Histograms of Pattern Sets for Image Classification and Object Recognition

Winn Voravuthikunchai, Bruno Crémilleux, Frédéric Jurie

▶ To cite this version:

Winn Voravuthikunchai, Bruno Crémilleux, Frédéric Jurie. Histograms of Pattern Sets for Image Classification and Object Recognition. IEEE Conference on Computer Vision and Pattern Recognition, Jun 2014, Columbus, Ohio., United States. pp.224-231. hal-00980894

HAL Id: hal-00980894

https://hal.science/hal-00980894

Submitted on 19 Apr 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.

Histograms of Pattern Sets for Image Classification and Object Recognition

Winn Voravuthikunchai, Bruno Crémilleux and Frédéric Jurie
Universitée de Caen Basse-Normandie – CNRS UMR 6072 – ENSICAEN
firstname.lastname@unicaen.fr

Abstract

This paper introduces a novel image representation capturing feature dependencies through the mining of meaningful combinations of visual features. This representation leads to a compact and discriminative encoding of images that can be used for image classification, object detection or object recognition. The method relies on (i) multiple random projections of the input space followed by local binarization of projected histograms encoded as sets of items, and (ii) the representation of images as Histograms of Pattern Sets (HoPS). The approach is validated on four publicly available datasets (Daimler Pedestrian, Oxford Flowers, KTH Texture and PASCAL VOC2007), allowing comparisons with many recent approaches. The proposed image representation reaches state-of-the-art performance on each one of these datasets.

1. Introduction

The representation of images, and more specifically the representation of the dependencies between visual features is a central question in computer vision. Interestingly, since a few years, pattern mining has been shown to be a promising avenue for discovering these dependencies *e.g.* [11, 12, 23, 33]. Pattern mining algorithms are used for extracting a subset of 'meaningful' groups of features (so-called *patterns*), such as groups of features which frequently appear together in the images of a given class. These algorithms use safe pruning strategies to extract patterns without having to evaluate all possible feature combinations, which is typically not tractable. In addition, since these algorithms use exhaustive search, the extracted patterns provide a fairly complete picture of the information content of the data.

Despite the use of pruning strategies, these mining algorithms are limited by the potential size of the search space which grows exponentially with the number of features. Most of the work in the literature addresses this problem by reducing the number of visual features and by down-scaling the problem to the mining of image subregions. However, large visual vocabularies are necessary to capture variations of object's appearance and obtaining good performance, while mining patterns in sub-regions cuts-off the possibility to capture dependencies between distant features.

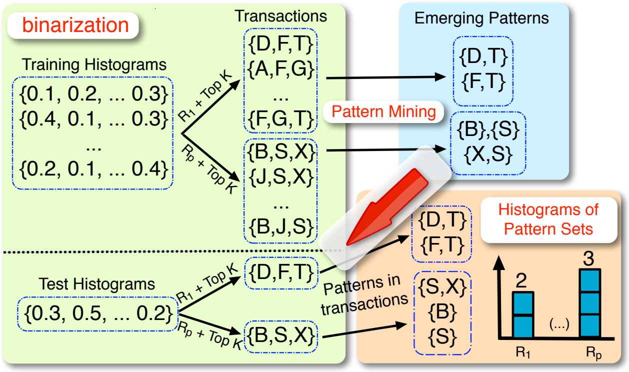


Figure 1: Overview of the approach: image histograms are tranformed into lists of binary items by projecting them into low-dimensional spaces (*e.g.* $R_1, ..., R_p$) and by top-K binarizing their projections (Section 3.1). Patterns are then mined from each set of training transactions. Images are finally encoded by counting the number of patterns found in each projection. The counts are concatenated to obtain the HoPS representation (Section 3.2).

Another issue raised by the use of mining algorithms is the production of binary visual features. Indeed, mining algorithms can process *items* (binary elements) that are included or not in the image. Such a binary representation is different from common image representations such as the Local Binary Patterns (LBP), the Histograms of Oriented Gradients (HOG), or the Bag-of-Words (BoW), which are real-valued histograms. Transforming these histogram-based representations into sets of meaningful items – with minimum information loss – is still an open issue.

Once the patterns are discovered, the following question raised by the use of mining algorithms for computer vision tasks is how to use them. In the literature, patterns are often used as new features. Due to the number of patterns (usually very high *i.e.* up to several millions), encoding each pattern as a single image feature in the learning process is often not feasible. A post processing step for selecting a smaller subset of patterns is usually applied *e.g.* [11]. However, finding the best subset is a nontrivial combinatorial problem for which greedy search is often the only feasible solution. Unfortunately, greedy search is not guaranteed to find a globally optimal solution, despite it's high computational cost. Another drawback of pattern selection is that some useful patterns can be inopportunely discarded.

This paper addresses the three previously mentioned issues (*i.e.* vocabulary size, binary visual items, and usage of mined patterns) by proposing a new framework for discovering sets of interesting groups of features from real-valued image representations (such as the LBP, HOG, BoW), and for efficiently constructing an distinctive image representation from the so-discovered patterns (Figure 1 gives an overview of the approach). It opens a new avenue for taking benefit of the integration of data mining techniques in computer vision applications.

