Hamming Embedding Similarity-based Image Classification

Mihir Jain, Rachid Benmokhtar, Patrick Gros, Hervé Jégou

Hamming Embedding Similarity-based Image Classification

Mihir Jain
INRIA Rennes
mihir.jain@inria.fr

Rachid Benmokhtar
INRIA Rennes
rachid.benmokhtar@inria.fr

Patrick Gros
INRIA Rennes
patrick.gros@inria.fr

Hervé Jégou
INRIA Rennes
herve.jegou@inria.fr

ABSTRACT

In this paper, we propose a novel image classification framework based on patch matching. More precisely, we adapt the Hamming Embedding technique, first introduced for image search to improve the bag-of-words representation. This matching technique allows the fast comparison of descriptors based on their binary signatures, which refines the matching rule based on visual words and thereby limits the quantization error. Then, in order to allow the use of efficient and suitable linear kernel-based SVM classification, we propose a mapping method to cast the scores output by the Hamming Embedding matching technique into a proper similarity space. Comparative experiments of our proposed approach and other existing encoding methods on two challenging datasets PASCAL VOC 2007 and Caltech-256, report the interest of the proposed scheme, which outperforms all methods based on patch matching and even provide competitive results compared with the state-of-the-art coding techniques.

Categories and Subject Descriptors

I.4.10 [**Image Processing and Computer Vision**]: Image Representation; H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval

General Terms

Algorithms, Experimentation, Measurement, Performance

Keywords

Similarity-based learning, Hamming Embedding, evaluation, image classification

1. INTRODUCTION

Image classification is a challenging problem and the key technology for many existing and potential mining applications. It has attracted a large interest from the Multimedia and Computer Vision communities in the past few decades, due to the ever increasing digital image data generated around the world. It is defined as the task of assigning one or multiple labels corresponding to the presence of a category in the image.

Recently, machine learning tools have been widely used to classify images into semantic categories. Local features combined with the bag-of-visual-words representation of images demonstrate decent performance on classification tasks [28]. The idea is to characterize an image with the number of occurrences of each visual word [22]. However, it is generally admitted that this setup is sub-optimal, as the discriminative power of the local descriptors is considerably reduced due to the coarse quantization [3] operated by the use of a pre-defined visual vocabulary. To address this problem, several encodings have been proposed such as locality-constrained linear [25], super vector [16, 29], kernel codebook [24], and the Fisher Kernel [20, 21]. These coding schemes are compared by Chatfield et al. [5] on the popular PASCAL'07 and Caltech-101 benchmarks. Considering dense SIFT sampling, which are shown to outperform interest points for classification, they use a linear classifier for better efficiency and confirm that these new coding schemes indeed achieve better classification accuracy than the spatial histogram of visual-words baseline. The superiority of the improved Fisher Kernel [21] is evidenced among all these schemes.

Another way to limit the quantization error introduced by the use of visual words instead of full descriptors consists in adopting a matching approach [3, 23]. These schemes require the use of full raw descriptors, which is not feasible when considering large learning sets such as those considered in large image databases like ImageNet [6]. Moreover, to our knowledge these methods have not been shown to exhibit a classification accuracy as good as those reported with the aforementioned new coding schemes.

Besides, in the context of image search, some solutions have been proposed to dramatically improve the matching quality while keeping a decent efficiency and memory usage. In particular, improved accuracy is achieved by incorporating additional information, jointly with the descriptors, directly in the inverted file. This idea was first explored by Jégou et al. [13] for image search, where a richer descriptor representation is obtained by using binary codes in addition to visual words and weak geometrical consistency. In our work,

Pre-print (author) version.
To be published in ICMR'2012.

we will mainly focus on the interest of the complementary information provided by the binary vectors, using the so-called Hamming Embedding method [15, 14]. The idea was recently pushed further by Jain et al. [12], who show that a vector-to-binary code comparison improves the Hamming Embedding baseline by limiting the approximation made on the query, leading to state-of-the-art results for the image search problem.

In this paper, in the spirit of recent works that have shown the interest of patch-based techniques for classification, we propose to adopt the state-of-the-art Hamming Embedding method for category-level recognition. This produces a representation which is more efficient and compact in memory than the solutions based on exact patch matching. However, the original Hamming Embedding technique can not be used off-the-shelf, since the similarity output by this technique is not a Mercer Kernel. A naive option would be to adopt instead a k-nearest neighbor classifier, but from our preliminary experiments the resulting classification accuracy is then low. To address this problem, we adopt a kernelization technique on top of our matching-based solution, which enables the use of support vector machines and thereby allows good generalization properties even when using a linear classifier. As a result, Hamming Embedding classification is efficient in both training and testing stages, and provides better performance and efficiency than the recently proposed concurrent matching-based classification techniques [3, 23].

Last but not least, the proposed approach is shown to outperform the most recent coding schemes benchmarked in [5]. The only noticeable exception is the latest improvement of the Fisher Kernel [21], which still remains competitive. Beside, we show that the combination of Hamming Embedding similarity with Fisher Kernel is complementary and achieves the current state-of-the-art performance. Most importantly and as noticed in [23], the high flexibility offered by a matching-based framework is likely to pave the way to several extensions.

