Bag-of-Bags of Words : Irregular graph pyramids vs spatial pyramid matching for image retrieval

Yi Ren, Aurélie Bugeau, Jenny Benois-Pineau

Bag-of-Bags of Words
Irregular graph pyramids vs spatial pyramid matching for image retrieval

Yi Ren[1]   Aurélie Bugeau[1]   Jenny Benois-Pineau[1]

University of Bordeaux, LaBRI, UMR 5800, F-33400 Talence, France[1]
{yi.ren, aurelie.bugeau, jenny.benois-pineau}@labri.fr

Résumé

Cet article présente une nouvelle approche appelée bag-of-bags of words (BBoW) pour la recherche d'images par le contenu (CBIR). Il s'agit d'une extension du modèle classique dit sac-de-mots (bag of words - BoW). Dans notre approche, une image est représentée par un graphe placé sur une grille régulière de pixels de l'image. Les poids sur les arêtes dépendent de caractéristiques locales de couleur. Le graphe est découpé en un nombre fixe de régions qui constituent une partition irrégulière de l'image par la méthode Normalized Cuts. Enfin, chaque partition est représentée par sa propre signature suivant le même schéma que le BoW. Une image est donc décrite par un ensemble de signatures qui sont ensuite combinées pour la recherche d'images similaires dans une base de données. Contrairement aux méthodes existantes telle que Spatial Pyramid Matching (SPM), le modèle BBoW proposé ne repose pas sur l'hypothèse que des parties similaires d'une scène apparaissent toujours au même endroit dans des images d'une même catégorie. L'extension de cette méthode à une approche multi-échelle, appelée Irregular Pyramid Matching (IPM), est également décrite. Les résultats démontrent la qualité de notre approche lorsque les partitions obtenues sont stables au sein d'une même catégorie d'images. Une analyse statistique est menée pour définir concrètement la notion de partition stable.

Mots Clef

Recherche d'images par le contenu, coupe de graphes, Normalized Cuts, Sac de mots.

Abstract

This paper presents a novel approach, named bag-of-bags of words (BBoW), to address the problem of Content-Based Image Retrieval (CBIR) from image databases. The proposed bag-of-bags of words model extends the classical bag-of-words (BoW) model. An image is represented as a connected graph of local features on a regular grid. Then irregular partitions (subgraphs) of images are further built via Normalized Cuts. Each subgraph in the partition is then represented by its own signature. Compared to existing methods for image retrieval, such as Spatial Pyramid Matching (SPM), the BBoW model does not assume that similar parts of a scene always appear at the same location in images of the same category. The extension of the proposed model to pyramid gives rise to a method we name irregular pyramid matching (IPM). The experiments demonstrate the strength of our method for image retrieval when the partitions are stable across an image category. The statistical analysis of subgraphs is discussed in the paper.

Keywords

Content-based image retrieval, graph partitioning, Normalized Cuts, Bag of words.

1   Introduction

Recent methods in Content-Based Image Retrieval (CBIR) mostly rely on the bag-of-visual-words (BoW) model [1]. The idea, borrowed from document processing, is to build a visual codebook from all the feature points in a training image dataset. Each image is then represented by a signature, which is a histogram of quantized visual features-words from the codebook. Image features are thus considered as independent and orderless. The traditional BoW model does not embed spatial layout of local features in image signature. However, this information has shown to be very useful in tasks like image retrieval, image classification, and video indexing. Ren et al. [2] put forward a concept of grouping pixels into "superpixels". Leibe et al. proposed to adopt codebooks to vote for object position [3]. Lazebnik et al. [4] partitioned an image into increasingly fine grids and computed histograms for each grid cell. The resulting spatial pyramid matching method (SPM) clearly improves the BoW representation. Nevertheless, this method relies on the assumption that a similar part of a scene generally appears at the same position across different images, which does not always hold.

Graphs are versatile tools to conveniently represent patterns in computer vision applications and they have been vastly investigated. By representing images with graphs, measuring the similarities between images becomes equivalent to finding similar patterns inside series of attributed

(sub)graphs representing them. Duchenne et al. [5] introduced an approximate algorithm based on graph-matching kernel for category-level image classification. Gibert et al. [6] proposed to apply graph embedding in vector spaces by node attribute statistics for classification. Bunke et al. [7] provided an overview of the structural and statistical pattern recognition, and elaborated some of these attempts, such as graph clustering, graph kernels and embedding etc., towards the unification of these two approaches.