In this context, our contribution is threefold. First, we propose a new method for transforming real-valued vectors (*e.g.* histogram-based image representations) into binary item transactions (set of binary visual elements), using random projections and adaptive thresholding. The proposed method addresses the two first mentioned problem regarding the dimensionality and the binarization loss. The patterns are mined from multiple sets of low-dimensional (projected) features and embed the relative order of initial real-valued representation (Section 3.1). Second, we introduce the concept of *Histograms of Pattern Sets* (HoPS); HoPS consist in finding interesting patterns (with the help of standard pattern mining algorithms), and using some relevant statistics of these patterns to represent images, hence condensing the information. Consequently, no pattern selection is required, which is an answer to the third issue (Section 3.2). Finally, we perform an extensive experimental validation on several different supervised image classification and object recognition/detection tasks. The proposed approach not only gives better performance than the baseline algorithms neglecting feature dependencies, but it also reaches state-of-the art performance on most of these tasks.

2. Related Work

Pattern mining allows the discovery of sets of features capturing local regularities and has been reported as being an efficient tool for discovering relevant dependencies between visual features (*e.g.* [11, 12, 22, 23, 25, 29, 33]). These methods differ in the way they transform images into sets of items which can be mined out.

Several works suggest extracting keypoints and representing local descriptors by visual words. The list of visual words contained in the image is then seen as a *transaction* [22, 23, 25]. However, such a coding is very sparse, losing potentially interesting information. Dense local features overcome this limitation, but in this case most of the visual words will appear at least once in the image, leading to transactions containing all possible items. Consequently, rather than encoding the whole image as a single transaction, [11, 29] divide the images into rectangular regions and produce one set of items per region. In addition to limiting the number of visual words per transactions, it allows the discovery of spatial relations between densely sampled quantized features. However, a drawback is that the combinations of features with high mutual separation distance cannot be discovered. Since the encoding has been reduced to local regions, a visual word can as, a consequence, appear multiple times in a single image region. Then, not representing its frequency would lead to an important binarization loss. To address this issue, [11] proposed a method using visual word frequencies when building items. This method improves the performance but exponentially increases the dimensionality of the binary features, which is not desirable, and also limits the size of the visual vocabulary that can be used. Reducing the size of the vocabulary is not a good option since vocabulary size is positively correlated with good performance. Differently, [32, 33] mine patterns from high-level semantic features for which smaller representational space are needed. However, such techniques can work only if the high-level features are correctly detected, which is an open question. In contrast, we show Section 3.1 how our binarization approach is able to preserve the information of the initial real-valued representation while providing a limited number of items.

The approaches using data mining for visual recognition also differ in the way they use mined patterns. Surprisingly, frequent patterns are often used [11, 23] even if contrast measures *e.g.* *emerging patterns* [21] would allow – by construction and more efficiently – to produce discriminative patterns suitable for classification tasks. As the set of frequent patterns can be very large (*e.g.* several millions) and redundant, a post-processing step is usually applied to select a tractable subset of discriminant patterns. [11] proposed an algorithm to select the frequent patterns that are discriminative, representative, and non-redundant. However, there is no efficient way to generate a set of patterns that can satisfy global constraints (*e.g.* cover all data while giving good prediction accuracy) [3] and all of the mentioned methods require costly or ad-hoc post-processing stages for selecting the patterns. In contrast, the proposed histogram of pattern sets does capture the discriminative power of the whole set of patterns and provides an efficient image representation for supervised tasks.

3. Method

As mentioned in the introduction, the proposed method has two steps: (i) real-valued histograms are turned into lists of binary items, Section 3.1 (Fig. 1, green block), and

(ii) image representations are computed from histograms of mined patterns, Section 3.2 (Fig. 1, red block). The discovering of patterns is done using standard mining algorithms.

In order to illustrate the presentation of the method, we use the bag-of-words (BoW) representation as a prototypical example *i.e.* an image I is represented as an histogram of visual words $h = (p(w_0|I), \ldots, p(w_d|I))$, where $L = \{w_0, \ldots, w_d\}$ is a vocabulary of size d. Any histogram-based representation, such as those used in our experiments, falls into this formalism and can be used in the same way.

3.1. From real-valued vectors to binary items

Pattern mining algorithms cannot be used to process high-dimensional real-valued representations as (i) pattern mining algorithms handle only binary items, and (ii) although pattern mining algorithms are very efficient in extracting patterns from a large number of transactions (*e.g.* millions), they can only tackle a moderate number of items, typically up to a few hundreds, depending on the density of the data. This is due to the search space which grows exponentially with the number of items. As image representations usually have several thousand components, mining patterns from such high-dimensional image data is not only slow but also memory consuming.

