

## Bag-of-colors for improved image search

Christian Wengert, Matthijs Douze, Hervé Jégou

► **To cite this version:**

Christian Wengert, Matthijs Douze, Hervé Jégou. Bag-of-colors for improved image search. MM 2011 - 19th ACM International Conference on Multimedia, Nov 2011, Scottsdale, United States. ACM, pp.1437-1440, 2011, <10.1145/2072298.2072034>. <inria-00614523>

**HAL Id: inria-00614523**

**<https://hal.inria.fr/inria-00614523>**

Submitted on 12 Aug 2011

**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.

# Bag-of-colors for improved image search

Christian Wengert  
kooaba AG  
Förrlibuckstrasse 178  
CH-8005 Zürich

Matthijs Douze  
INRIA Grenoble  
655 avenue de l'Europe  
F-38334 Saint Ismier Cedex

Hervé Jégou  
INRIA Rennes  
Campus de Beaulieu  
F-35042 Rennes Cedex

## ABSTRACT

This paper investigates the use of color information when used within a state-of-the-art large scale image search system. We introduce a simple yet effective and efficient color signature generation procedure. It is used either to produce global or local descriptors. As a global descriptor, it outperforms several state-of-the-art color description methods, in particular the bag-of-words method based on color SIFT. As a local descriptor, our signature is used jointly with SIFT descriptors (no color) to provide complementary information. This significantly improves the recognition rate, outperforming the state of the art on two image search benchmarks. We will provide an open source package of our signature.

## 1. INTRODUCTION

We consider the problem of content-based image retrieval, for applications such as object recognition or similar image retrieval. This problem requires to produce a description of the image, typically a fixed-size vector. This description must be discriminative, but sufficiently invariant to handle the transformations the image may have undergone (cropping, different viewpoints, rotation, etc).

Most of the recent state-of-the-art large scale image search systems [7, 3] rely on local descriptors, in particular the SIFT descriptors [4] and its variants. They are extracted from regions of interest using a scale-invariant detector [4, 6, 5]. These descriptors are often used jointly with the the bag-of-words (BOW) framework [11], which produces a single vector from a set of local descriptors.

Surprisingly and despite its assessed relevance for image classification [13], most of the recent papers on large scale image search ignore color information. The best results reported for this task on popular benchmarks, such as the UKB [7] or Holidays datasets [1], which both contain colorful images, are reported for techniques based on grey-level SIFT descriptors only.

**Related work on color descriptors:** The body of literature dealing with color descriptors is vast. Some of these descriptors have been normalized in the MPEG7 standard. The color version of the GIST descriptor [8] is widely used for scene recognition. For the sake of conciseness, two color signatures will be used as reference, namely color GIST [8] and color SIFT [13]. The color SIFT descriptor is the counterpart of SIFT [4] with color integrated. Its

best variant [13] is the OpponentSIFT [14]. Similar to SIFT, it describes regions of interest extracted by a detector. It has mainly be considered for image and video classification tasks. For this reason, it is used in conjunction with the BOW aggregation framework [11]. However, this paper shows that much better results can be achieved using these descriptors by combining them with the state-of-the-art local descriptor matching method of [3].

The main contribution of our paper is a simple color description method, which is used to produce either global or local descriptors. In the first case (global), computing the image signature is much faster than producing a BOW vector with [13]: the main bottleneck is the computation of a color histogram, which is fast. Most importantly, improved search results are obtained on the evaluation datasets compared to other global descriptors.

The paper is organized as follows. Section 2 introduces our color signature generation procedure and analyzes the impact of several parameters and variants. As a global descriptor, our signature outperforms other global descriptors, including BOW generated from SIFT or color SIFT descriptors. Section 3 explains how the signature is used to describe local patches. Our color signature used jointly with SIFT descriptors outperforms the state of the art for the same memory usage and complexity. It is also better than color SIFT used within a comparable search system.

## 2. BAG-OF-COLORS SIGNATURES

In this section, we propose a method to extract a color signature, called bag-of-colors (BOC). This signature is extracted either for the whole image, producing a global descriptor, or from a patch (extracted by a region detector [6, 5]), in which case we obtain local color descriptors.

Since the method relies on computing a color histogram, the proposed signature is conceptually close to the work of [12]. However it departs from this method in several ways that drastically improve the recognition results. First, instead of using a regular color partition as in [12], the color codebook is learned on a set of real-world images. Most importantly, the histogram is constructed and normalized in a way inspired by the successful ingredients of the BOW framework [11] and recent improvement proposed for the Fisher kernel [10], but here applied to color components. As a result, the signature is more robust and avoids the typical problem of [12], where the most frequent colors dominate the other colors in the final representation and deteriorate the performance.