This paper presents a new approach for Content-Based Image Retrieval (CBIR) that extends the bag-of-words (BoW) model. We aim at embedding color homogeneity and limited spatial information through irregular partitioning of an image into a set of predefined number of graphs. Each partition results from applying graph partitioning methods to an initial full connected graph, in which nodes are positioned on a dense regular grid of pixels. We adopt the graph segmentation method called Normalized Cuts [8]. The BoW approach is then applied to each of resulting subgraph independently. An image is finally represented by a set of graph signatures (BoWs), leading to our new representation called *bag-of-bags of words* (BBoW). As in the spatial pyramid matching approach [4], we also consider a pyramidal representation of images with a different number of (sub)graphs at each level of the pyramid. The comparison of images in a CBIR paradigm is achieved via comparison of the irregular pyramidal partitions. We call this pyramidal approach *Irregular Pyramid Matching* (IPM). Hence in this paper, we try to address a challenging question : will an irregular segmentation-like partition of images outperform a regular partition (SPM) ? Intuitively, it is invariant to the rotation and reasonable shift transformations of image plane. Nevertheless, what can be its resistance to noise and occlusions ? How will it compete with SPM when embedded into pyramidal paradigm ?

The remainder of the paper is organized as follows. In section 2 we briefly introduce the notations and prerequisites. The proposed bag-of-bags of words model is discussed in section 3 and the irregular pyramid matching in section 4. Section 5 presents the experimental results and we conclude in section 6.

2 Terminology

The input database $\Omega = (I, V)$ is composed of N RGB images $I = \{I_1, \ldots, I_N\}$ and of a visual codebook $V = \{V_1, \ldots, V_B\}$ of size B. The codebook is a collection of vector quantized features created by k-means clustering over local SIFT [9] features. The latter are extracted from all the grid dense sampling points of the images in a randomly selected training sample. Let us denote by $G_j = (\mathcal{V}_j, \mathcal{E}_j, W_j)$ an *undirected weighted* graph constructed on the image I_j. The set $\mathcal{V}$ of vertices contains a regularly sampled subset of pixels $\mathcal{P}$ of the image and at the limit can contain all of them. The graph edges $\mathcal{E}$ connect these vertices with a 8-connected neighbourhood system. The af-

finity matrix W of size $|\mathcal{V}| \times |\mathcal{V}|$ is defined as :

$$W_{pq} = \begin{cases} w_{pq} & \text{if } p, q \in \mathcal{E} \\ 0 & \text{if } p, q \notin \mathcal{E}, \end{cases}$$

where w_{pq} represents the edge-based similarity between two vertices p and q. For each image I_j in the database, we aim to partition the graph G_j into fixed K disjoint unconnected subgraphs $\{g_{j,1}, \ldots, g_{j,K}\}$, such that $\forall k \neq l, g_{j,k} \cap g_{j,l} = \emptyset$ and $G_j = \{\bigcup_{k=1}^{K} g_{j,k}\} \cup E_j$, where $E_j \subset \mathcal{E}_j$ are removed edges to divide G_j by graph cuttings. We denote this K-*way partitioning* by $\Gamma_{j,K} = \{g_{j,1}, \ldots, g_{j,K}\}$. Graph partitioning is a labeling problem. Given a set of vertices $\mathcal{V}$ and a set of labels $L = \{1, 2, \ldots, K\}$, for all node $p \in \mathcal{V}$, we are looking for the optimal label $l_p \in L$, such that the joint labeling $\mathcal{L} = \{l_1, \ldots, l_{|\mathcal{V}|}\} \in L^{|\mathcal{V}|}$ satisfies a specified objective function.

3 Bag-of-Bags of Words Model

Our BBoW model has been inspired by the *bag-of-words model* [1] for image description and the idea that an irregular partition into a fixed number of graphs is a step in-between an arbitrary regular partition as in [4], and a segmentation of an image plane which can be redundant. The imperfections of this intermediate partition should be compensated by a statistical nature of the original BoW model for each graph in the partition. Furthermore, we embed BBoW into multi-resolution schemes as *spatial pyramid matching* [4] for using image partitions in coarse to fine manner. For each image, the construction of a bag-of-bags of words (figure 1) can be decomposed into four steps :

1) Select a reduced number of pixels $\mathcal{V}$.