Multiple projections of the original representation into small dimensional spaces is one of the key ideas for addressing this issue. The rational for doing this is that (i) following the Johnson-Lindenstrauss lemma, representing high-dimensional data by multiple projections leads to good approximations of the data *e.g.* [15] and (ii) mining is more efficient when the dimensionality is low. In practice, the projection is done by simply randomly selecting p visual words from the original BoW. This can be seen as projecting the original d-dimensional data to a p-dimensional subspace where the projection matrix is obtained by randomly choosing the p basis vectors from the standard basis (more complex projections have been investigated, without improving the performance). Let R denote the $d \times p$ projection matrix, such that $h^p = h \times R$, $\forall h^p \in H^p$. H^p is denoted as the set of p-dimensional projected histograms. Once the histogram is projected, the bins whose values are among the top-*K* highest values are set to '1' while other bins are set to '0' (*e.g.* if the histograms are projected into a 10-d space and if $k = 4$, we will obtain 10-bit histogram signatures having 4 ones and 6 zeros). More formally, $h_j^{bp} = 1 \iff h_j^p \geq \tau$ where $\tau = h^p_{rank^k(h^p)}$, and $rank^k(h)$ returns the index of the histogram bin with the k^{th} highest value (Figure 2).

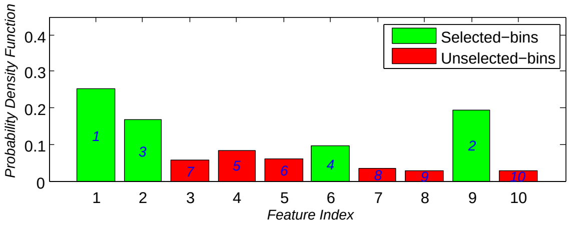


Figure 2: Top-K binarization. Numbers inside bins are ranking orders. Trans. {1,2,6,9} is produced in this case.

This process to obtain binary representation bears some similarity with Locality-Sensitive Hashing (LSH) [13]. However, the main difference is that in LSH, a fixed threshold is used instead of top-K. The motivation in using top-K binarization instead of a fixed threshold is to reduce and control the size of the search space. Top-K binarization ensures that all images will be represented by exactly K items. Moreover, we believe (as illustrated by [31]) that the relative frequency of visual words is more important than their absolute frequency.

We repeat this two-step process several times. After a sufficient number of projections, the relative order of the feature frequencies is captured. To illustrate this, let us consider the toy example given Figure 3. The original BoW representation contains four different real-valued features namely p(A), p(B), p(C), and p(D). The first random process selects the visual words A, B, and C. When applying the top-K binarization with $K = 2$, the visual words A and C are kept (and considered as a transaction). This infers that both p(A) and p(C) have higher values than p(B). After the second random projection, the fact that p(C) is smaller than p(A) and p(D) can also be deduced. From these two iterations, we are able to conclude that (i) both p(A) and p(D) are higher than p(C), and (ii) p(C) is higher than p(B). In this example, it is not possible to determine which one of p(A) or p(D) is higher. However, this information can be discovered if we negate the features $h^{p*} = (1 - h_0^p, \ldots, 1 - h_p^p)$ and apply the same binarization[1]. Section 4.1 gives experiments showing that multiple random projections can handle the instability which occurs with a single projection.

As an alternative to random projections, we evaluated the use of principal component analysis (PCA) to reduce the dimensionality of input histograms. However, we obtained much worse results, which can be explained by the fact that it produces a single (low-dimentional) representation per input vector and therefore loses a lot of information once thresholded. In contrast, the proposed method to transform real-valued vectors into multiple sets of binary transactions reduces the dimensionality as well as limits the loss caused by the binarization process.

3.2. Representing images by histograms of patterns

Each random projection generates a set of binary items so-called *a transaction*. At this stage, standard data mining algorithms can be used to discover interesting combinations

[1] For the experiments on image classification and object recognition/detection, an image transaction is the concatenation of two sets, the non-negate and the negate top-K visual words.

Input BoW: p(A)=.4 p(B)=.1p(C)=.2 p(D)=.3	
BoW relative order	p(A) $>$ p(D) $>$ p(C) $>$ p(B)

Random projections	Top-K, $K = 2$
R_1 : p(A) p(B) p(C)	{A C}
R_2 : p(A) p(C) p(D)	{A D}

Discovered from R_1	p(C) $>$ p(B)
Discovered from R_2	p(A) $>$ p(C)
Discovered from R_2	p(D) $>$ p(C)

Figure 3: Toy example showing that after two random projections followed by top-K ($K = 2$) binarizations, relative relations between the BoW features are preserved and implicitely encoded within the representaion.

I	Class	Vis. words distrib.						After proj. R1				Transactions
		A	B	C	D	E	F	A	C	E	F	
1	H^+	.2	.3	.2	.0	.1	.2	.2	.2	.1	.2	{A,C,F}
2		.0	.3	.4	.1	.1	.1	.0	.4	.1	.1	{C,E,F}
3	H^-	.3	.3	.0	.2	.1	.1	.3	.0	.1	.1	{A,E,F}
4		.2	.2	.0	.3	.1	.2	.2	.0	.1	.2	{A,E,F}
5	Test	.0	.3	.1	.1	.4	.1	.0	.1	.4	.1	{C,E,F}
6		.1	.3	.0	.2	.1	.3	.1	.0	.1	.3	{A,E,F}

Figure 4: Toy example showing how patterns are obtained.

of features in each projection-related transactions.

