Structured Representations in a Content Based Image Retrieval Context

Romain Raveaux, Jean-Christophe Burie, Jean-Marc Ogier

Structured Representations in a Content Based Image Retrieval Context

Romain Raveaux[a], Jean-Christophe Burie[b] and Jean-Marc Ogier[b]

[a] *Computer Science, LI Laboratory at the University of Tours, av Jean Portalis, Tours, France*

[b] *Computer Science, L3I Laboratory at the University of La Rochelle, av M. Crepeau, La Rochelle, France*

Abstract

Here, we propose an automatic system to annotate and retrieve images. We assume that regions in an image can be described using a vocabulary of blobs. Blobs are generated from image features using clustering. Features are locally extracted on regions to capture Color, Texture and Shape information. Regions are processed by an efficient segmentation algorithm. Images are structured into a region adjacency graph to consider spatial relationships between regions. This representation is used to perform a similarity search into an image set. Hence, the user can express his need by giving a query image, and thereafter receiving as a result all similar images. Our graph based approach is benchmarked to conventional Bag of Words methods. Results tend to reveal a good behavior in classification of our graph based solution on two publicly available databases. Experiments illustrate that a structural approach requires a smaller vocabulary size to reach its best performance.

1 Introduction

With the development of the Internet, and the availability of image capturing devices such as digital cameras, image scanners, the size of digital image collection is increasing rapidly. Efficient image searching, browsing and retrieval tools are required by users from various domains, including remote sensing, fashion, crime prevention, publishing, medicine, architecture, etc. For this purpose, many general purpose image retrieval systems have been developed. There are two frameworks: text-based and content-based. The text-based approach can be tracked back to 1970s. In such systems, the images are manually annotated by text descriptors, which are then used by a database management system (DBMS) to perform image retrieval. There are two disadvantages with this approach. The first is that a considerable level of human labor is required for manual annotation. The second is the annotation inaccuracy due to the subjectivity of human perception [1] [2]. To overcome the above disadvantages in text-based retrieval system, content-based image retrieval(CBIR) was introduced in the early 1980s. In CBIR, images are indexed by their visual content, such as color, texture, shapes. A pioneering work was published by Chang in 1984, in which the author presented a picture indexing and abstraction approach for pictorial database retrieval [3]. The pictorial database consists of picture objects and picture relations. To construct picture indexes, abstraction operations are formulated to perform picture object clustering and classification. In the past decades, a few commercial products and experimental prototype systems have been developed, such as QBIC [4], Photobook

[5], Virage [6], VisualSEEK [7], Netra [8], SIMPLIcity [9]. Comprehensive surveys in CBIR can be found in Refs. [10] [11].
Image retrieval has been an active research area over the last decades. There are many researches and review articles that mention the importance, requirements and applications of CBIRS [12], [13], [14] and [15]. Most researchers provide an extensive description of image archives, various indexing methods and common searching tasks, using different techniques and technologies. Currently CBIR techniques can be classified into two categories: Global approach by using global visual features to describe images and Local approach by considering images as the combination of multiple objects, keypoints or regions.

1.1 Global methods

This technique deals with image globally and tries to characterize it by using visual/statistical features calculated from the entire image. Visual features are classified into primitive features such as color or shape, logical features such as identity of objects shown and abstract features such as significance of scenes depicted [15].

Color In domain of photograph retrieval, color has been the most effective feature and almost all systems employ colors. Although most of the images are in the red, green, blue (RGB) color space. Color histograms are used to compare images in many applications. Their advantages are efficiency, and insensitivity to small changes in camera viewpoint. However, color histograms lack spatial information, so images with very different appearances can have similar histograms.

Texture Some of the most common measures for capturing the texture of images are wavelets and Gabor filters. These texture measures try to capture the characteristics of the image or image parts with respect to changes in certain directions and the scale of the changes. This is most useful for regions or images with homogeneous texture. Again, invariances with respect to rotations of the image, shifts or scale changes can be included into the feature space. Other popular texture descriptors contain features derived from co-occurrence matrices, features based on the factors of the Fourier transform and the so-called Wold features [16].

Shape Features There are many shape representation and description techniques in the literature. Marr and Nishihara [17] and Braddy [18] have thoroughly discussed representation and sets of criteria for the evaluation of shape. Shape description or representation is an important issue both in object recognition and classification. It has been used in CBIR in conjunction with color and other features for indexing and retrieval. Many techniques, including chain code, polygonal approximations, curvature, Fourier descriptors and moment descriptors have been proposed and used in various applications [19]. The query images are represented by Fourier descriptors which serve powerful boundary-shape representation tools because of invariance property in affine transformation. Among the well-known shape descriptors, the Zernike moments have been successfully used in many shape contests [20].

