Asymmetric Hamming Embedding

Mihir Jain, Hervé Jégou, Patrick Gros

▶ **To cite this version:**

Mihir Jain, Hervé Jégou, Patrick Gros. Asymmetric Hamming Embedding. ACM Multimedia, Nov 2011, Scottsdale, United States. inria-00607278v1

HAL Id: inria-00607278

https://inria.hal.science/inria-00607278v1

Submitted on 8 Jul 2011 (v1), last revised 8 Jul 2011 (v2)

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.

Asymmetric Hamming Embedding

Taking the best of our bits for large scale image search

Mihir Jain
INRIA Rennes
mihir.jain@inria.fr

Hervé Jégou
INRIA Rennes
first.last@inria.fr

Patrick Gros
INRIA Rennes
patrick.gros@inria.fr

ABSTRACT

This paper proposes an asymmetric Hamming Embedding scheme for large scale image search based on local descriptors. The comparison of two descriptors relies on an vector-to-binary code comparison, which limits the quantization error associated with the query compared with the original Hamming Embedding method. The approach is used in combination with an inverted file structure that offers high efficiency, comparable to that of a regular bag-of-features retrieval systems. The comparison is performed on two popular datasets. Our method consistently improves the search quality over the symmetric version. The trade-off between memory usage and precision is evaluated, showing that the method is especially useful for short binary signatures.

1. INTRODUCTION

Large scale image search is still a very active domain. The task consists in finding in a large set of images the ones that best resemble the query image. Typical applications include finding searching images on web [15], location [13] or particular object [11] recognition, or copy detection [8]. Earlier approaches were based on global descriptors such as color histograms or GIST [12]. These are sufficient in some contexts [3], such as copy detection, where most of the illegal copies are very similar to the original image. However, global description suffer from well-known limitations, in particular they are not invariant to significant geometrical transformations such as cropping. Here we focus on the bag-of-words (BOW) framework [14] and its extension [6], where local descriptors are extracted from each image [9] and used to compare images.

The BOW representation of images was proved be very discriminant and efficient for image search on millions of images [5, 11]. Different strategies have been proposed to improve it. For instance, [11] improves the efficiency in two ways. Firstly, the assignment of local descriptors to the so-called visual words is much faster thanks to the use of a hierarchical vocabulary. Secondly, by considering large vocabularies (up to 1 million visual words), the size of the inverted lists used for indexing is significantly reduced. Accuracy is improved by a re-ranking stage performing spatial verification [9], and by query expansion [1], which exploits the interaction between the relevant database images.

Another way to improve accuracy consists in incorporating additional information on descriptors directly in the inverted file. This idea was first explored in [5], where a richer descriptor representation is obtained by Hamming Embedding (HE) and weak geometrical consistency [5]. HE, in particular, was shown successful in different contexts [15], and improved in [6, 7]. However, this technique has a drawback: each local descriptor is represented by relatively large signatures, typically ranging from 32 [15] to 64 bits [6].

In this paper, we propose to improve HE in order to better exploit the information conveyed by the binary signature. This is done by exploiting the observation, first made in [2], that the query should not be approximated. We therefore adapt the voting method to better exploit the precise query location instead of the binarized query vector. This requires, in particular, two regularization steps used to adjust the dynamic of the local query distribution. This leads to an improvement over the reference symmetric HE scheme. As a complementary contribution, we evaluate how our approach trades accuracy against memory with smaller number of bits. To our knowledge, such a comparison has never been published.

The paper is organized as follows. The datasets representing the application cases and the evaluation protocol are introduced in Section 2. Section 3 briefly describes the most related works: BOW and HE. Our asymmetric method is introduced in Section 4. Finally, experiments in Section 5 compare the performance of our asymmetrical method with the original HE, and provides a comparison with the state of the art on image search. Its shows a significant improvement: we obtain a mean average precision of 70.4% on the Oxford5K Building dataset before spatial verification, i.e., +4% compared with the best concurrent method.