Data Mining has defined several types of patterns, but Frequent Patterns (FPs) [1] and Jumping Emerging Patterns (JEPs) [7] are among the most common ones. Given a specified minimum threshold F_{min}, a pattern (*i.e.* a set of items) is frequent if it appears in no less than F_{min} transactions. Finding the *exhaustive* set of patterns is a major issue. Fortunately, the collection of frequent patterns can be condensed by the set of frequent closed patterns [27]. The intuition is that a closed pattern condenses a set of patterns whose frequency is computed from the same transactions. By definition, closed patterns are patterns to which it is impossible to add any item without decreasing its frequency. We can therefore derive frequent patterns from closed patterns, resulting in a lower complexity: mining closed patterns is *on average* polynomial in the number of items [17] whereas usual techniques are exponential in the worst case. While FP mining is designed for discovering patterns from single classes, JEP mining is based on the growth-rate – a contrast measure [21] – and is relevant for two-class problems. More precisely, a JEP is defined as a pattern appearing in one and only one class. If the frequency of a JEP is no less than F_{min}, it is called a frequent JEP. Closed patterns make the mining of JEP mining much easier. Indeed, closed patterns have optimal growth rates, as they concentrate those patterns having highest growth rate values [18]. In the following, we take benefit from the good properties JEPS offer regarding the representation of classes.

One key idea of the paper is to compute statistics of patterns found for each random projection to build the new image representation. The complete representation is indeed obtained by aggregating histograms of the whole set of patterns coming from the different projections. For supervised tasks – which are those considered in this paper – JEPs are highly relevant as they are discriminative by construction. More precisely, for each random projection, a set of positive JEPs (found only in the positive images) and a set of negative JEPs (found only in the negative images) are extracted. We therefore build a histogram with two bins, the first is the count of positive JEPs (those found in positive images) and the other is the count of negative JEPs. For P random projections, we hence obtain a $(2 \times P)$-dimensional histogram image representation, so-called the HoPS representation. The HoPS representation is intended to be used with classifiers. The underlying idea is to condense the information given by the set of patterns, rather than using each individual pattern as an individual image feature. As mentioned in the introduction, the number of patterns is high and using each pattern as a feature would result in a too large representation.

3.3. Toy example

Figure 4 shows a toy example illustrating the encoding of HoPS using JEPs. There are 6 image histograms (4 training ones for discovering the JEPs, and 2 for testing). The dimensionality of the input space is $d = 6$ while the dimensionality of the projected space is $p = 4$. We set the number of random projection P to 1 in order to simplify the illustration (the other projections follow the same principle). This projection is such that the projected vectors are made of the 1st, 3rd, 5th and the 6th component of the original ones. In this example $K = 3$, meaning that the 3 visual words having the highest probabilities are kept in the transactions. After obtaining the transactions (also given in Figure 4), the JEPs with minimum frequency threshold $F_{min} = 1$ are computed. In this example, positive JEPs are {A,C,F}, {C,E,F}, and {C,F} while {A,E,F} is the single negative JEP. Note that *e.g.* {C} is not generated by our method since {C} is not a closed pattern ({C} is covered by {C,F}.) Finally, the two test images are coded by counting the number of positive and negative JEPs. The final representation consists in these two values, *i.e.* $(2, 0)$ for image 5 and $(0, 1)$ for image 6.

3.4. Complexity analysis

Encoding new images is very fast: the histogram binarization procedure involves the search of the k-th highest values, which can be done in $\mathcal{O}(p \times log(k))$ using a partial sort algorithm. Counting the JEPs supported by the transactions can be done in $\mathcal{O}(N_p)$ where N_p is the number of mined patterns. To give some insights about the run time, encoding 100,000 images with 500 JEPs (which is in average the number of JEPs in our experiments), on a single core of CPU, takes around 0.3 second for the binarization and around 1 sec. for counting JEPs. Regarding training, mining the JEPs takes about 1 sec. with the typical settings/datasets used in our experiments.

4. Experiments

These experiments validate our approach on several supervised classification tasks. These experiments show that HoPS are compact, perform well, and are very generic. HoPS are generic in the sense that they can be built on top of various histogram-based image representations. In our experiments, HOG, LTP and FLH are used, but other could be considered.

We validated HoPS in four recognition tasks: (a) *texture recognition* on the KTH-TIPS2a dataset [4], (b) *pedestrian recognition* on the Daimler Multi-Cue, Occluded Pedestrian Classification dataset [8], (c) *image classification* on the Oxford-Flowers 17 dataset [20], and (d) *object detection* on the PASCAL VOC 2007 dataset [9]. The type of patterns used are positive/negative JEPs. We use the code of DPM-delta [18] to extract them. We set the F_{min} threshold to 1% of the number of positive images. For many categories, the number of positive images is about one hundred and 1% is the minimum frequency that can be set. For each task, HoPS is compared against the baseline which is the original first order representation *i.e.* without taking into account feature dependencies. The same classifier and the same protocol are used for HoPS and the baseline representation. Since all of the representations including HoPS are histograms, we L1-normalize and square-root the representations before training linear SVM classifiers.