2) Build an initial graph G.

3) Partition the graph G into K subgraphs.

4) Compute a signature for each subgraph.

The signature of a subgraph is a histogram of codeword occurrences, i.e. obtained by assigning each feature node of this subgraph to the closest visual word in the codebook. Hence, an image being composed of K subgraphs is characterized by a set of K histograms. In the following, we detail each step of the method.
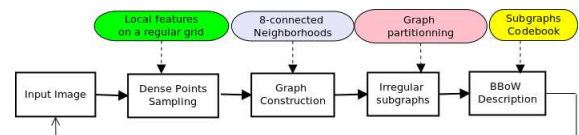

FIGURE 1 – Bag-of-bags of words pipeline

3.1 Graph nodes selection

We adopt a dense sampling strategy. All the areas contribute equally to the final image representation. We choose 8-pixels spacing and 16-pixels patch size for the SIFT [9] features, called *Bag-of-Features Grid SIFT (BF-GSIFT)* [10].

3.2 Construct initial weighted graph

The set of nodes $\mathcal{V}$ of our initial graph G contains all the points resulting from the dense sampling. The edges $\mathcal{E}$ are obtained by linking these points with a 8-connected neighbourhood system. The weights on the edges are :

$$w_{pq} = \exp\left(-\lambda \left(\bar{C}_p - \bar{C}_q\right) \Sigma^{-1} \left(\bar{C}_p - \bar{C}_q\right)^T\right), \quad (1)$$

where $\bar{C}_p$ accounts for the mean color vector over a $n \times n$ patch centred on point $p = (x, y)$. λ is a parameter to differentiate color similarity. We use the YUV color space because it has independent color channels and allows to better deal with changes in lightning conditions. We denote $Y_p = Y(x, y)$ (respectively $U_p = U(x, y)$ and $V_p = V(x, y)$) as the color channel value at point p. The mean color vector becomes $\bar{C}_p = (\bar{Y}_p, \bar{U}_p, \bar{V}_p)$ where :

$$\bar{Y}_p = \frac{1}{n^2} \sum_{u=-n}^{n} \sum_{v=-n}^{n} Y(x+u, y+v).$$

The covariance matrix $\Sigma = diag[\sigma_Y^2, \sigma_U^2, \sigma_V^2]$ is diagonal because of channels' independence. As in [11], the channel covariance $\sigma_Y^2, \sigma_U^2, \sigma_V^2$ are computed over all the edges $(p, q) \in \mathcal{E}$ of the graph G.

3.3 Graph Partitioning with NCuts

We now present how the initial connected full graph G is partitioned into K subgraphs. Let us denote $links(\mathcal{A}, \mathcal{B})$ to be the sum of the edge weights between nodes in two *disjoint* graph nodes' sets $\mathcal{A}$ and $\mathcal{B}$:

$$links(\mathcal{A}, \mathcal{B}) = \sum_{i \in \mathcal{A}, j \in \mathcal{B}} w_{ij}.$$

Shi et al. introduced the *Normalized Cuts (NCuts)* [8], a graph-theoretic criterion for measuring the goodness of partitions. The algorithm optimizes two criteria, measuring the total dissimilarity between the different subgraphs

$$WNcuts(G_j) = \min_{g_{j,1}, \dots, g_{j,K}} \sum_{k=1}^{K} \frac{links(g_{j,k}, \mathcal{V}_j \setminus g_{j,k})}{w(g_{j,k})}, \quad (2)$$

as well as the similarity within each subgraph

$$WNassoc(G_j) = \max_{g_{j,1}, \dots, g_{j,K}} \sum_{k=1}^{K} \frac{links(g_{j,k}, g_{j,k})}{w(g_{j,k})}. \quad (3)$$

where $w(g_{j,k})$ is the sum of edge weights in the subgraph $g_{j,k}$. K is the number of subgraphs.