The rest of the paper is organized as follows: Section 2 describes the most related works. Section 3 presents our system architecture. It includes the feature extraction procedure, where an improved SIFT descriptor is introduced, and the Hamming Embedding similarity-based representation. Section 4 reports the experimental results conducted on the PASCAL VOC 2007 and Caltech-256 collections and compare them to the main state-of-the-art methods discussed in Section 2. Section 5 concludes the paper.

2. RELATED WORK

The bag of visual-words (BOW) is one of the most popular image representations, due to its conceptual simplicity, computational efficiency and discriminative power stemming for the use of local image information. It represents each local feature with the closest visual word and counts the occurrence frequencies in the image. The length of the histogram is given by the number of visual words of a codebook dictionary. Van Gemert et al. [24] introduced an uncertainty model based on kernel density estimation to smooth the hard assignment of image features to codewords. However, BOW discards the spatial order of local descriptors, which severely limits the descriptive power of the image representation. To take into account the rough geometry of a scene, the spatial pyramid matching (SPM) proposed by Lazebnik et al. [18] divides the image into blocks and concatenates all the histograms to form a vector descriptor which incorporates the spatial layout of the visual word. The BOW and this SPM extension are generally used in conjunction with non-linear classifiers. In this case, the computational complexity is $O(N^3)$ and the memory complexity is $O(N^2)$ in the training phase, where N is the size of the training dataset. This complexity limits the scalability of BOW- and SPM-based non-linear SVM methods.

In order to limit the quantization error, Yang et al. [27] propose a linear spatial pyramid matching method based on sparse coding (ScSPM). A *max* pooling spatial pooling replaces the average pooling method for improved robustness to local spatial translations. A very successful method is the improved Fisher Kernel (FK) proposed by Perronnin et al. [21] in the context of image categorization. The idea of FK [11, 20] is to characterize an image with the gradient vector of the parameters associated with a pre-defined generative probability model (a gaussian mixture in [20]). This representation is subsequently fed to a linear discriminative classifier. andcan be used jointly with other techniques such as the SPM representation, power-law [21] and L_2 normalizations. The FK boosts the classification accuracy, at the cost of a high descriptor dimensionality, which is two orders of magnitude larger than BOW for the same vocabulary size. However, the FK is classified using a linear SVM, which counter-balance the higher cost of the non-linear classifiers involved in BOW-based classification. Other improvements are achieved by combining different types of local descriptors or by integrating objects localization task [10].

This paper follows another line of research on building a kernelized efficient matching system. Various similarity matching methods are proposed in the literature. Some of these use feature correspondences to construct an image comparison kernel which is compatible with SVM-based classification [4]. Bo and Sminchisescu [2] propose an effective kernel computation through low-dimensional projection. Duchenne et al. [7] extend the region-to-image matching method of Kim et al. [17], and formulate image graph matching as an energy optimization problem, whose nodes and edges represent the regions associated with a coarse image grid and their adjacency relationships.

A very simple matching-based method is the one proposed by Boiman et al. [3], who use a Nearest Neighbor (NN) as a nonparametric model, with does not require any training phase. Two approaches are considered: NN Images-to-Images and NN Images-to-Classes. For the first approach, each test image is compared to all known images and the class of the closest image is chosen and assigned to queried image. The second approach pools all descriptors of all the images belonging to each class to form a single representation of that class. A given image is then compared to all the classes. NN Images-to-Classes achieves good results on standard benchmarking datasets. Tuytelaars et al. [23] exploited the kernel complementarity by combining NN and BOW. However, as shown in our experimental section, our matching-based method is the first to report competitive results against the best encoding method, namely the FK.

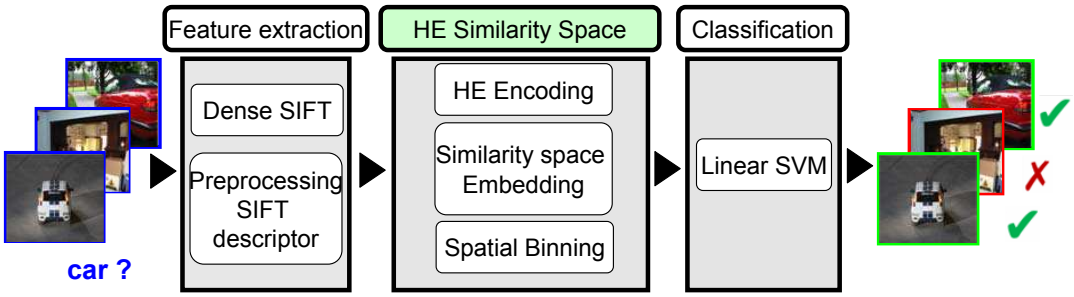

Figure 1: Proposed image classification system architecture.