Literature on image content indexing is very large, see for example [21] for a survey. A common approach to model image data is to extract a vector of features from each image in the database (e.g. a color histogram) and then use the Euclidean distance between those feature vectors as similarity measure for images. But the effectiveness

of this approach is highly dependent on the quality of the feature transformation. Often it is necessary to extract many features from the database objects in order to describe them sufficiently, which results in very high-dimensional feature vectors. Those extremely high-dimensional feature vectors cause many problems commonly described by the term 'curse of dimensionality'. Especially for image data, the additional problem arises how to include the structural information contained in an image into the feature vector. As the structure of an image cannot be modeled by a low-dimensional feature vector, the dimensionality problem gets even worse.

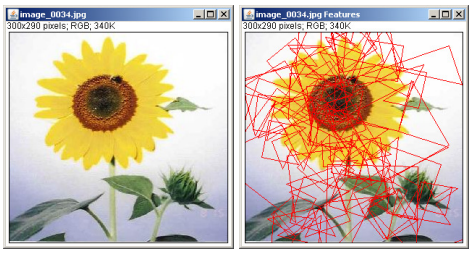
To address this topic, several solutions were proposed, involving spatial relationships between entities in images which can be symbolic objects(e.g. objects highlighted after a phase of automatic detection or recognition, localization and labeling) as well as low-level features(e.g. salient points).

1.2 Local approaches

In this part, details about local approaches are provided. Two main stages are mentioned : Blob extraction and Blob arrangement.

1.2.1 Blob extraction

The detection and description of local image features can help in object recognition. The Scale Invariant Feature Transform (SIFT) features are local and based on the appearance of the object at particular interest points, and are invariant to image scale and rotation. They are also robust to changes in illumination, noise, and minor changes in viewpoint. In addition to these properties, they are highly distinctive, relatively easy to extract, allow for correct object identification with low probability of mismatch and are easy to match against a (large) database of local features. Object description by set of SIFT features is also robust to partial occlusion; as few as 3 SIFT features from an object are enough to compute its location and pose. Here, we use SIFT (scale invariant feature transform) [22] to lead a comparative study. The full SIFT feature set is a 128 dimensional vector that captures the spatial structure and the local orientation distribution of a region surrounding a keypoint. Recently studies have shown that SIFT is one of the best descriptors for keypoints [23]. On the other hand, SIFT provides subsamples of the image leading to a high number of regions of interest. This phenomenon is illustrated in figure 1. This subsampling effect is not suitable for a meaningful topological arrangement. A given image is convolved with Gaussian filters at different scales, and then the difference of successive Gaussian-blurred images are taken. Keypoints are then taken as maxima/minima of the Difference of Gaussians. This keypoints detection is quite light to execute, however SIFT produces a high number areas which are most of the time involved into a Bag Of Words strategy, in the literature. Others techniques, Barnard and Forsyth [24] and Duygulu et al. [25] used general purpose segmentation algorithms like Blobworld [26] and Normalized-cuts [27] to extract regions. These algorithms do not always produce good segmentations but are useful for building and testing models. For each segmented region, features such as color, texture, position and shape information are computed. Duygulu et al [25] used Normalized-cuts to segment images and then extracted 33 features from the images.



(a) (b)

Figure 1: Regions of Interest found by the SIFT algorithm. Processing SIFT took 407ms, 60 features were identified and processed

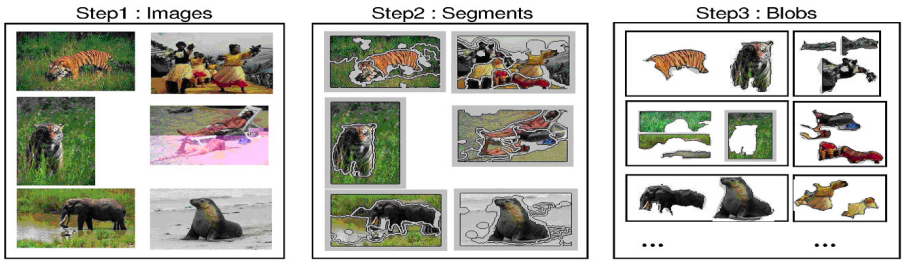


Figure 2: Image preprocessing: Step 2 shows the segmentation results from a typical segmentation algorithm (Blobworld) The clusters in step 3 are manually constructed to show the concept of blobs. Both the segmentation and the clustering often produce semantically inconsistent segments (breaking up the tiger) and blobs (seals and elephants in the same blob). This figure was directly taken from [28] since it illustrates well how to obtain blobs.