4.1. Influence of the parameters

To get some insight on the proposed method, we present several experiments on a simplified dataset and show the influence of the parameters on the behaviour of the algorithm (especially the performance). This simple dataset is a subpart of the KTH-TIPS2a [4] dataset, in which the *lettuce leaf* category is considered as the positive class while the remaining categories are considered as the negative class. We observed that there results generalize to other categories. The training set contains 3,420 images with 324 positives. The category of test images are predicted taking HoPS as input, and the performance is measured by the Average Precision (AP). As low-level features, we use single scale circular sampling of uniform-LTP, as proposed by [26]. Consequently, the HoPS are computed from 118-d histograms of LTP.

To encode HoPS, there are three key parameters to be considered: (i) the number of projections (P), (ii) the dimensionality of the projection space (p) and (iii) the number of items (K) per transactions obtained by top-K binarization of the projected histograms.

Regarding the number of random projections (P), Figure 5a shows the AP as a function of P for different values of K and p. The AP increases with P until it saturates. As described in Section 3.1, the loss from binarization can be reduced by applying multiple projections. It seems that after a certain number of projections, the information contained in the original representation is well covered, explaining why the performance stops increasing. P depends on the dimensionality of the original feature d. In the toy example, in Figure 3, there are only 4 features, and only 2 random selections are sufficient to cover the relative order of the features. Clearly, when the original feature has higher dimensions, more random processes are required to capture the relative order of the features.

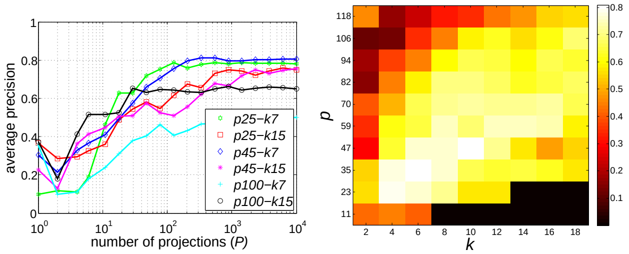
The influence of the binarization threshold (K) is represented Figure 5(b), which shows the performance according to the number of items (K) and the dimensionality of the projection space (p). K has been set from 2 to 18 while p varies from 10% to 100% of the size of the input dimensionality (in this experiment $d = 118$). We observed that K and p are correlated. The optimal score is reached when K is set to about 20% of p.

Finally, the impact of the size of the projection space p is illustrated Figure 5(b). p has to be chosen accordingly to the dimension of the input space d. Let's refer once again to the toy example in Figure 3 to explain this. In the toy example, if instead of 3, p was set to 4 which is equal to d, there will be only one possible transaction per image. After the top-K binarization, only the information that p(A) and p(D) are higher than p(C) and p(B) would be encoded. The relative order between p(A) and p(D), and the relative order between p(C) and p(B) could not be discovered. For this reason p should not be too high. In addition, as discussed earlier that p and K are related to each other, setting p to a high value leads to a high value of K. Recall that the mining complexity increases with K, although different values of p can result in similar performance, a low value of p is preferred. However, if p is too small, transactions will not have enough items and finding useful combinations will not be possible. This explains behavior observed in Figure 5(b).

In the forthcoming experiments (next sections), the parameters are cross-validated using a validation set.

4.2. Non-linear SVM and combination of first and higher order features.

As the proposed representation encodes some non-linearities of the features, it is interesting to compare it against a non-linear classifier applied to the features taken as independent (*i.e.* the original histogram). In this experiment, we compared the performance of a linear SVM using



(a) AP according to the number of projections.

(b) AP as a function of the dimensionality of the projection spaces p and the number of items per transactions K

Figure 5: Influence of the parameters on the behavior of the algorithm.

Kernel	1st order	HoPS	1st + HoPS
Linear	68.8	74.1	75.0
RBF-χ^2	71.2	73.7	74.0

Table 1: Comparing linear / non-linear classifiers (mAP).

our HoPS with a RBF-χ^2 SVM (known to perform well on histograms) using histograms of the original features. The parameters of the RBF-χ^2 kernel were set by cross validation. We computed the Average Precision (AP) on the 11 classes of the KTH- TIPS2a dataset and report in Table 1 the mean Average Precision (mAP). It is worth pointing out that the HoPS used with a linear classifier outperformed the first order features with the non-linear SVM. As combining, by a simple concatenation, first-order features (histograms of features) and HoPS gave a slightly better performance than HoPS alone, this combination has been done in all classification experiments.

4.3. Image classification

Our primary motivation for doing these experiments was to provide comparison with the approach of [11], the most recent pattern mining based approach on image classification. The interesting point is that the authors made the image features (FLH) for the Oxford-Flowers 17 database publicly available[2], allowing us to give some meaningful comparisons (as our patterns are made from strictly the same low-level image features as theirs).