3. PROPOSED APPROACH

Figure 1 gives an overview of our method. We first extract local features on a dense grid. For this purpose, we propose an improved variant of the SIFT [19] descriptor. It better takes into account the contrast information than the original one. These descriptors are *individually* encoded using the Hamming Embedding technique [15], which represents each descriptor by a visual word and a short binary signature. This binary vector is shown to integrate some residual information about the class which is not captured by the visual word. At this stage, the similarity between two images is done by computing the score produced by HE. More precisely, we used the extended HE method [14] by Jegou et al., which integrates a regularization technique to address the visual burstiness phenomenon encountered in images. The images are then described in a *similarity space*, which amounts to constructing a vector whose components correspond to a similarity to a fixed set of training images. A linear SVM classifier is then learned in this similarity space.

3.1 Feature Extraction

We extract SIFT [19] descriptors from a dense grid. More precisely, we adopt the same grid parameters as used in [5], i.e., a spatial stride of 3 pixels with multiple resolutions. These extracted patches are described using a local descriptor derived from the original SIFT descriptor [19]. The proposed variant of SIFT aims at addressing the following issues:

- A strong gradient, such as generated by a boundary, gives an overwhelming importance to a few components in the SIFT descriptor. Lowe proposes a solution [19] to address this problem by clipping the components whose value is larger than 20% of the whole energy. However, this solution is not satisfactory since it does not correct the components which magnitude is lower than this threshold.
- The SIFT descriptor are L2-normalized[1], in order to ensure invariance to intensity changes. However, this solution completely discards the absolute value of the gradient, which is a meaningful information.
- Dense patch sampling produces many uniform patches which are not very informative. Worst, uniform patches have a low signal to noise ratio. Consequently, the normalization that is performed to achieve intensity-invariance magnifies the noise.

[1]In typical implementations, they are finally multiplied by a constant such that the components lie in the range [0..255], in order to encode each component with 1 byte.

To address these issues, the SIFT generation procedure is modified as follows. Starting with the SIFT descriptor before clipping and normalization,

1. The descriptors with zero norm are filtered out, which amounts to removing the uniform patches.
2. Instead of the clipping procedure, each component is square rooted. This power-law component-wise regularization is similar to the one performed in the improved Fisher Kernel [21], but here applied directly on the local descriptor.
3. Finally, instead of using the L2 normalization, the final vector is normalized by the square root of the L2 norm of the transformed descriptor. This gives a better trade-off between full invariance to intensity change and keeping the information about the absolute intensity measure.

We have evaluated the interest of this SIFT variant on the PASCAL VOC 2007 classification benchmark. For this we use our proposed approach described in Section 3.4 as well as the improved FK method [21]. We observe gain of around 2% of mAP when this SIFT variant is used with our approach for classification. Fisher Kernel with this variant achieves an mAP of 59.8% and 62.2% without and with spatial grid, respectively. These results are 0.5 to 1.5% better than the regular SIFT descriptor in the same setup.

3.2 Hamming Embedding

The Hamming Embedding method of [13] is a state of the art method for image retrieval. It provides an accurate way of computing the similarity between two images based on the distance between their local descriptors. It can be seen as an extension of BOW, where a better representation of the images is obtained by adding, to the visual word, a short binary signature that refines the representation of each local descriptor.

To generate the binary signature, each descriptor is first projected onto a m-dimensional space by a fixed random rotation matrix. Each projected component is compared with

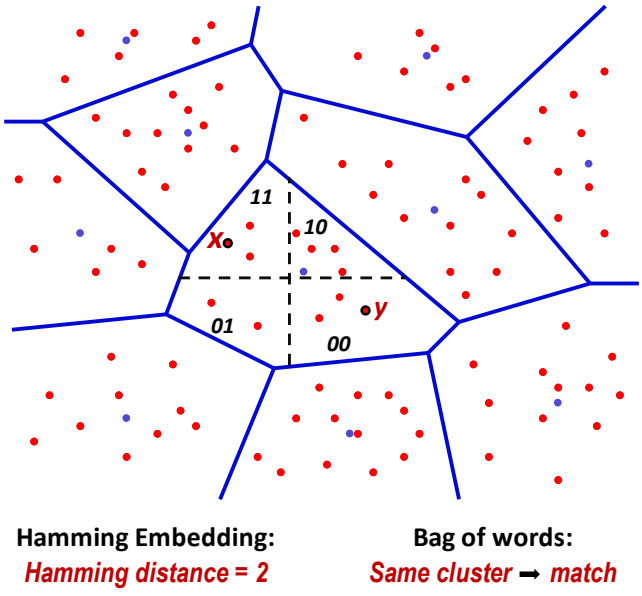

Figure 2: Hamming Embedding matching: In case of bag of words, being in the same cluster descriptors x and y match. While with HE matching depends on the relative location of the descriptors.

a median value learned, in an unsupervised way, on an independent dataset. This comparison produces either a **0** or a **1** per component, producing a bit-vector of length m. This binary signature gives a better localization of the local descriptor in the Voronoi cell associated with the visual word. Figure 2 illustrates this method for a 2-dimensional feature space ($m = 2$). Each red dot shows a descriptor. The cluster centers (visual words) are represented by the blue dots. For a given cell, the hyperplanes or axes (shown in dashed lines) represent the decision boundaries associated with the comparison of the projected components with the median values. As a result, the cell is partitioned into 2^m sub-cells, each of which being associated with a given binary signature.