The principle is to find a minimal cut which is a combination of edges having minimal sum of edge values (i.e. find the *least alike* pairs of nodes). Removing these edges divides the graph G into unconnected subgraphs, such that the similarity between nodes within a subgraph is greater than the similarity between nodes in separated subgraphs.

The advantage of the normalized cut method is that it considers two aspects of graph segmentation : minimal cut (i.e. better separation) and preferring segments of large size. In practice, we used the code from [8, 12] to directly apply *NCuts* to partition a weighted graph G.

3.4 Signatures of subgraphs and BBoW

As in the standard BoW approach, we first build a codebook from all dense features in a training image dataset. The codebook is thus independent from any partitioning scheme. Let us denote $H_{j,k}$, a signature of the subgraph $g_{j,k}$, $k = 1 \dots K$ in image I_j. $H_{j,k}$ is a Bag of Words (BoW) histogram, obtained by assigning the SIFT features of all graph nodes within the subgraph to the nearest codeword (hard assignment). Hence, the signature of an image I_j will be a histogram vector $H_j = \{H_{j,1}, \dots, H_{j,K}\}$ of length K, normalized by the number of nodes in the initial image graph G_j. With such a normalization, the larger subgraphs are privileged. We call H_j *"Bag-of-Bags of Words"* *(BBoW)*.

4 Irregular Pyramid Matching

Our model directly follows the strategy from Spatial Pyramid Matching [4]. We consider several "resolutions" ($r = 0 \dots R$) of a partition pyramid. At each resolution r, the image graph G_j is split into $K_r = 2^{2r}$ subgraphs. Hence, the BBoW representation of image I_j at resolution r is the vector $H_j^r = \{H_{j,1}^r, \dots, H_{j,K_r}^r\}$ of length K_r. An example of subgraphs and BBoWs is given in figure 2.
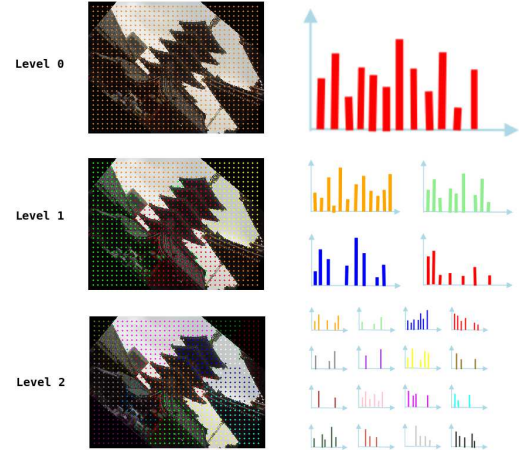

FIGURE 2 – A schematic illustration of the BBoW representation at each level of the pyramid. At level 0, the decomposition has a single graph, and the representation is equivalent to the classical BoW. At level 1, the image is subdivided into four subgraphs, leading four features histograms, and so on. Each subgraph is represented by its own color.

In order to compare two images I_i and I_j, we need to compare the histogram signatures of their subgraphs, and build a similarity measure upon them. For this purpose, we can use the histogram intersection function [13] :

$$\mathcal{I}\left(H_{i,k}^r, H_{j,k}^r\right) = \sum_{b=1}^{B} \min\left(H_{i,k}^r(b), H_{j,k}^r(b)\right), \quad (4)$$

where B is the codebook size.

In our method we cannot directly apply this function to match histograms of the subgraphs in the partition. At the end of the partitions steps, for two images, the two histograms $H^r_{i,k}$ and $H^r_{j,k}$ may not correspond one to the other. Indeed, after applying a graph partitioning method, the subgraph attributed to label k can be at any position in the image. The spatial arrangement is lost if an image in a database undergoes rotation, for instance. To overcome this issue, we first need to reorganize our histograms in BBoW to find an optimal matching of one partition to the other. We thus reorganize the labels between the two sets of histograms $\{H^r_{i,k}\}_{k=1,...,K_r}$ and $\{H^r_{j,k}\}_{k=1,...,K_r}$. We call this step *bipartite subgraphs matching*.