The Oxford-Flowers 17 database[3] contains 17 flower categories with 80 images per category. To compare results, we followed the same protocol as [11], *i.e.* 20-fold cross validation, and reported the mean classification accuracy (mA). Results are given in Table 2. Note that we did not give comparisons with the features of [11] obtained by combining shape (FLH_S) and color (FLH_C), as the authors has only publicized FLH_S features. With a mA of $\mathbf{93.8} \pm \mathbf{1.4}$, our approach outperformed the performance of all the recent approaches we are aware of, including [11].

method	mA($\%$)
Nilsback [20]	88.3 ± 0.3
CA [16]	89.0
L1BRD [31]	89.0 ± 0.6
FLH_S [11] (baseline)	92.0 ± 1.5
HoPS (ours)	$\mathbf{93.8} \pm \mathbf{1.4}$

Table 2: Comparison with state-of-the-art results on the Oxford Flower 17 dataset.

Method	mA($\%$)
Chen et al. [5]	64.7
Caputo et al.[4]	71.0
LHS [24]	73.0 ± 4.7
Color+MLBP[19]	73.1 ± 4.6
histogram of LTP (baseline)	69.9 ± 3.0
HoPS (ours)	**75.0±3.3**

Table 3: mean accuracy(mA) on the KTH-TIPS2a dataset.

4.4. Texture recognition

The experiments on texture recognition have been done on the KTH-TIPS2a [4] dataset, which is a dataset including images of 11 different materials (*e.g.* wool, linen, cork). For each material there are 4 image samples in which the samples were photographed at 9 scales, 3 poses and 4 different illumination conditions. The evaluation was done by following the protocol proposed in [4], which consists in reporting the mean Accuracy (mA) over the 4 runs (multi-class classification). During each run, all images of one sample were taken for testing while the remaining images of the 3 samples were used for training. As in Section 4.1, the features used for the experiments were 118-d histograms of uniform-LTP features [26]. As shown in Table 3, our method improves the performance the baseline more than 5%, validating the relevance of modeling the dependency between features. In addition, our method significantly outperformed other approaches[4].

4.5. Object detection

The experiments on object detection are achieved on the PASCAL VOC 2007 dataset, consisting in 9,963 images of 20 different object classes with 5,011 training images and 4,952 testing images. The task is to predict bounding boxes of the objects of interest if they are present in the images. It is the most popular dataset for object detec-

[2] http://homes.esat.kuleuven.be/~bfernand/eccv_flh

[3] http://www.robots.ox.ac.uk/~vgg/data/flowers

[4] We did not report the results of [4] using multi scale features and complex decision trees (with non-linear classifiers at every node), as our point is to demonstrate the strength of our features for a given classifier.

tion and several competitive methods have been proposed for this dataset. Note that it is very difficult to improve results on this dataset. The top recently reported results are only slightly different.

We built on the well known part-based detector [10] version 5 (the latest released). As our goal is to validate the proposed image representation (and not to propose a new detector), we used the detection framework of [10] as it is but replaced the original image features (HOG) by ours in the scoring function. Following several works showing that combining texture and shape together gives better results (e.g. [14, 34]), we combined LTP and HOG features. In practice, the representation was obtained as follows. We first rescaled the bounding boxes of training object (roots and parts obtained by using the standard pre-trained detector) to the size of 128 × 128 pixels. Uniform LTP features were then extracted from a 6 × 6 non-overlapping grid, giving a total dimension of 4,248.

With the combination of (HOG+LTP), we obtained a gain of 1.0% mAP over the original detector. Then, the distribution of HOGs and LTPs are used as the input of our algorithm for computing the HoPS. As shown in Table 4, the gain of our full system is of 1.7% mAP over the original detector. We believe than the improvement is less on this dataset as JEP suffers from the very high imbalance of the positive/negative samples in this task (*i.e.* few hundreds positives vs hundred thousand negatives). Other types of patterns which can handle this issue have to be investigated. Since our method is independent of the type of patterns used, JEPs can easily be replaced by other types of patterns. Nevertheless, JEPs improve the baseline. Most of all, we achieved state-of-the-art result on this extremely competitive dataset.

4.6. Pedestrian recognition

The Daimler pedestrian classification dataset [8] consists of manually labeled pedestrian and non-pedestrian bounding boxes in images captured from a vehicle. The images have a resolution of 48 x 96 pixels with a 12-pixel border around the pedestrians. The dataset is split into three subsets (i) 52,112 non-occluded pedestrian samples for training (ii) 25,608 non-occluded pedestrians for testing and (iii) 11,160 partially occluded pedestrian samples, also for testing.