Two descriptors assigned to the same visual word are compared with the Hamming distance between their binary signatures. They are said to be matched only if the distance is less than a fixed threshold h_t. This provides a more accurate comparison between descriptors than in the BOW model, where the descriptors are assumed to match if they are assigned to the same visual word. In the example of Figure 2, the descriptors **x** and **y** belong to the same cluster, but have binary signatures **00** and **11**, respectively, which means that they are not similar.

Each successful matching pair votes, which increases the similarity score by a quantity that depends on the Hamming distance between the binary vectors. The final image similarity score is computed as the sum of voting scores and then normalized as in BOW. For the sake of efficiency, the method uses a modified inverted file structure which incorporates the binary signature. As in the original work of Jegou et al. [13] we use $m = 64$, and consider Hamming thresholds between $h_t = 20$ and $h_t = 24$.

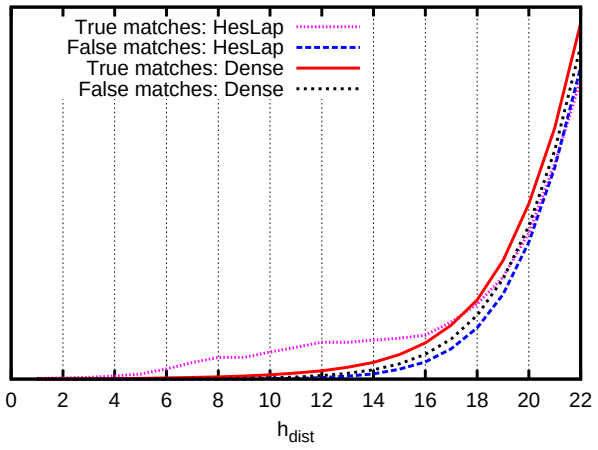

Figure 3: Empirical distribution of true matches and false matches as a function of the Hamming distance ($h_{\mathbf{dist}}$). We only show a zoomed version for $h_{\mathbf{dist}} = 0$ to 22. Measurements are performed on PASCAL VOC 2007 dataset (category *boat*).

Score weighting. As mentioned above, the Hamming distance between two descriptors is used to weight the voting score. This was first done by considering a Gaussian function [14] of this distance. In this work, we adopt a more simple choice in order to remove the parameter σ associated with the Gaussian function. More precisely, we use the linear scoring function $\frac{h_t - h}{h_t}$, where h is the Hamming distance between the binary signatures. From our preliminary experiments, the results obtained by this linear weighting scheme are comparable to the original Gaussian weighting function, which requires to optimize σ by cross-validation.

Burstiness Regularization. In [14], a Burstiness regularization procedure is proposed to achieve improved results in image search. The so-called *burstiness* handling method regularizes the score associated with each match, to compensate the bursty statistics of regular patterns in images. Following these guidelines, we also apply this regularization to obtain better similarity scores.

3.3 HE for classification: motivation

Compared to the BOW representation, the main interest of HE is the additional information provided by the binary signature. We have conducted an analysis to evidence that the Hamming distances between local descriptors provide a complementary and discriminative information for image classification. This analysis is performed on the PASCAL VOC 2007 dataset. SIFT descriptors are extracted from a dense grid as well as from the Hessian Laplace interest points. Any pair of descriptors is referred to as a *true match* if the two descriptors come from same object category, otherwise it is considered as a *false match*.

Figure 3 gives the empirical distribution, zoomed on small distances, of the true and false matches, as a function of the Hamming distance h_{dist}, for the category *boat*. One can observe that the Hamming distance provides a strong prior

about the class: the expectation of false matches is clearly an increasing function of h_{dist}, which confirms that low Hamming distances are more often related to true matches.

Note that all the false matches are accepted in the case of the BOW framework, that only uses vector quantization. Hamming Embedding based matching is able to filter out many false matches by choosing a proper threshold h_t. Moreover, the contribution of the matches are advantageously weighted based on the Hamming distance, in order to reflect the true/false match prior, and therefore to achieve better image classification. Note that setting a high threshold h_t would allow many false-matches to vote (as in BOW). On the other hand, a very low value is not satisfactory because too few matches are kept. It is therefore important to choose a threshold in an appropriate range, which is done by cross validation for a given dataset, see Section 4.1.

3.4 Hamming Embedding Similarity Space

We propose to apply HE to represent images in a similarity space. The idea is to represent an image by its similarity, as output by a strong matching system, to a set of sample images. There are few methods in the literature that employ such a similarity space for classification. These include nearest neighbors based approaches like NBNN [3] and its variations [1, 23] or graph based matching methods [7], see Section 2 for a short survey. However, none of these works is able to compete with the state-of-the-art Fisher kernel [21].