2. EVALUATION DATASETS

This section introduces the datasets used in our experiments, as well as the measures of accuracy used to evaluate the different methods. These datasets reflect two application use-cases for which our method is relevant, namely place and object recognition. They are widely used to evaluate image search systems.

Oxford5K and Paris. These two datasets of famous building in Oxford and Paris contain 5,062 and 6,412 images, respectively. We use Paris as an independent learning set to estimate the parameters used by our method. The quality is measured on Oxford5K by mean average precision (mAP), as defined in [13]: for each query image we obtain a precision/recall curve, and compute its average precision (the area under the curve). The mAP is then the mean for a set of queries.

INRIA Holidays. This dataset contains 1491 images of personal holiday photos, partitioned into 500 groups, each of which

Pre-print (author) version.
To be published in ACM Multimedia 2011.

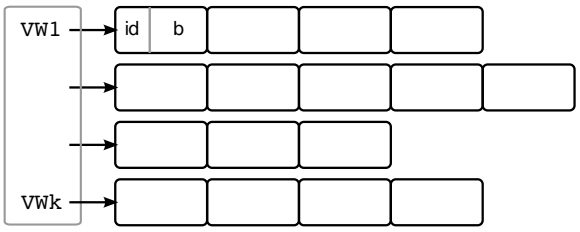

Figure 1: Overview of the HE indexing structure: it is a modified inverted file. Each inverted list is associated with a visual word. Each local descriptor is stored in a cell containing both the image identifier and a binary signature (from 4 to 64 bits in our paper). This indexing structure is also used in our AHE method: only the score similarity is modified.

represents a distinct scene, location or object. The first image of each group is the query image and the correct retrieval results are the other images of the group. Again, the search quality is measured by the mAP, see [13, 5] for details. A set of images from Flickr is used for learning the vocabulary, as done in [5].

Flickr1M. In order to evaluate the behavior of our method on a large scale, we have used a set of up to one million images. More precisely, we have used the descriptors shared online[1] by Jegou et al., which were downloaded from Flickr and described by the same local descriptor generation procedure as the one used for Holidays. This dataset is therefore merged with Holidays and the mAP is measured using the Holidays ground-truth, as in [7].

The recall@R measure is used for this large scale experiment. It measures, at a particular rank R, the ratio of relevant images ranked in top R positions. [3] states that it is a good measure to evaluate the filtering capability of an image search system, in particular if the large scale image search is followed by a precise spatial geometrical stage, as classically done in the literature.

3. RELATED WORK

3.1 Bag-of-features representation

The BOW framework [14] is based on local invariant descriptors [10, 9] extracted from covariant regions of interest [10]. It matches small parts of images and can cope with many transformations, such as scaling, local changes in illumination and cropping.

The feature extraction is performed in two steps: detecting regions of interest with the Hessian-Affine detector [10], and computing SIFT descriptors for these regions [9]. We have used the features provided by the authors for all the datasets.

The fingerprint of an image is obtained by quantizing the local descriptors using a nearest-neighbor quantizer, produced the so-called *visual words* (VW). The image is represented by the histogram of VW occurrences normalized with the L2 norm. A *tf-idf* weighting scheme is applied [14] to the k components of the resulting vector. The similarity measure between two BOW vectors is, most often, cosine similarity. The visual vocabulary of the quantizer is produced using k-means. It contains a large number k of visual words. In this paper, $k = 20,000$ for the sake of consistency with [5] and [6]. Therefore, the fingerprint histograms are sparse, making queries in the inverted file efficient.