For our baseline, we extracted Histograms of Oriented Gradients (HOG) descriptors (HOG) from intensity grayscale images, using the same setting as in [8], which is 12 orientation bins and 6x6 pixel cells, accumulated to overlapping 12 × 12 pixel blocks with a spatial shift of 6 pixels. As shown in Figure 6 our baseline is already better than the best result of [8] which combined intensity, stereo, and optical flow, probably due to the square-root normalization. The proposed HoPS improves significantly over the baseline, with a reduction of about 30% of the false positive rate. HoPS outperforms other state-of-the-art approaches such as [30] as well.

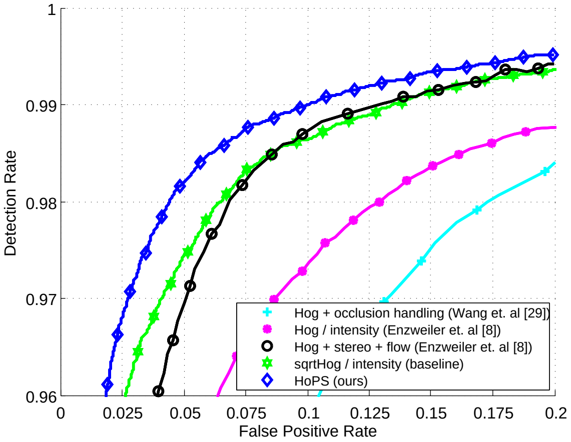


Figure 6: Pedestrian recognition on the non-occluded testset of Daimler pedestrian dataset.

5. Conclusions

This paper proposes a new method for discovering dependencies between image features, and encode them through Histograms of Pattern Sets (HoPS). The key of the proposed approach lies in (i) multiple random selections and top-K binarization – making pattern mining tractable while minimizing the information loss caused by the binarization – allowing to represent images by compact sets of meaningful transactions, and (ii) the introduction of Histograms of Pattern Sets, shown to be a very efficient and compact way to represent images. The proposed approach is generic and can be built on top of several image features. The proposed approach is validated on four different datasets, achieving state-of-the-art performance in the context of image classification and object detection.

References

[1] R. Agrawal, T. Imieliński, and A. Swami. Mining association rules between sets of items in large databases. In *ACM SIGMOD Record*, volume 22, pages 207–216. ACM, 1993. 4

[2] A. D. Bagdanov. Color attributes for object detection. In *CVPR*, 2012. 8

[3] B. Bringmann and A. Zimmermann. The chosen few: On identifying valuable patterns. In *ICDM*, 2007. 2

[4] B. Caputo, E. Hayman, and P. Mallikarjuna. Class-specific material categorisation. In *ICCV*, 2005. 5, 6

[5] J. Chen, S. Shan, C. He, G. Zhao, M. Pietikainen, X. Chen, and W. Gao. WLD: A Robust Local Image Descriptor. *IEEE PAMI*, 32(9):1705–1720, 2010. 6

	mAP	plane	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv
Best 2007 [9]	*23.3*	26.2	40.9	9.8	9.4	21.4	39.3	43.2	24.0	12.8	14.0	9.8	16.2	33.5	37.5	22.1	12.0	17.5	14.7	33.4	28.9
UCI [6]	*27.1*	28.8	56.2	3.2	14.2	29.4	38.7	48.7	12.4	16.0	17.7	24.0	11.7	45.0	39.4	35.5	15.2	16.1	20.1	34.2	35.4
LEO [35]	29.4	**55.8**	9.4	14.3	**28.6**	**44.0**	51.3	21.3	20.0	19.3	25.2	12.5	**50.4**	38.4	36.6	15.1	**19.7**	**25.1**	36.8	39.3	29.6
Oxford-MKL [28]	*32.1*	37.6	47.8	**15.3**	15.3	21.9	50.7	50.6	**30.0**	17.3	**33.0**	22.5	21.5	51.2	45.5	23.3	12.4	23.9	28.5	45.3	48.5
LS [34]	*34.3*	36.7	59.8	11.8	17.5	26.3	49.8	58.2	24.0	**22.9**	27.0	24.3	15.2	58.2	49.2	**44.6**	13.5	21.4	34.9	47.5	42.3
CA [2]	*34.8*	34.5	**61.1**	11.5	19.0	22.2	46.5	58.9	24.7	21.7	25.1	27.1	13.0	59.7	**51.6**	44.0	19.2	24.4	33.1	48.4	**49.7**
DPM rel.5 [10]	*33.7*	33.2	60.3	10.2	16.1	27.3	54.3	58.2	23.0	20.0	24.1	26.7	12.7	58.1	48.2	43.2	12.0	21.1	36.1	46.0	43.5
LTP+HOG (baseline)	*34.7*	36.2	60.7	10.8	18.6	27.4	53.7	**59.2**	24.2	20.0	25.8	26.9	14.0	59.3	49.7	43.1	12.1	22.4	**38.5**	48.4	43.6
HoPS (ours)	***35.4***	37.0	60.7	11.2	18.6	27.8	**54.5**	59.1	26.9	20.5	25.8	**29.0**	15.3	**59.9**	49.8	43.0	13.4	23.2	38.4	**48.8**	45.1

Table 4: Comparison with state- of-the-art approaches on the PASCAL VOC 2007 dataset.