Similarity space image representation. Unlike NBNN, which relies on pure NN classification, our motivation is to produce an image representation that can be fed to a strong classifier such as an SVM. This is more similar to [23] and [7], which use NBNN and graph-matching in their matching system. In our case, the HE similarity between a given image and the training images is obtained as the sum of voting scores, with burstiness regularization. Based on the analysis in Section 3.3, such a *similarity space embedding* is expected to be more discriminative than BOW. The image is represented by an N dimensional vector, where N is the number of training images. Each of its component is a similarity score to one of the training images. A given image I is therefore represented as:

$$I_{HE} = [\text{HE}_{\text{sim}}(I, I_1) \;\; \text{HE}_{\text{sim}}(I, I_2) \;\; ... \;\; \text{HE}_{\text{sim}}(I, I_N)] \quad (1)$$

where $\text{HE}_{\text{sim}}(I, I_i)$ is the similarity computed by HE between images I and I_i.

Remark: One of the key advantage of HE over NBNN based methods [3, 23] is that it does not need to compute the Euclidean nearest neighbors of the descriptors, which is costly both in terms of memory (to store the raw SIFT descriptors) and efficiency. In contrast, HE efficiently achieves accurate matching based on the binary signatures, which in addition are compact in memory (8 bytes per descriptor).

Normalization of similarity scores. Figure 4 (top) shows the distribution of the scores produced by HE. One can observe that most of the scores are low, while few are high, due to the high discriminative power of this matching method.

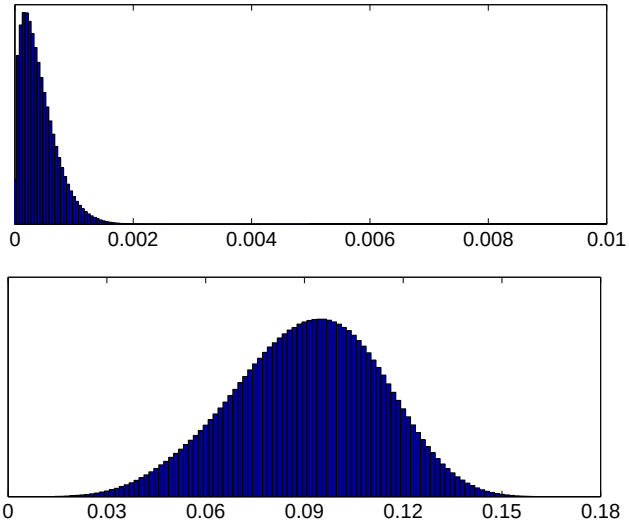

Figure 4: Distribution of similarity scores (a) before and (b) after power normalization (with $\alpha = 0.3$). Note the change in the scales. The scores are obtained for *trainval* set of PASCAL VOC 2007 dataset.

A similar observation was done for the Fisher Kernel [21]. Such a score distribution is not desirable for classification, because large values may generate some artifacts. In order to distribute the scores more evenly, we therefore adopt the power normalization method proposed to improve the Fisher Kernel [21]. It consists in applying the following component-wise function: $f(x) = x^\alpha$. Note that, in our case, the scores are all non-negative.

When optimizing the parameter α by cross-validation, we consistently obtain a value between 0.2 and 0.35. The values in this range provide comparable results, which suggests that this parameter can be set to a constant, e.g., $\alpha = 0.3$. Figure 4 (bottom) shows the distribution of images scores after power normalization with $\alpha = 0.3$, again computed on PASCAL VOC 2007 dataset. As one may observe, the power-law emphasizes the relative importance of low scores. Finally and similar to the Fisher Kernel, the vector is L2-normalized, producing the following final image representation:

$$\hat{I}_{\text{HE}} = \frac{[\text{HE}_{\text{sim}}(I, I_1)^\alpha \;\; \text{HE}_{\text{sim}}(I, I_2)^\alpha \;\; ... \;\; \text{HE}_{\text{sim}}(I, I_N)^\alpha]}{\sqrt{\Sigma_{i=1}^N \text{HE}_{\text{sim}}(I, I_i)^{2\alpha}}}, \quad (2)$$

where the denominator is computed such that the Euclidean norm of the final vector is 1.

Spatial Grid. The spatial pyramid matching proposed in [18] is a standard way to introduce some partial geometrical information in a bag-of-words representation. It consists in subdividing the image in a spatial grid and in computing histograms separately for each of the spatial regions thus defined. These spatial histograms are weighted according to the size of the region, normalized separately and then concatenated together to produce the final representation.

This idea is adapted to our HE-based representation. It is done by computing the HE similarities between each of the spatial regions and the training images. A noticeable difference is that the full training images are used to compute the similarity and not just the associated regions, because we observed that larger region gives slightly better results. The image is represented as 1×1 and 1×3 (three horizontal stripes) grids, that is 4 regions in total. Other methods usually draw 8 or 21 regions (add 2×2 or 4×4).

Another difference w.r.t. the method of [18] is that we train a linear SVM separately for each grid. Two SVMs are trained, one for 1×1 grid and another for 1×3 grid. The similarity scores of the three regions of 1×3 grid are stacked together to make $3N$ dimensional representation for training. The final classification scores are obtained as a weighted sum of the scores from both the classifiers. These weights are learned by cross-validating on the validation data.

4. EXPERIMENTS AND RESULTS

In this section, we first present some implementation details and then evaluate the proposed method on two challenging datasets for image classification: PASCAL VOC 2007 [8] and Caltech-256 [9].