3.2 Hamming embedding

The Hamming Embedding method of [5] is a state of the art method extension of BOW, where a better representation of the images is obtained by adding a short signature that refines the representation of each local descriptor. In this approach, a descriptor x is represented by a tuple $(q(x), b(x))$, where $q(x)$ is the visual word and $b(.)$ is a binary signature of length m computed from the descriptors to refine the information provided by $q(x)$. Two descriptors are assumed to match if

$$\begin{cases} q(x) = q(y) \\ h\left(b(x), b(y)\right) = \sum_{i=1..m} |b_i(x) - b_i(y)| \leq h_t \end{cases}, \quad (1)$$

where $h(b, b')$ is the Hamming distance between binary vectors $b = [b_1, .., b_m]$ and $b' = [b'_1, .., b'_m]$, and h_t is a fixed threshold. The image score is obtained as the sum [5] or weighted sum [7] of the distances of the matches satisfying (1), then normalized as BOW.

As BOW, the method uses an inverted file structure, which is modified to incorporate the binary signature, as illustrated by Figure 1. The matches associated with the query local descriptors are retrieved from the structure and used as follows.

- Find the nearest centroid of the query descriptor x, producing quantized indexes $q(x)$, i.e., the visual word (VW). The entries of the inverted list associated with $q(x)$ are visited.
- Compute the vector $\mathbf{Q} \times x = b^* = [b^*_1(x), \ldots, b^*_m(x)]$ associated with descriptor x, where $\mathbf{Q}$ is a rotation matrix for which only the first m rows are kept so that $\mathbf{Q} \times x \in \mathbb{R}^m$.
- The binary signature is obtained by comparing each component $b^*_i, i = 1..m$ with a threshold $\tau_{q(x),i}$. This amounts to selecting $b_i = 1$ if $b^*_i - \tau_{q(x),i} > 0$, else $b_i = 0$. The thresholds $\tau_{c,i}$ are the median values of b^*_i measured on an independent learning set for all VWs c and all bit components i.
- Only the database descriptors satisfying Equation 1 make a vote for the corresponding image, i.e., they vote only if their Hamming distance is below a pre-defined threshold h_t. The vote's score is 1 in [5]. Scoring with a function of the distance improves the results [6]. We therefore adopt this choice.
- All images scores are finally normalized.

Additionally, we consider in this paper two techniques [6] that improve the results. First, multiple assignment (MA) reduces the number of matches that are missed due to incorrect quantization indexes. Second, the so-called *burstiness* (denoted by "burst") handling method regularizes the score associated with each match, to compensate the bursty statistics of regular patterns in images.

4. ASYMMETRIC HAMMING EMBEDDING

This section introduces our approach. It is inspired by the works of Dong [2] and Gorda [4], where the use of asymmetric distances was investigated in the context of Locality Sensitive hashing. This method has to be significantly adapted in our context. Using the distance to hyperplanes may suffice for pure nearest neighbor search, where the objective is the find the Euclidean k-nearest neighbor of a given query [2]. However, in our case, this is not sufficient, because computed distances are used as match quality measurements. Our goal is therefore to provide a soft weighting strategy that better exploits the confidence measures of all matches to produce the aggregated image scores.

Intra-cell distance regularization. We first adapt the local distances so that they become more comparable for different visual words. This is done, in our AHE scheme, by computing the standard deviation σ_c of the distance $b^*_i - \tau_{c,i}$ to the separating hyperplanes for each of the k visual words c. This estimation is carried out using a large set of vectors from an independent learning set. We used 50M Flickr descriptors for Holidays and all the

[1] `http://lear.inrialpes.fr/people/jegou/data.php`

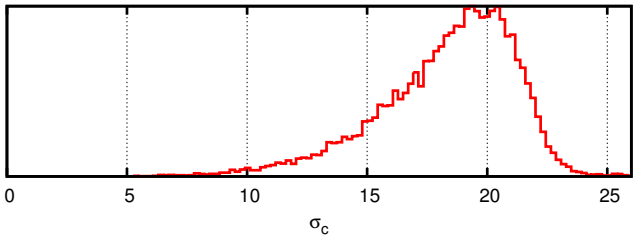

Figure 2: Empirical probability distribution function of σ_c measured for all visual words. The large variation of density across cells shows the need for a per-cell variance regularization.