We are facing an assignment problem which requires the use of combinatorial algorithm. In our method, we rely on the *Hungarian algorithm* to minimize the discrete transport cost between the two sets of histograms. The cost matrix $D^r_{i,j}$ between the pair of images I_i and I_j at level r reads :

$$D^r_{i,j} = \begin{bmatrix} d^{11}_{i,j} & \cdots & d^{1K_r}_{i,j} \\ & \vdots & \\ d^{K_r 1}_{i,j} & \cdots & d^{K_r K_r}_{i,j} \end{bmatrix}. \tag{5}$$

where $d^{k,l}_{i,j} = \sum_{b=1}^{B} |H^r_{i,k}(b) - H^r_{j,l}(b)|$ is the L_1 distance between $H^r_{i,k}$ and $H^r_{j,l}$, which is equivalent to the *histogram intersection* kernel. The *Hungarian algorithm* finds the minimum cost assignment by associating each label k of subgraph $g_{i,k}$ in image I_i to one label $k' = f_i(k,r)$ of subgraph $g_{j,k'}$ in image I_j, thus the pairs of histograms to compare are identified.

Now that bipartite subgraphs matching has been fulfilled, we can directly apply the *level weighted intersection* [4] :

$$\begin{aligned} \kappa(I_i, I_j) &= \mathcal{I}\big(H^R_{i,k}, H^R_{j,k'}\big) + \sum_{r=0}^{R-1} \frac{1}{2^{R-r}}\big(\mathcal{I}^r - \mathcal{I}^{r+1}\big) \\ &= \frac{1}{2^R}\mathcal{I}\big(H^0_{i,1}, H^0_{j,1}\big) + \sum_{r=1}^{R} \frac{1}{2^{R-r+1}}\mathcal{I}\big(H^r_{i,k}, H^r_{j,k'}\big). \end{aligned}$$

The weight $\frac{1}{2^{R-r}}$ allows for penalizing low resolutions of a partition, reflecting the fact that higher levels localize the features inside smaller graphs more precisely.

5 Experimental evaluation of IPM

In this section, we present the experimental evaluation on Caltech101 benchmark [14]. The Caltech101 dataset includes 101 objects categories such as faces, watches, ants, pianos, etc... and a background category for a total of 102 categories. Following the same setting as in [4], we use the dense grid of points generated from their codes to build initial full connected graphs. The dictionary of size 400 is learnt over 30 sample images per class using k-means and query images are chosen from the rest of the dataset for retrieval.

5.1 Selection and impact of parameters

We first estimate the impact of the different parameters on image retrieval performance. Two parameters have been

evaluated : the patch size n defined in section 3.2 and the parameter λ for the edges weight (equation 1). The mean Average Precisions (mAP) on Caltech101 are presented in tables 1 and 2. For efficiency, these experiments were conducted on smaller databases containing only 4 test images per category. The results highlight that these two parameters do not remarkably influence the quality of retrieval. Therefore, we decide to select $n = 5$ and $\lambda = 5$ for the following experiments.

	mAP for the IMP approach at 3 levels pyramid			
	$\lambda = 0$	$\lambda = 5$	$\lambda = 20$	$\lambda = 100$
IPM	0.386	**0.390**	0.386	0.381

TABLE 1 – Influence of parameter λ (equation 1) on image retrieval. For each value, the mean average precision is given. For this experiment, mAP(SPM)=0.409, the patch size is set to $n = 5$.

	mAP for the IMP approach at 3 levels pyramid			
	n = 3	n = 5	n = 7	n = 9
IPM	0.381	**0.389**	**0.389**	0.388

TABLE 2 – Influence of parameter n (section 3.2) on image retrieval. For each value, the mean average precision is given. For this experiment, mAP(SPM)=0.409, $\lambda = 5$.

5.2 Image retrieval evaluation

The performance of our proposed IPM algorithm is measured by the *mean Average Precision (mAP)* and its standard deviation for each category and on the whole dataset, as described in TREC-style evaluation [1]. Table 3 presents this global mAP for Caltech101 at each level of the pyramid. We compare ours with SPM [4] and with the bag-of-words approach which simply corresponds to the results at level 0. SPM outperforms BBoW in general, but our method achieves better results than BoW (level 0) overall. By looking closer to the global mAPs, we can nevertheless observe that the pyramidal approach is not needed for SPM on this dataset. Indeed, the results at single level 2 are better than the ones with the pyramidal approach. These results have been obtained with the official codes of SPM. On the contrary, the pyramidal approach is a good strategy in our case, as it improves the retrieval performance with increasing levels.