4.1 Implementation Details

Only one type of feature is used in all our experiments, namely the SIFT descriptor computed on a dense grid. The descriptors are extracted from patches densely located with a spatial stride of 3 pixels on the image, under five scales. In [5], it is observed that such a dense sampling has a positive impact on classification accuracy. Also, it allows us to provide a consistent comparison of our method with several recent encodings evaluated in [5], and shows the interest of the variant of the SIFT descriptor introduced in Section 3.1. As we use dense features, *burstiness* handling [14] becomes more important as visual burst increases. Therefore in all the experiments we use burstiness regularization. For the sake of consistency, the vocabulary size is set $K = 4096$ for all our experiments with BOW and HE.

Key parameters. There are two important parameters in our method, namely the HE threshold (h_t) and the parameter α involved in the power-law component-wise normalization. The impact of these parameters on the performance is shown in Figure 5, on the PASCAL VOC 2007 benchmark. The best choice of h_t, as obtained by cross-validation for binary signatures of length 64, is a threshold between 20 to 24. Interestingly, these values are consistent with those used with HE [15] in an image retrieval context. As suggested in Section 3, the parameter α is not very sensitive in the range [0.2,0.35], and is therefore set to the constant $\alpha = 0.3$ for all the experiments.

4.2 PASCAL VOC 2007

Evaluation protocol. The PASCAL VOC 2007 dataset contains about 10,000 images split into *train*, *validation* and *test* sets. The objective is to perform the classification for 20 object categories. A 1-versus-rest linear SVM classifier is trained for each category and the performance is evaluated

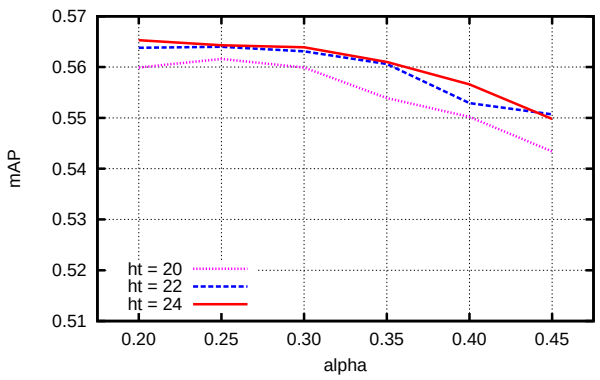

Figure 5: Impact of HE threshold (h_t) and α on mAP for test set of PASCAL VOC 2007.

in terms of average precision (AP) for each class. The overall performance is measured as the average of these APs, i.e., it is the mean average precision (mAP). We follow the standard practice of training on *train+validation* and testing on *test*. The cross-validation of the different parameters, in particular the threshold h_t and the C parameter of the SVM (regularization-loss trade off), is performed by training on the train set and testing on the validation set. The C parameter is validated for each class whereas h_t is validated for the dataset and finally fixed to $h_t = 22$.

The state of the art. The best results reported on this dataset using only the SIFT feature were obtained with the Fisher Kernel [21]. They report 58.3% with grid and 55.3% without grid. Their descriptor dimensionality is typically about $32K$ (or more) without spatial grid, and 8 times more with. In our case, the final representation is equal to the number of training images (i.e. 5011 here) and 4 times more when using the spatial grid.

Better results have been reported using more than one feature channel. For instance, the best classification method (by INRIA) in the original competition [8] obtained 59.4% using multiple feature channels and costly non-linear classifiers. Similarly, Kernel Codebook [24] and Yang et al [26] use many channels with soft assignment or sophisticated multiple Kernel learning to achieve mAP=60.5% and 62.2% respectively. To our knowledge, the best result ever reported on this dataset was achieved by Harzallah et al. [10], who combine very costly object localization with their classification system.

Since different methods employ varying experimental settings (single/multiple feature, various sampling density and codebook sizes), it is difficult to have a consistent comparison. This issue is addressed by Chatfield et al. [5], who perform an independent evaluation of the recent encoding methods. With a consistent setting of parameters for all methods, the Fisher Kernel improves to 61.69% and the super vector coding [29] achieves a score of 58.13% (reported 64.0%). In Table 1, we refer to those results for a fair comparison. All the results reported in this table are, except for HE, HE* and FK*, from this paper [5]. Other elements like

Methods	Codebook	Spat grid	SVM	mAP
FK	256	yes	Lin	61.69
FK*	256	yes	Lin	62.22
SV	1k	yes	Lin	58.13
BOW	4k	yes	Lin	46.54
BOW	4k	yes	Chi	53.42
LLC	4k	yes	Lin	53.79
LLC	4k	yes	Chi	53.47
LLC-F	4k	yes	Lin	55.87
KCB	4k	yes	Chi	54.60
HE	4k	no	Lin	**53.98**
HE	4k	yes	Lin	**56.68**
HE*	4k	no	Lin	**56.31**
HE*	4k	yes	Lin	**58.34**
HE* + FK*	4k, 256	no	Lin	**60.84**
HE* + FK*	4k, 256	yes	Lin	**62.78**

Table 1: Image classification results using PASCAL VOC 2007 dataset with consistent setting of parameters. [FK: Fisher Kernel, FK*: Fisher Kernel with our SIFT variant, SV: super vector coding, BOW: bag of words, LLC: locally constrained linear coding, LLC-F: LLC with with original+left-right flipped training images, KCB: Kernel codebook, HE: Hamming Embedding similarity, HE*: HE with our SIFT variant; Lin/Chi: linear/χ^2 Kernel map].

vocabulary size, classifiers used, spatial grid are mentioned in the table.