[6] C. Desai, D. Ramanan, C. Fowlkes, and U. C. Irvine. Discriminative models for multi-class object layout. In *ICCV*, 2009. 8

[7] G. Dong and J. Li. Efficient mining of emerging patterns: discovering trends and differences. In *KDD*, 1999. 4

[8] M. Enzweiler, A. Eigenstetter, B. Schiele, and D. M. Gavrila. Multi-cue pedestrian classification with partial occlusion handling. In *CVPR*, 2010. 5, 7

[9] M. Everingham, L. Van Gool, C. Williams, J. Winn, and A. Zisserman. The pascal visual object classes challenge 2007. In *2th PASCAL Challenge Workshop*, 2009. 5, 8

[10] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part-based models. *IEEE PAMI*, 32(9):1627–1645, 2010. 7, 8

[11] B. Fernando, É. Fromont, and T. Tuytelaars. Effective use of frequent itemset mining for image classification. In *ECCV*, 2012. 1, 2, 6

[12] A. Gilbert, J. Illingworth, and R. Bowden. Scale invariant action recognition using compound features mined from dense spatio-temporal corners. In *ECCV*, 2008. 1, 2

[13] A. Gionis, P. Indyk, and R. Motwani. Similarity search in high dimensions via hashing. In *Proceedings of the 25th International Conference on Very Large Data Bases*, VLDB, 1999. 3

[14] S. U. Hussain and B. Triggs. Feature sets and dimensionality reduction for visual object detection. In *BMVC*, 2010. 7

[15] H. Jégou, F. Perronnin, M. Douze, J. Sánchez, P. Pérez, and C. Schmid. Aggregating local image descriptors into compact codes. *IEEE PAMI*, 34(9):1704–1716, 2012. 3

[16] F. S. Khan, J. van de Weijer, and M. Vanrell. Top-down color attention for object recognition. In *ICCV*, 2009. 6

[17] L. Lhote, F. Rioult, and A. Soulet. Average number of frequent (closed) patterns in bernouilli and markovian databases. In *ICDM*, 2005. 4

[18] J. Li, G. Liu, and L. Wong. Mining statistically important equivalence classes and delta-discriminative emerging patterns. In *KDD*, pages 430–439, 2007. 4, 5

[19] W. Li and M. Fritz. Recognizing materials from virtual examples. In *ECCV*, 2012. 6

[20] M.-E. Nilsback and A. Zisserman. Automated flower classification over a large number of classes. In *ICVGIP*, 2008. 5, 6

[21] P. K. Novak, N. Lavrac, and G. I. Webb. Supervised descriptive rule discovery: A unifying survey of contrast set, emerging pattern and subgroup mining. *Journal of Machine Learning Research*, 10:377–403, 2009. 2, 4

[22] S. Nowozin, K. Tsuda, T. Uno, T. Kudo, and G. BakIr. Weighted substructure mining for image analysis. In *CVPR*, 2007. 2

[23] T. Quack, V. Ferrari, B. Leibe, and L. J. V. Gool. Efficient mining of frequent and distinctive feature configurations. In *ICCV*, 2007. 1, 2

[24] G. Sharma, S. U. Hussain, and F. Jurie. Local higher-order statistics (lhs) for texture categorization and facial analysis. In *ECCV*, 2012. 6

[25] J. Sivic and A. Zisserman. Video data mining using configurations of viewpoint invariant regions. In *CVPR*, 2004. 2

[26] X. Tan and B. Triggs. Enhanced local texture feature sets for face recognition under difficult lighting conditions. In *Analysis and Modeling of Faces and Gestures*. 2007. 5, 6

[27] T. Uno, T. Asai, Y. Uchida, and H. Arimura. An efficient algorithm for enumerating closed patterns in transaction databases. In *DS*, 2004. 4

[28] A. Vedaldi, V. Gulshan, M. Varma, and A. Zisserman. Multiple kernels for object detection. In *ICCV*, 2009. 8

[29] L. Wang, Y. Wang, T. Jiang, and W. Gao. Instantly telling what happens in a video sequence using simple features. In *CVPR*, 2011. 2

[30] X. Wang, T. X. Han, and S. Yan. An hog-lbp human detector with partial occlusion handling. In *ICCV*, 2009. 7

[31] N. Xie, H. Ling, W. Hu, and X. Zhang. Use bin-ratio information for category and scene classification. In *CVPR*, 2010. 3, 6

[32] B. Yao and L. Fei-Fei. Grouplet: A structured image representation for recognizing human and object interactions. In *CVPR*, 2010. 2

[33] J. Yuan, M. Yang, and Y. Wu. Mining discriminative co-occurrence patterns for visual recognition. In *CVPR*, 2011. 1, 2

[34] J. Zhang, K. Huang, Y. Yu, and T. Tan. Boosted local structured hog-lbp for object localization. In *CVPR*, 2011. 7, 8

[35] L. Zhu, Y. Chen, A. L. Yuille, and W. T. Freeman. Latent hierarchical structural learning for object detection. In *CVPR*, 2010. 8