Hereafter, we introduce the signature generation procedure, analyze its results, and finally give a comparison with a BOW approach which shows that our signature provides consistently better results.

### 2.1 Color descriptor pipeline

Our color signature generation pipeline includes several steps. Different choices of parameters are investigated and evaluated.

**Color space:** Similar to most color descriptors, we use the CIE-



Figure 1: Color dictionaries learned by our procedure for  $k_c = 16, 32, 64$ . *Best viewed in color.*

Lab<sup>1</sup> color space in all our methods. This space is preferred to, e.g., the opponent space because it is more consistent with the Euclidean space structure, in particular the color dictionary generation procedure, which is based on k-means.

**Color codebook:** Our method needs a color dictionary  $\mathcal{C} = \{c_1, \dots, c_{k_c}\}$ , defined as a set of  $k_c$  colors (often referred to as a *palette*). We first consider building a codebook by quantizing each L, a and b components separately and regularly in 4 or 8 bins, leading to  $k_c = 64$  or 512 colors in total.

Then, we also learn a color dictionary that is more adapted to natural images. It is obtained by selecting and clustering typical colors from real-world images. In order to reduce the impact of some images containing large uniformly colored areas, we propose the following procedure in order to collect typical and unique colors for producing the color dictionary.

- 1) Collect 10000 random images from Flickr.
- 2) Resize each image to  $256 \times 256$  pixels, convert it to CIE-Lab and split it in blocks of  $16 \times 16$  pixels (i.e., 256 blocks in total).
- 3) For each block, find the most occurring color. Ties are randomly resolved. If this color corresponds to less than 5 occurrences (out of 256), then select an arbitrary color from the block.
- 4) At this point, we have extracted 256 Lab colors per image. The set of  $256 \times 10000$  colors from all images is clustered using a k-means algorithm, producing  $k_c$  Lab colors palette.

The palettes learned by our procedure are shown in Figure 1 for different values of  $k_c$ . Some colors (e.g., dark grey/brown) are more represented than others in the palette, due to the unbalanced color statistic of natural images.

**Color signature:** The first step of our color signature method consists in computing the histogram of colors in the image for the fixed color codebook  $\mathcal{C}$ . This is done by first resizing (isotropic scaling) the image to a fixed number ( $\approx 16384$ ) of pixels. For each pixel  $p$  we select the closest color (w.r.t. the Euclidean distance) in the codebook  $\mathcal{C}$ , and increment the corresponding bin of the output  $k_c$ -dimensional histogram.

**Inverse document frequency** is a successful technique in BOW. It downweights the contribution of the most common visual words and increases the importance of rare visual words by multiplying the frequency component with its corresponding *idf* weighting term. Similar to this technique, we update the histogram components by applying color *idf* weighting terms.

**Normalization: power-law and normalization.** Two techniques are considered. First, the “power-law” transformation

<sup>1</sup>More precisely, we used the ITU-R recommendation BT.709 using the D65 white point reference.

exp no.	palette	$k_c$	metric	Power-law	idf	mAP
#1	structured	64	L1			38.0
#2	structured	512	L1			45.3
#3	structured	512	FastEMD			43.1
#4	structured	512	L1	x		54.4
#5	learned	256	L1			54.5
#6	learned	256	L1	x		62.8
#7	learned	256	L1	x	x	<b>63.8</b>
#8	learned	256	$\chi^2$	x	x	63.5
#9	learned	256	SJD	x	x	63.3
#10	learned	256	FastEMD	x	x	59.6

Table 1: Color signature construction in CIE-Lab space. Impact of the dictionary: *structured* refers to a typical palette (Lab is split in either  $4 \times 4$  or  $8 \times 8$  components, leading to  $k_c = 64$  or 512), *learned* refers to our color learning procedure for natural images. Our BOC method is #8.

proposed in [10] for Fisher vectors is applied component-wise. For a vector  $x = (x_1, \dots, x_d)$ , it consists in updating each component as  $x_i := \sqrt{x_i}$ .

Similar to *idf*, the objective of the power-law update is to down-weight colors in images that have very high frequencies, but in contrast to *idf* it does not take into account the statistics of other images in the dataset. The empirical distribution of colors gives very limited importance to less occurring colors, for instance those associated with small objects. The power-law method regularizes the contribution of each color in the final descriptor.