	Level 0	Level 1	Level 2	Pyramid
IPM	0.117 ±0.18	**0.107** ±0.16	**0.125** ±0.19	**0.129** ±0.20
SPM	0.117 ±0.18	0.144 ±0.21	**0.162** ±0.23	**0.157** ±0.22

TABLE 3 – Retrieval performance (mAP and its standard deviation) on the whole Caltech-101 dataset.

We then propose to investigate more details for fifteen typical categories of Caltech-101 dataset (figure 3), for which the mAP of IMP is contrast to that of SPM. There are seven categories in which SPM has notable better retrieval performance. These also explain why SPM is globally better than IPM on whole database. Nevertheless BBoW achieves

---

1. http ://trec.nist.gov/

comparative results with SPM for most of categories, and even outperforms it in eight categories.
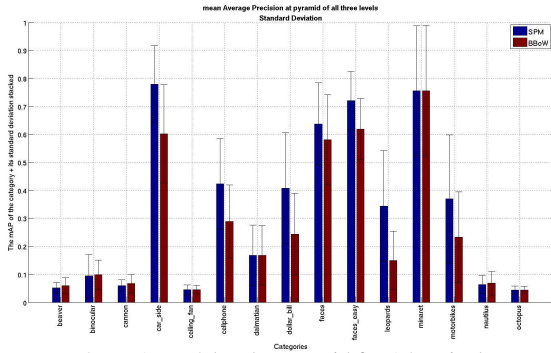

FIGURE 3 – mAP at 3 levels pyramid for 15 typical categories in Caltech101. SPM outperforms IPM considerably in 7 categories.

5.3 Statistical (sub)graph analysis

In order to give a deeper understanding on why our method suits for certain categories and not for the others, let us go through three case studies : a "good" category (*minaret*), and two "bad" ones (*cellphone*, *dollar_bill*).

Figures 4 and 5 present our partitioning results at level 1, i.e. 4 subgraphs. Figure 4 illustrates good queries from *minaret* class in which BBoW outperforms SPM. As can be seen, our method is able to represent the *minaret* (object) with only one or two subgraph(s). The other subgraphs in the background are also stable in the sense that almost the same subgraphs can be found in each high-ranking retrieved image within *minaret* category. Figure 5 shows four bad queries from *cellphone* and *dollar_bill* categories in which SPM achieves better performance than our method. The obtained partitions almost do not exhibit any consistency across the images from the same category. The resulting histograms are therefore each time very different and we can obviously not expect any good retrieval.
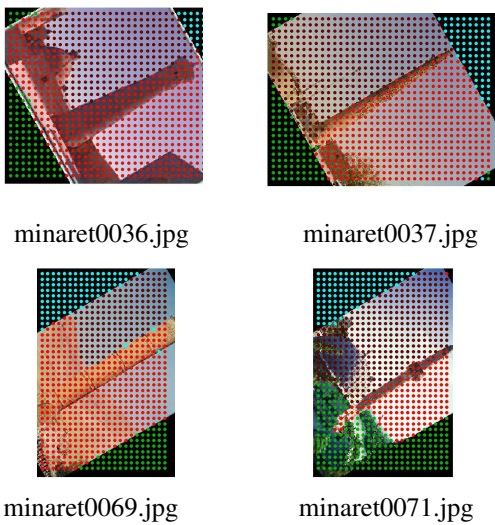

minaret0036.jpg   minaret0037.jpg

minaret0069.jpg   minaret0071.jpg

FIGURE 4 – The 4 irregular subgraphs (in red, blue, green, brown colors) from "good" query images at level 1.

cellphone0034.jpg   cellphone0045.jpg

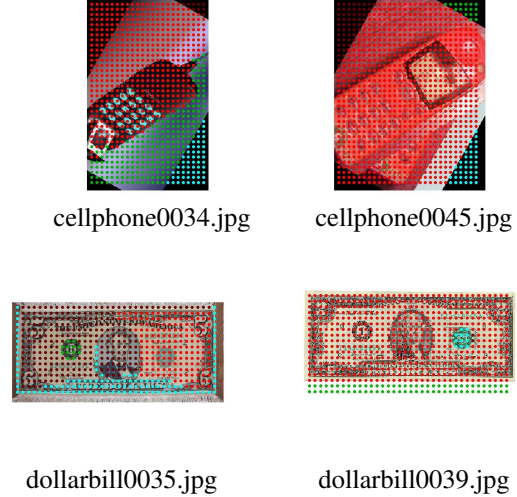dollarbill0035.jpg   dollarbill0039.jpg

FIGURE 5 – The 4 irregular subgraphs (in red, blue, green, brown colors) from "bad" query images at level 1.