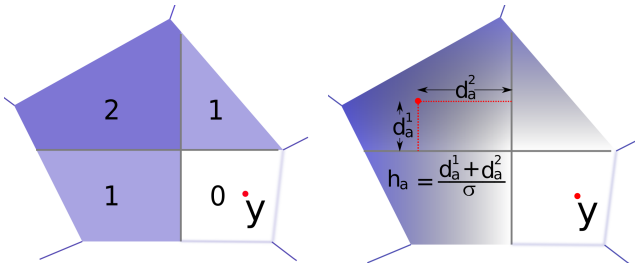

Figure 3: Illustration of HE and AHE for binary signatures of length 2. In the symmetric case, only three distances are possible (0, 1 or 2) between query and database descriptor y. AHE gives a continuous distance (reflected by the intensity of blue).

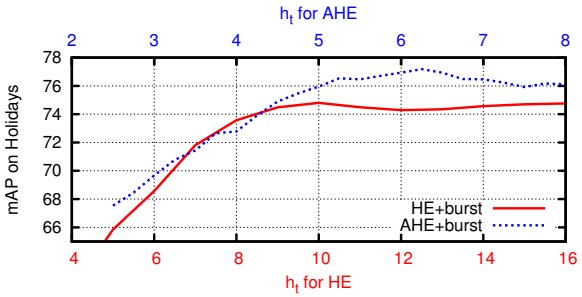

Figure 5: Impact of the threshold on accuracy ($m = 32$ bits). Note that the ranges for h_t differ for HE (Hamming distance) and AHE (derived from normalized distance to hyperplanes).

	Oxford5K	Holidays
BOW [14]	40.3	-
BOW+soft MA [14]	49.3	-
HE+MA5 [6]	61.5	77.5
HE+burst [5]	*64.5*	*78.0*
HE+burst+MA5 [5]	*67.4*	*79.6*
AHE+burst	66.0	79.4
AHE+burst+MA5	69.8	**81.9**
AHE+burst+MA10	**70.4**	81.7

Table 1: State of the art. For [5], we report *in italics* the results obtained with the best descriptors ([5] reports inferior results with different descriptors and with geometrical information only).

descriptors from Paris for Oxford5K. The standard deviation is either computed component-wise (one per bit dimension) of for the whole cell. In our case we chose the simple choice of estimating a single parameter per cell used for all bits (isotropic assumption).

As observed in Figure 2, the standard deviations significantly vary from one cell to another. It is then worth obtaining more comparable values when using distances as quality measurements.

Distance to hyperplanes. In the symmetric version, the query projected by $\mathbf{Q}$ is binarized and compared with the database descriptors. We instead compute the distance between the projected query ($b^*(x) = \mathbf{Q} \times x$) and the database binary vectors that lie in the same cell (associated with $q(x)$). The "distance" between the i^{th} component of $b^*(x)$ and the binary vector $b(y)$ is given by

$$d_a^i\left(b_i^*(x), b_i(y)\right) = |b_i^*(x) - \tau_{q(x),i}| \times |b_i(x) - b_i(y)|\,. \quad (2)$$

This quantity is zero when x is on the same side of the hyperplane associated with the i^{th} component. The distances are added for all the m components to get an asymmetric "distance" between a query descriptor x and a database descriptor y, defined as

$$h_a\left(b^*(x), b(y)\right) = \frac{1}{\sigma_{q(x)}} \times \sum_{i=1..m} d_a^i\left(b_i^*(x), b_i(y)\right). \quad (3)$$

The descriptors are assumed to match if $h_a\left(b^*(x), b(y)\right) \leq h_t$, as for the symmetric version. For a given query x, the values $|b_i^*(x) - \tau_{q(x),i}|$ are precomputed before being compared to the database vectors. The similarity is penalized according to the distance from the hyperplane in the embedded Hamming space, providing improved measurements, as illustrated by Figure 3. In the symmetric case, it does not matter how far b_i^* lies from $\tau_{q(x),i}$. In contrast, the distance is a continuous function in our method.