Impact of our SIFT's variant. The method denoted by HE* is our Hamming Embedding similarity approach combined with the proposed SIFT variant detailed in Section 3.1. Similarly, FK* represents the Fisher Kernel combined with our SIFT variant. A considerable improvement of around 2% is observed by using HE* over HE both with and without spatial grid. The difference is only that HE uses original SIFT descriptors. As one can observe the variant also improves in case of Fisher Kernel, FK (original SIFT) and FK* use exactly the same parameters otherwise.

HE classification. Our method, HE* with spatial grid, performs better than all the methods except the improved Fisher Kernel. With original sift descriptors (HE) mAP of 56.68% is obtained, which again compares favorably to most of the methods. Even without spatial grid HE* achieves a competitive mAP of 56.31%, while relying on a matching-based method. To our knowledge, it is the first method of that kind that approaches the best coding method FK on the PASCAL VOC 2007 benchmark.

Moreover, one would expect such a matching based method to be complementary with the coding based methods, even by using the same local descriptors in input. To confirm this, we combine our Hamming Embedding method (HE*) with the Fisher Kernel (FK*), using a late fusion of confidence scores. Doing so, we obtain a mAP of 60.84% and 62.78% without and with spatial grid respectively. Class-wise APs are reported in Table 2 for our approach and its combination with the Fisher Kernel. This combination improves the results for most of the categories.

Method/Class	HE*	FK	FK*	HE* + FK*
Aeroplane	76.50	78.97	**80.92**	80.75
Bicycle	62.70	57.43	67.39	**67.70**
Bird	50.23	51.94	**57.10**	56.77
Boat	68.62	**70.92**	69.01	69.86
Bottle	28.40	30.79	33.17	**33.77**
Bus	63.35	**72.18**	69.08	69.68
Car	79.37	79.94	80.42	**81.27**
Cat	61.20	61.35	61.51	**62.71**
Chair	52.52	55.98	55.43	**56.26**
Cow	45.24	49.61	49.89	**51.67**
DiningTable	52.85	58.40	58.71	**59.22**
Dog	47.11	44.77	48.98	**49.96**
Horse	77.06	78.84	79.56	**80.12**
Motorbike	64.27	**70.81**	70.02	70.27
Person	83.18	84.96	84.56	**85.20**
PottedPlant	32.46	31.72	34.65	**37.00**
Sheep	41.29	**51.00**	49.80	46.71
Sofa	50.15	**56.41**	55.08	55.54
Train	77.02	80.24	81.36	**81.81**
TVmonitor	53.24	57.46	57.76	**57.80**
mAP	**58.34**	**61.69**	**62.22**	**62.78**

Table 2: Image classification results per class using PASCAL VOC 2007 dataset. Again, recall that FK* is the improved Fisher Kernel [21] combined with our better SIFT variant.

4.3 Caltech-256

Evaluation protocol. The Caltech-256 dataset contains approximately 30K images falling into 256 categories. Each category contains at least 80 images. There is no provided division of dataset into train and test though. However, the standard practice is to split the dataset into train and test sets and repeat each experiment multiple times with different splits. We run experiments with different numbers of training images per category: $ntrain = 15,\ 30,\ 45,\ 60$. The remaining images are used for testing. Validation is done on 5 images from train set by training on $ntrain - 5$ images. The validated h_t is equal to 20 for this dataset. We run experiments for five random splits for each $ntrain$. Again a 1-vs-rest linear SVM is trained for each class. We report the average classification accuracy (standard practice) across all classes.

Results. Table 3 compares our results with the best reported ones. We divide the methods as matching or coding based, all of them use only SIFT feature. Compared to PASCAL VOC, matching-based methods perform comparatively better on Caltech-256, outperforming many coding approaches such as Kernel-Codebook [24], Sparse-Coding [27], Standard FK [20] and the baseline by the authors of Caltech 256 dataset [9]. Overall, our method outperforms all the matching and coding based approaches. Only the improved Fisher Kernel [21] and LLC [25] perform better in the case of 15 training images. This is not surprising, because in our case the dimensionality of the final representation is equal to the number of training images. With more training images, the dimensionality of our descriptor increases and leads to the best results.

	Methods/ntrain	15	30	45	60
Coding methods	**Baseline [9]**	-	34.10	-	-
	Kernel Codebook [24]	-	27.17	-	-
	EMK [2]	23.20	30.50	34.40	37.60
	SCSPM [27]	27.70	34.02	37.50	40.14
	Standard FK [20]	25.60	29.00	34.90	38.50
	Improved FK [21]	**34.70**	40.80	45.00	47.90
	LLC [25]	-	41.19	-	47.68
Matching-based	**Kim et al. [17]**	-	36.30	-	-
	NBNN [3]	-	38.00	-	-
	Duchenne et al. [7]	-	38.10	-	-
	HE*	32.49	**41.80**	**46.69**	**49.83**

Table 3: Comparison of HE similarity-based representation with the state-of-the-art on Caltech-256.