We now detail the retrieval performance and the statistics of the (sub)graphs for these eight query images. One can see in figure 6 that in "good" categories, our method outperforms SPM [4]. In this case, the partitions from NCuts are stable across the images of the same category. The mean number of nodes in "correctly" matched subgraphs together with the standard deviation are presented in figure 7. The standard deviation is high for "bad" categories, whereas it is more uniform for "good" category.
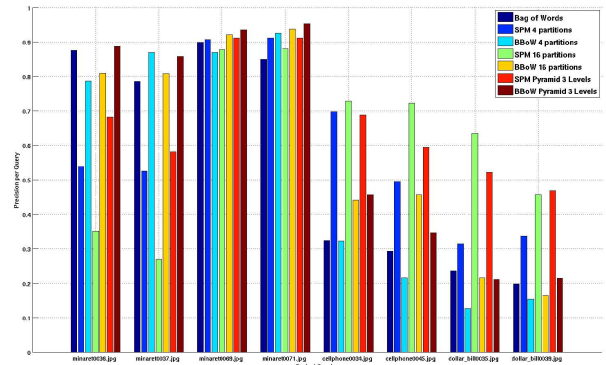

FIGURE 6 – The precision of 8 typical query images : 1st-4th query images are from a "good" category - *minaret*, 5th-8th query images are from "bad" categories : *cellphone* and *dollar_bill*.

In order to qualify performance of our approach, we also introduce a new measure. We call it *Partition Precision (PP)*. For particular query image I_j, we denote N_{rel} as the number of relevant images rel for this query in the whole database Ω. If I_j and I_i are from the same category, then $I_i \in rel$. We say $g_{j,k}$, $g_{i,k'}$ are matched, if they are bipartite matching subgraphs after rearrangement of histograms via *Hungarian algorithm*, moreover they fall into same regions (background, sky, object etc.) of the image I_j and I_i
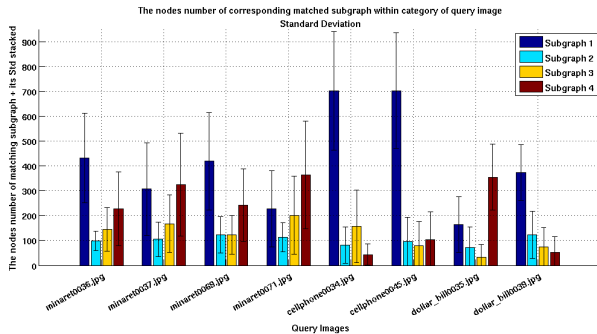
FIGURE 7 – The mean and standard deviation of nodes' numbers of corresponding subgraphs for intra-category, for 8 typical queries in Caltech-101 dataset - at single level 1, *i.e.* 4 subgraphs.

respectively. Then we can define an $N_{rel} \times K$ matrix M :

$$M_{d,k} = \begin{cases} 1 & g_{j,k}, g_{i,k'} \text{ are matched,} \\ 0 & \text{Otherwise.} \end{cases}$$

to describe the stability of partitioning G_j into $\Gamma_{j,K} = \{g_{j,1}, \ldots, g_{j,K}\}$. The *Partition Precision (PP)* for query image I_j is thus given by a row vector of length K :

$$\{P(g_{j,k})\}_{k=1,\ldots,K} = \frac{1}{|N_{rel}|} \sum_{d=1\ldots|N_{rel}|} M_{d,k}$$

where $M_{d,k}$ is an element of matrix M. The above vector thus characterizes the goodness of match of each subgraph in a query image to the subgraphs of images of the same category in the database Ω. Table 4 and 5 show figures for the partition precision of 4 typical queries of a "good" category and of two "bad" categories. One can see that in the "good" categories, the worst matching percentage (in bold) is better than that in "bad" queries. Hence, this correlates with better stability of nodes' numbers, as in figure 7.