Finally, a L1 vector normalization is performed to make the vectors more comparable. The vector is updated as  $x := x / \sum_{i=1..d} x_i$ . Although an Euclidean normalization is often adopted in the BOW framework (e.g., in [11, 1]), the L1 normalization provides consistently better results for our color signatures.

**Comparison metric:** Signatures are compared to find the nearest neighbors of the query image in a database of signatures. The choice of the comparison metric is therefore critical. We compare the simple Manhattan (L1) and Euclidean (L2) distances, as well as distances specifically used for histograms: the fast Earth-Mover-Distance of [9], the  $\chi^2$  distance, the Shannon-Jenson divergence (SJD). The histogram intersection (HI) which were used in [12] for color histograms are equivalent to the L1 distance if the vectors are L1-normalized.

## 2.2 Evaluation

To motivate our final method, we analyze the impact of the choices involved in the bag-of-colors generation process. The search quality is evaluated as the mAP on the INRIA Holiday dataset [1], computed by the evaluation package available online. In this section, for the sake of exposition, our analysis is conducted on global descriptors only. The results for local descriptors obtained with the same signature generation method are analyzed in Section 3.

**Dictionary, normalization and comparison metric** have an impact that is summarized in Table 1. First observe that a learned vocabulary gives much better results than the original scheme by Swain which regularly splits the L, a and b axes (leaving many bins empty, as the Lab colorspace is not a cube). Second, the power-law transform has an important impact: an improvement of at least +8% of mAP is achieved by this simple component-wise preprocessing, showing that it is important not to consider dominant colors only. Third, the *idf* weighting provides a slight improvement of 1 point of mAP, comparable to what is typically observed for the standard

$k_c$	16	32	64	128	256	512	1024
mAP	41.2	51.8	57.1	61.0	63.8	64.0	64.6
ms/signature	22	25	27	47	65	128	220
ms/search	31	59	112	223	432	851	1692

**Table 2: Impact of the palette size on accuracy (mAP on Holiday), signature extraction time and search time for querying a dataset of 1 million vectors (one processor core).**

SIFT-based BOW framework.

Overall, the best comparison metric is the L1 distance, which is slightly better than the  $\chi^2$  distance and JSD. The HI and FastEMD distances provide poor results on our features.

The RGB space has also been investigated. Results are consistently worse than for the CIE-Lab space in all our experiments: for reference, using this color space provides mAP=59.9% at best, compared to 63.8% in the CIE-Lab space.

The number  $k_c$  of colors in the learned palette is the key parameter of the method. Its impact on both quality and timings is given by Table 2. We measured the signature extraction time with our Matlab implementation (the image is assumed to be resized already to 16k pixels). The search timings correspond to the cost of finding the 100 nearest L1-neighbors in a dataset of 1 million vectors. They have been measured by querying 500  $k_c$ -dimensional vectors using exhaustive search.

Our bag-of-colors method is denoted by BOC in the following. It corresponds to the best choices for the global color signature: a palette learned with our procedure, the power-law, *idf* weighting and L1 normalization. The best comparison metric is L1, which is fortunate as the triangular inequality can therefore be exploited for indexing purpose.

A comparison with the standard BOW method [11] directly applied on raw color pixels (i.e., using CIE-Lab, Euclidean normalization, cosine similarity and *idf*, but without power-law) leads to mAP=38.4%. This score is comparable with that of Swain’s method (Experiments #1 to #4), which gives a mAP between 38.0% and 45.3%, for  $k_c = 64$  and 512, respectively.

The comparison with the state of the art is provided in Table 3. It is carried out on two benchmarks: the INRIA Holiday dataset and the UKB benchmark [7]. On UKB the accuracy is measured as the average number of images correctly ranked in top four positions (best score=4).

BOC significantly outperforms BOW: [1] reports mAP=54.9% for this BOW baseline on Holiday and [3] a score=2.99 on UKB. Besides, it does not achieve the best results of [3], which reports mAP=81.3% and score=3.42, respectively, with an improved matching scheme based on geometry and Hamming Embedding. However, our approach is significantly simpler, the descriptor extraction orders of magnitude faster, and our descriptors requires only 256 floats, while [3] requires about 30KB per images. In the next section, we will comment the other results of Table 3, which shows that BOC is also a powerful descriptor for local patches.

### 3. LOCAL COLOR SIGNATURE

The color descriptor proposed in the previous section has been shown to provide excellent performance on standard benchmarks. However, because at this point we used it only as a global descriptor, it is not able to cope with the most difficult cases such as strong cropping. Moreover, it does not capture texture information which had made the success of gradient based descriptors such as SIFT.

