Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *CVPR' 06*.

Lee, H., Chung, Y., Kim, J., and Park, D. (2010). Face image retrieval using sparse representation classifier with gabor-lbp histogram. In *WISA' 10*.

Li, F., Carreira, J., and Sminchisescu, C. (2010). Object recognition as ranking holistic figure-ground hypotheses. In *CVPR' 10*.

Li, L. (2007). What, where and who? classifying event by scene and object recognition. In *CVPR '07*.

Liao, S., Zhu, X., Lei, Z., Zhang, L., and Li, S. Z. (2007). Learning multi-scale block local binary patterns for face recognition. In *ICB*.

Lowe, D. G. (2009). Object recognition from local scale-invariant features. In *ICCV' 99*.

Mairal, J., Bach, F., Ponce, J., and Sapiro, G. (2009). Online dictionary learning for sparse coding. In *ICML '09*.

Marcel, S., Rodriguez, Y., and Heusch, G. (2007). On the recent use of local binary patterns for face authentication. *International Journal on Image and Video Processing Special Issue on Facial Image Processing*.

Ojala, T., Pietikäinen, M., and Mäenpää, T. (2002). Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Trans. PAMI*, 24(7).

Oliva, A. and Torralba, A. (2001). Modeling the shape of the scene: A holistic representation of the spatial envelope. *International Journal of Computer Vision*, 42.

Oliveira, G. L., Nascimento, E. R., Viera, A. W., and Campos, M. F. M. (2012). Sparse spatial coding: A novel approach for efficient and accurate object recognition. *ICRA' 12*.

Paris, S. and Glotin, H. (2010). Pyramidal multi-level features for the robot vision@icpr 2010 challenge. In *ICPR' 10*.

Perronnin, F., Sánchez, J., and Mensink, T. Improving the fisher kernel for large-scale image classification. In *ECCV' 10*.

Sadat, R. M. N., Teng, S. W., Lu, G., and Hasan, S. F. (2011). Texture classification using multimodal invariant local binary pattern. In *WACV '11*.

Shalev-Shwartz, S., Singer, Y., Srebro, N., and Cotter, A. (2007). Pegasos: Primal estimated sub-gradient solver for svm.

Sohn, K., Jung, D. Y., Lee, H., and Hero III, A. O. (2011). Efficient Learning of Sparse , Distributed , Convolutional Feature Representations for Object Recognition. *ICCV' 11*.

Tan, X. and Triggs, B. (2010). Enhanced local texture feature sets for face recognition under difficult lighting conditions. *Trans. Img. Proc.*, 19(6):1635–1650.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society (Series B)*, 58.

Todorovic, S. and Ahuja, N. (2008). Learning subcategory relevances for category recognition. In *CVPR' 08*.

Viola, P. and Jones, M. (2004). Robust real-time face detection. *International Journal of Computer Vision*, 57.

Welinder, P., Branson, S., Mita, T., Wah, C., Schroff, F., Belongie, S., and Perona, P. (2010). Caltech-UCSD Birds 200. Technical Report CNS-TR-2010-001, California Institute of Technology.

Willamowski, J., Arregui, D., Csurka, G., Dance, C. R., and Fan, L. (2004). Categorizing nine visual classes using local appearance descriptors. In *ICPR' 04*.

Wu, J., Geyer, C., and Rehg, J. M. (2011). Real-time human detection using contour cues. In *ICRA' 11*.

Wu, J. and Rehg, J. (2009). Beyond the euclidean distance: Creating effective visual codebooks using the histogram intersection kernel. In *ICCV' 09*.

Wu, J. and Rehg, J. M. (2008). Where am i: Place instance and category recognition using spatial pact. *CVPR' 2008*.

Yang, J., Li, Y., Tian, Y., Duan, L., and Gao, W. (2009a). Group-sensitive multiple kernel learning for object categorization. In *ICCV' 09*.

Yang, J., Yu, K., Gong, Y., and Huang, T. S. (2009b). Linear spatial pyramid matching using sparse coding for image classification. In *CVPR' 09*.

Yao, B. and Bradski, G. (2012). A Codebook-Free and Annotation-Free Approach for Fine-Grained Image Categorization. In *CVPR' 12*.

Yao, B., Khosla, A., and Li, F.-F. (2011). Combining randomization and discrimination for fine-grained image categorization. In *CVPR' 11*.

Zhang, B., Gao, Y., Zhao, S., and Liu, J. (2010). Local derivative pattern versus local binary pattern: Face recognition with high-order local pattern descriptor. *IEEE Trans. Img. Proc.*, 19(2).

Zhang, L., Chu, R., Xiang, S., Liao, S., and Li, S. Z. (2007). Face detection based on multi-block lbp representation. In *ICB' 07*.

Zhang, W., Shan, S., Qing, L., Chen, X., and Gao, W. (2009). Are gabor phases really useless for face recognition? *Pattern Anal. Appl.*, 12(3):301–307.

Zheng, Y., Shen, C., Hartley, R. I., and Huang, X. (2010). Effective pedestrian detection using center-symmetric local binary/trinary patterns. *CoRR*, abs/1009.0892.