5. CONCLUSIONS

In this paper, we have presented a novel approach to image classification based on a matching technique. It consists in combining the Hamming-Embedding similarity-based matching method with a similarity space encoding, which subsequently allows the use of a linear SVM. This method is efficient and achieves state-of-the-art classification results on two reference image classification benchmarks: the PASCAL VOC 2007 and Caltech-256 datasets. Moreover, it is shown to be complementary with the other best classification method based, namely the Fisher kernel. To our knowledge, this method is the first matching-based approach to provide such competitive results. We believe that the flexibility offered by this framework is likely to be extended, in particular for a better integration of the geometrical constraints. As a secondary contribution, we have proposed an effective variant of the SIFT descriptor, which gives a slight yet consistent improvement on classification accuracy. Its interest has been validated with the Fisher Kernel.

Acknowledgments. This work was done within the Quaero project, funded by Oseo, French agency for innovation.

6. REFERENCES

[1] R. Behmo, P. Marcombes, A. Dalalyan, and V. Prinet. Towards optimal naive bayes nearest neighbors. In *ECCV*, September 2010.

[2] L. Bo and C. Sminchisescu. Efficient match kernels between sets of features for visual recognition. In *NIPS*, 2009.

[3] O. Boiman, E. Shechman, and M. Irani. In defense of nearest neighbor based image classification. In *CVPR*, June 2008.

[4] B. Caputo and L. Jie. A performance evaluation of exact and approximate match kernels for object recognition. *Electronic Letters on Computer Vision and Image Analysis*, 8(3):15–26, 2009.

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

[6] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR*, June 2009.

[7] O. Duchenne, A. Joulin, and J. Ponce. A graph-matching kernel for object categorization. In *ICCV*, September 2011.

[8] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results, 2007.

[9] G. Griffin, A. Holub, and P. Perona. Caltech-256 object category dataset. Technical Report 7694, California Institute of Technology, 2007.

[10] H. Harzallah, F. Jurie, and C. Schmid. Combining efficient object localization and image classification. In *ICCV*, September 2009.

[11] T. Jaakkola and D. Haussler. Exploiting generative models in discriminative classifiers. In *NIPS*, 1998.

[12] M. Jain, H. Jégou, and P. Gros. Asymmetric hamming embedding: taking the best of our bits for large scale image search. In *ACM Multimedia*, November 2011.

[13] H. Jégou, M. Douze, and C. Schmid. Hamming embedding and weak geometric consistency for large scale image search. In *ECCV*, October 2008.

[14] H. Jégou, M. Douze, and C. Schmid. On the burstiness of visual elements. In *CVPR*, June 2009.

[15] H. Jégou, M. Douze, and C. Schmid. Improving bag-of-features for large scale image search. *IJCV*, 87(3):316–336, February 2010.

[16] H. Jégou, M. Douze, C. Schmid, and P. Pérez. Aggregating local descriptors into a compact image representation. In *CVPR*, June 2010.

[17] J. Kim and K. Grauman. Asymmetric region-to-image matching for comparing images with generic object categories. In *CVPR*, June 2010.

[18] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: spatial pyramid matching for recognizing natural scene categories. In *CVPR*, June 2006.

[19] D. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 60(2):91–110, 2004.

[20] F. Perronnin and C. R. Dance. Fisher kernels on visual vocabularies for image categorization. In *CVPR*, June 2007.

[21] F. Perronnin, J.Sánchez, and T. Mensink. Improving the fisher kernel for large-scale image classification. In *ECCV*, September 2010.

[22] J. Sivic and A. Zisserman. Video Google: A text retrieval approach to object matching in videos. In *ICCV*, pages 1470–1477, October 2003.

[23] T. Tuytelaars, M. Fritz, K. Saenko, and T. Darrel. The NBNN kernel. In *ICCV*, September 2011.

[24] J. van Gemert, C. Veenman, A. Smeulders, and J. Geusebroek. Visual word ambiguity. *PAMI*, 32(7):1271–1283, July 2010.

[25] J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, and Y. Gong. Locality-constrained linear coding for image classification. In *CVPR*, June 2010.

[26] J. Yang, Y. Li, Y. Tian, L. Duan, and W. Gao. Group sensitive multiple kernel learning for object categorization. In *ICCV*, September 2009.

[27] J. Yang, K. Yu, Y. Gong, and T. Huang. Linear spatial pyramid matching using sparse coding for image classification. In *CVPR*, pages 1794–1801, 2009.

[28] J. Zhang, M. Marszalek, S. Lazebnik, and C. Schmid. Local features and kernels for classification of texture and object categories: A comprehensive study. *IJCV*, 73:213–238, June 2007.

[29] X. Zhou, K. Yu, T. Zhang, and T. S. Huang. Image classification using super-vector coding of local image descriptors. In *ECCV*, September 2010.