Method ↓	base descriptor	$D$	Holidays	UKB
<i>Results reported in the literature</i>				
BOW [3]	SIFT	20,000	46.9	2.88
BOW [3]	SIFT	200,000	57.2	2.95
HE [3]	SIFT	N/A	74.5	3.30
<i>Results with other methods (reimplemented)</i>				
color GIST [8]	RGB pixels	960	37.6	2.06
BOW	color SIFT	20,000	57.9	2.28
HE	color SIFT	N/A	76.5	2.58
<i>Proposed approach</i>				
BOC	Lab pixels	256	63.8	3.34
LBOC	SIFT+BOC	N/A	78.9	3.50

**Table 3: Comparison with the state of the art. quality: mAP on Holiday and score/4 on UKB (higher is better). All schemes are compared without the spatial verification re-ranking scheme. N/A means that the image is not represented by a single vector of fixed length  $D$  but by a set of descriptors’ signatures (typically requires 30kB/image in memory).**

In this section, our color signature is applied directly on patches and used in conjunction with SIFT descriptors [4] to provide a rich combination of both color and texture, which was shown successful in the recent color extension of SIFT descriptors [13]. It is integrated within our own implementation of the image matching method of [3], called Hamming Embedding (HE), which is a state-of-the-art extension of BOW.

#### 3.1 Binary signatures

In [3], the SIFT descriptors are assigned to visual words, but are additionally described by a binary signature of 64 bits. Each binary signature is computed from the SIFT descriptor, complementing the information given by the visual word. The Hamming distance between binary signatures is used to remove non-matching feature pairs and assign a voting score to the selected matches.

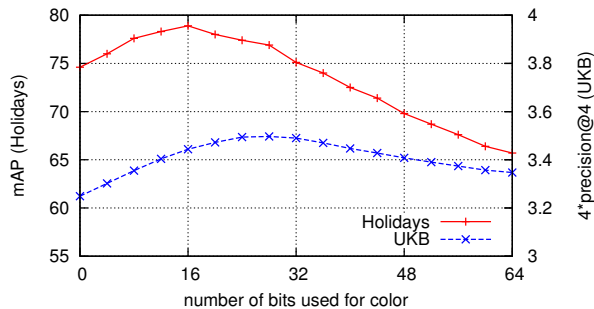
Since this distance removes mismatches, it offers an alternative to the dramatic increase in visual vocabulary sizes required by other approaches (up to 1M visual words in [7]). It is also an elegant way of taking into account the density variations in descriptor space. Indeed, the Hamming distance is implicitly normalized with respect to the local density in the BOW quantization cell. This property facilitates the early fusion of heterogeneous binary signatures, in our case the binary color signature described below.

#### 3.2 Color binary signatures

Our approach follows the guidelines of [3], but instead of using the 64 bits to describe the SIFT, some of the bits are dedicated to represent color information. A local BOC descriptor is extracted from the elliptical region on the image that is used to compute the SIFT descriptor. We use a palette of  $k_c=256$  colors. The vector is then binarized into a short binary signature of fixed size. The Hamming distance between two binary color signature should, at best, reflect the similarity between the original BOC descriptors.

The 256-dimensional BOC vector needs to be reduced to a small number of bits  $b_c$  ( $b_c = 0..64$ , depending on the balance between SIFT and color). This descriptor tends to be sparse because it is computed on small regions: on average 80% of its components are 0. We therefore adopt a dimensionality reduction inspired by [2], which was shown effective to reduce sparse BOW vectors. We first generate a sparse binary matrix  $B$  of size  $k_c \times k_c$  such that each row and each column of  $B$  sum up to 8, and keep only the first  $b_c$  rows, producing a sparse binary matrix  $B'$  of size  $b_c \times k_c$ .

The color binary signature is obtained from a given local BOC



**Figure 2: Trade-off between SIFT and color on Holiday and UKB.** Each detected region is represented by 64 bits in total, shared by the representation of the SIFT and the LBOC descriptor. The performance scale on the left is for the Holiday dataset, on the right it is for UKB.

vector  $x$  by first computing the projected vector  $x' = B' \times x$ . Second, each component of  $x'$  is compared to the median value for this component (learned on a training set), which produces a bit.

*Note:* From our experiments, this strategy is better than using a projection obtained by principal component analysis. This might be due to the fact our sparse projection is more consistent with the L1 distance, which is clearly better than L2 for BOC vectors. It is also better than binarizing a BOC vector generated by a palette comprising  $k_c = b_c$  colors.