Score weighting. Similar to what is done in [6] for the symmetric case, the distance obtained by Equation 3 is used to weight the voting score. In [6] weights are obtained as a gaussian function of Hamming distance. Here the weights are simply the difference $h_t - h_a\left(b^*(x), b(y)\right)$ between the threshold and the normalized "distance". We also apply the burstiness regularization method of [6]. As we will show in Section 5, its impact is very important in our case because the aforementioned variance regularization does not sufficiently balance the amount of score received by the different *query* vectors, leading individual descriptors from the query image to have a very different impact in the final score. The burstiness regularization effectively addresses this issue.

5. EXPERIMENTS

Search quality: HE *vs* AHE. Figure 4 evaluates our AHE method introduced in Section 4 against the original HE one, for varying numbers of bits. For both HE and AHE, we report the results obtained with the best threshold. As shown by Figure 5, the performance is stable around this best value.

Using the asymmetric version significantly improves the results, especially for short signatures. As stated in Section 4, the burstiness regularization of [6] is important in our case: without it AHE only achieves a slight improvement for short signatures.

Observe the important trade-off between the search accuracy and the signature length: using more bits clearly helps. However it is important to keep this signature short so that database images remain indexed in memory. Multiple Assignment helps in both cases, at the cost of increased query time. Unless specified, we used MA to the 5 nearest visual words, denoted by MA5.

Comparison with the state of the art. Table 1 compares our results with, to our knowledge, the best ones reported in the literature. Our approach clearly outperforms the state of the art on both Holidays and Oxford5K. Interestingly, for symmetric HE, our results (*in italics*) on Oxford5K are better than those reported in [6] with a geometry check. This is because on these datasets, the

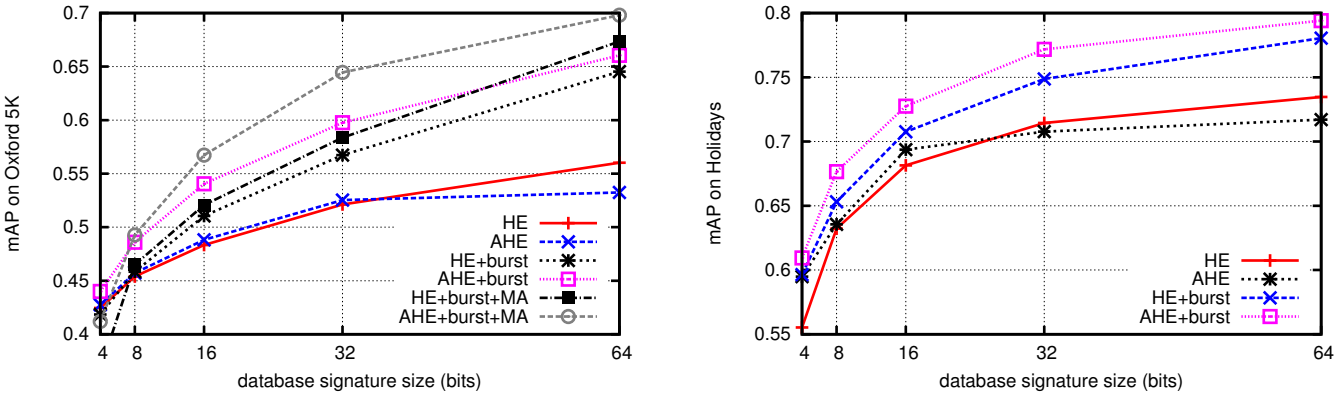

Figure 4: HE vs AHE: Trade-off between memory usage (per descriptor) and search quality.