Query image I_j	$g_{j,1}$	$g_{j,2}$	$g_{j,3}$	$g_{j,4}$
minaret0036	0.773	0.746	0.626	**0.6**
minaret0037	0.786	0.72	0.626	**0.6**
minaret0069	0.773	**0.44**	0.6	0.613
minaret0071	0.8	0.506	0.6	**0.413**

TABLE 4 – The $\{P(g_{j,k})\}_{k=1,\ldots,4}$ of corresponding subgraphs for 4 typical "good" queries.

Query image I_j	$g_{j,1}$	$g_{j,2}$	$g_{j,3}$	$g_{j,4}$
cellphone0034	0.724	0.638	0.569	**0.241**
cellphone0045	0.724	**0.362**	0.569	0.552
dollar_bill0035	0.431	0.686	0.392	**0.176**
dollar_bill0039	0.275	0.824	**0.06**	0.418

TABLE 5 – The $\{P(g_{j,k})\}_{k=1,\ldots,4}$ of corresponding subgraphs for 4 typical "bad" queries.

6 Conclusion and perspectives

In this paper, the bag-of-bags of words model over irregular image partition via *Normalized Cuts* and its pyramidal extension, the *irregular pyramid matching (IPM)* have

been introduced. Normalized Cuts method is used to separate the graph of an image, which is composed of full connected dense grid-points, into several subgraphs. Each subgraph is then described by a bag-of-words model. Our approach incorporates color and limited spatial information into image representation for image retrieval. The obtained results are encouraging, especially when the partitions obtained are stable across the images from the same category. A criterion for measuring the stability of subgraphs has been defined in the experimental section. The stability is nevertheless not always insured with the aforementioned graph weights functions. In the future, we consider embedding more image descriptors in graph weights and bipartite subgraphs matching process in order to reach more stability.

Acknowledgment

Références

[1] J. Sivic and A. Zisserman, "Video google : a text retrieval approach to object matching in videos," in *ICCV*, pp. 1470 –1477, 2003.

[2] X. Ren and J. Malik, "Learning a classification model for segmentation," in *ICCV*, 2003.

[3] B. Leibe, A. Leonardis, and B. Schiele, "Combined object categorization and segmentation with an implicit shape model," in *In ECCV workshop on statistical learning in computer vision*, 2004.

[4] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features : Spatial pyramid matching for recognizing natural scene categories," in *CVPR*, 2006.

[5] O. Duchenne, A. Joulin, and J. Ponce, "A graph-matching kernel for object categorization," in *ICCV*, 2011.

[6] J. Gibert, E. Valveny, and H. Bunke, "Graph embedding in vector spaces by node attribute statistics," *Pattern Recognition*, vol. 45, no. 9, pp. 3072 – 3083, 2012.

[7] H. Bunke and K. Riesen, "Towards the unification of structural and statistical pattern recognition," *Pattern Recognition Letters*, vol. 33, no. 7, pp. 811 – 825, 2012.

[8] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell. (PAMI)*, vol. 22, no. 8, pp. 888–905, 2000.

[9] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. journ. of comp. vis. (IJCV)*, vol. 60, no. 2, pp. 91–110, 2004.

[10] T. Tuytelaars, "Dense interest points," in *CVPR*, pp. 2281–2288, 2010.

[11] C. Rother, V. Kolmogorov, and A. Blake, "Grabcut : Interactive foreground extraction using iterated graph cuts," *ACM Trans. Graph.*, vol. 23, no. 3, pp. 309–314, 2004.

[12] T. Cour, S. X. Yu, and J. Shi, "Normalized cut segmentation code," 2004.

[13] F.-D. Jou, K.-C. Fan, and Y.-L. Chang, "Efficient matching of large-size histograms," *Pattern Recognition Letters*, vol. 25, no. 3, pp. 277–286, 2004.

[14] L. Fei-Fei, R. Fergus, and P. Perona, "Learning generative visual models from few training examples : an incremental bayesian approach tested on 101 object categories," *Workshop on Generative-Model Bas. Vis.*, 2004.