### 3.3 Experiments

As in Section 2.2, our evaluation is performed on Holiday and UKB. To obtain a fair comparison, we have adjusted the threshold of the color SIFT detector so that, on average, it approximately produces the same number of descriptors per image, as shown below:

Benchmark	Holiday	UKB
Hessian-Affine + SIFT	2988	1048
color SIFT	3244	1066

The trade-off between color and texture is shown by Figure 2 for a fixed signature of 64 bits, where  $b_c$  bits are used for color and the remaining  $64 - b_c$  bits are the ones defined for SIFT by HE. The importance of color clearly depends on the dataset: SIFT alone performs better than the LBOC descriptor on Holiday, but this observation is reversed for UKB<sup>2</sup>. Overall, the optimum is clearly obtained when sharing the bits between these descriptors, with approximately 70 to 75% of the signature dedicated to SIFT and the rest used to represent color.

The comparison with the state of the art is shown by Table 3. Abusing the notation, the method that uses jointly SIFT and our binarized color signature is denoted by LBOC (local bag-of-colors). The color SIFT descriptor provides better results than SIFT on Holiday. However on UKB this descriptor suffers from not being invariant to rotation. The results of HE are significantly better than those obtained with BOW, as reported in the literature. The best results are clearly obtained with our LBOC approach, outperforming the reference method (SIFT+HE) and color SIFT by 4.4% and 2.4% points of mAP, respectively.

Large scale experiments have been performed on Holiday merged with 100K images from Flickr. With SIFT+LBOC, we ob-

<sup>2</sup>Note, however, that even for UKB with all 64 bits allocated to color in the signature, the visual word already provides a partial information about the SIFT descriptor. Therefore one should not conclude that color is more important than SIFT on this benchmark.

tain mAP=65.3% against 61.2% for HE with SIFT only. Querying the inverted file takes 0.56 s using a single processor core (2.5Ghz). This timing and the memory usage, about 30 kB/image, do not depend on the input descriptor (SIFT, LBOC or a combination).

## 4. CONCLUSION

We have proposed a color signature inspired by the bag-of-words framework, and used it either as a global or a local descriptor. Our experiments show that color information significantly improve the retrieval quality in large scale image search, which is often ignored in this context. Our global descriptor exhibits very competitive performance compared to more sophisticated methods. Used as a local descriptor, the results are better than those obtained by a state-of-the-art system based on SIFT or color SIFT descriptors.

## Acknowledgments:

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

## 5. REFERENCES

- [1] H. Jégou, M. Douze, and C. Schmid. Hamming embedding and weak geometric consistency for large scale image search. In *ECCV*, October 2008.
- [2] H. Jégou, M. Douze, and C. Schmid. Packing bag-of-features. In *ICCV*, September 2009.
- [3] H. Jégou, M. Douze, and C. Schmid. Improving bag-of-features for large scale image search. *IJCV*, 87(3):316–336, February 2010.
- [4] D. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 60(2):91–110, 2004.
- [5] J. Matas, O. Chum, U. Martin, and T. Pajdla. Robust wide baseline stereo from maximally stable extremal regions. In *BMVC*, pages 384–393, September 2002.
- [6] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, and L. V. Gool. A comparison of affine region detectors. *IJCV*, 65(1/2):43–72, 2005.
- [7] D. Nistér and H. Stewénius. Scalable recognition with a vocabulary tree. In *CVPR*, pages 2161–2168, June 2006.
- [8] A. Oliva and A. Torralba. Modeling the shape of the scene: a holistic representation of the spatial envelope. *IJCV*, 42(3):145–175, 2001.
- [9] O. Pele and M. Werman. Fast and robust earth mover’s distances. In *ICCV*, 2009.
- [10] F. Perronnin, J. Sánchez, and T. Mensink. Improving the fisher kernel for large-scale image classification. In *ECCV*, 2010.
- [11] J. Sivic and A. Zisserman. Video Google: A text retrieval approach to object matching in videos. In *ICCV*, pages 1470–1477, October 2003.
- [12] M. J. Swain and D. H. Ballard. Color indexing. *IJCV*, 7:11–32, 1991.
- [13] K. E. A. van de Sande, T. Gevers, and C. G. M. Snoek. Evaluating color descriptors for object and scene recognition. *PAMI*, 32(9):1582–1596, 2010.
- [14] J. van de Weijer, T. Gevers, and A. Bagdanov. Boosting color saliency in image feature detection. *PAMI*, 28(1):150–156, January 2006.