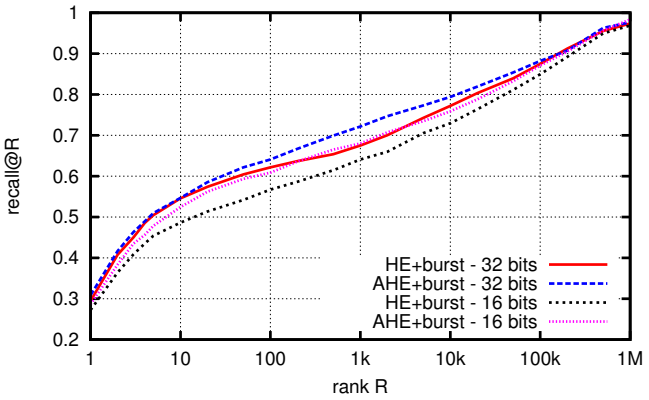

Figure 6: Search quality on a large scale (1 million images): Holidays merged with Flickr1M.

features used in [6] are not as good as those we used here. We only include in our comparison the results reported with learning done on an independent dataset itself. Some papers shows that learning on the test set itself improves the results, as to be expected. But as stated in [7] such results do not properly reflect the expected accuracy when using the system on a large scale.

Large scale experiments. As shown in Figure 6, the filtering capability of AHE is better than HE: the recall@R measure is almost as good for AHE with 16 bits as HE with 32 bits. Equivalently, the performance is much better for a given memory usage.

The complexity of the method is increased compared to the original symmetric method. In both HE and AHE, the vector has to be projected. The main difference appears in the similarity computation, which is a simple `XOR` operation following by a bit count in HE, while we need to add pre-computed floating point values to get h_a in Equation 3. As a result, on 1 million images the search time is roughly 1.7 times slower in the asymmetric case, on average, compared to [6]: on average searching in one million images (using 64 bits) with AHE it takes 2.9s on one processor core, against 1.7s for HE.

Acknowledgments:

This work was realized as part of the Quaero Project, funded by OSEO, French State agency for innovation.

6. CONCLUSION

This paper shows that a vector-to-binary code comparison significantly improves the state-of-the-art Hamming Embedding technique by reducing the approximation made on the query. This is done by exploiting the vector-to-hyperplane distances. The improvement is obtained at no additional cost in terms of memory. As a result, we improve the best results ever reported on two popular image search benchmarks before geometrical verification.

7. REFERENCES

[1] O. Chum, J. Philbin, J. Sivic, M. Isard, and A. Zisserman. Total recall: Automatic query expansion with a generative feature model for object retrieval. In *ICCV*, October 2007.

[2] W. Dong, M. Charikar, and K. Li. Asymmetric distance estimation with sketches for similarity search in high-dimensional spaces. In *SIGIR*, July 2008.

[3] M. Douze, H. Jégou, H. Singh, L. Amsaleg, and C. Schmid. Evaluation of GIST descriptors for web-scale image search. In *CIVR*, July 2009.

[4] A. Gordo and F. Perronnin. Asymmetric distances for binary embeddings. In *CVPR*, June 2011.

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

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

[7] H. Jégou, M. Douze, and C. Schmid. Improving bag-of-features for large scale image search. *IJCV*, 2010.

[8] J. Law-To, L. Chen, A. Joly, I. Laptev, O. Buisson, V. Gouet-Brunet, N. Boujemaa, and F. Stentiford. Video copy detection: a comparative study. In *CIVR*, 2007.

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

[10] K. Mikolajczyk and C. Schmid. Scale and affine invariant interest point detectors. *IJCV*, 60(1):63–86, 2004.

[11] D. Nistér and H. Stewénius. Scalable recognition with a vocabulary tree. In *CVPR*, pages 2161–2168, June 2006.

[12] A. Oliva and A. Torralba. Modeling the shape of the scene: a holistic representation of the spatial envelope. *IJCV*, 42(3):145–175, 2001.

[13] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman. Object retrieval with large vocabularies and fast spatial matching. In *CVPR*, June 2007.

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

[15] Z. Wu, Q. Ke, M. Isard, and J. Sun. Bundling features for large scale partial-duplicate web image search. In *CVPR*, pages 25–32, 2009.