1.2.2 Bag Of Words (BoW)

Given a set of training images, a K-means clustering algorithm is applied to cluster the regions on the basis of these features. These clusters which they [26] call ”blobs” compose the vocabulary for the set of images. Each blob is assigned a unique integer to serve as its identifier (analogous to a word’s ASCII representation). All images in the training set can now be represented as a set of blobs from this vocabulary. Figure 2 shows the segmentation and the clustering process for some training images. Given a new test image, it can be segmented into regions and region features can be computed. The blob which is the closest to it in the cluster space is assigned to it. The basic idea of Bag of Words is to depict each image as an orderless collection of local features. For compact representation, a visual vocabulary is usually constructed to describe BoW through the clustering of features. With the visual vocabulary, we can describe the image as a feature vector according to the presence or count of each visual word. Under the supervised learning platform, the feature vector forms the basic visual cue for object and scene classification. In a BoW approach, the classification stage turns into a histogram based classification, although the paradigm is simple, it do not contain any geometry information.

1.2.3 Spatial relationships

Similarity retrieval by spatial image content is done by using multiple objects and their relationships in space. The main idea of this technique is to consider an image as a group of objects or Regions Of Interest (ROI). Therefore, normally this approach requires segmentation process. Once an image is segmented to many regions, we can use both of their local features and spatial features for retrieval. In retrieval by spatial image content, not only the shape, color and texture properties of individual image regions must be similar, but also they must have the same arrangement (spatial relationships).

Set representation and affine transformations The query region features are matched to each target image according to the best fit of affine transformations, see [29] and [22]. These transformations cover situations such as a change in zoom or camera distance to the scene, foreshortening and vertical shear. The advantage is a low computation complexity and the drawback is the linear property of the transformation which captures only linear distortion.

Tree and Graph representation Ideally, the object relationships are described with a graph as the Attributed Relational Graphs (ARGs) or Containment Trees (CT) [30], [31]. Among the more known categories of spatial relationships, we can mention the directional [32], [33], [34], topological [35], geometrical [36], and orthogonal [37] ones. These representations are often invariant to scale, rotation and translation. Therefore, photos can be taken from any views and no strong assumptions are inserted. The flip side of the coin is that such approaches are coupled with matching techniques which are time consuming. Different approaches have been proposed during the last decades to tackle the problem of graph classification. A first one consists in transforming the initial problem in a common statistical pattern recognition one by describing the objects with vectors in a Euclidean space. In such a context, some features (vertex degree, labels occurrence histograms,. . .) are extracted from the graph. Hence, the graph is projected in a Euclidean space and classical machine learning algorithms can be applied [38]. Such approaches suffer from a main drawback: to have a satisfactory description of topological structure and graph content, the number of such features has to be very large and dimensionality issues occur.

Other approaches propose to use embeddings of the graphs in a Euclidean space of a given dimensionality using an optimization process the aim of which is to best fit the distance matrix between each of the graphs. In such cases, a measure allowing graph comparison has to be designed. It is the case for multidimensional scaling methods proposed in [39] and [40].

Another family of approaches also consists in using classical machine learning algorithms. At the opposite of the approaches mentioned above, the graphs are not explicitly but implicitly projected in a Euclidean space, through the use of a similarity measure adapted to the processed data in the learning algorithm.

In such a context, many kernel-based methods such as Support Vector Machine or Kernel Principal Analysis were proposed recently [41], [42]. They consist in designing an appropriate graph-based kernel for computing inner products in the graph space. Many kernels have been proposed in the literature [43], [44], [45]. In most cases, the graph is embedded in a feature space composed of label sequences through a graph traversal. According to this traversal, the kernel value is then computed by measuring similarity between label sequences. Even if such approaches have proven to achieve high performance, they suffer from their computationally intensive cost if the dataset

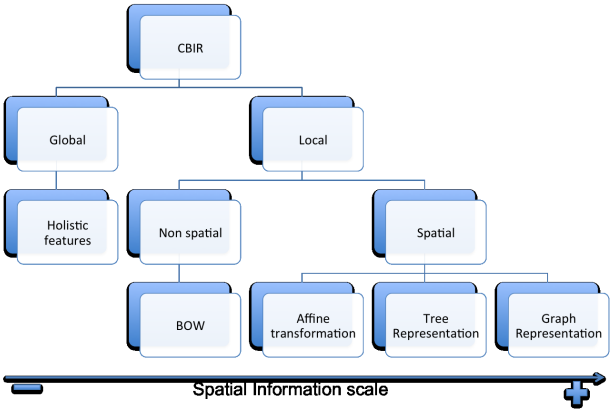


Figure 3: CBIR approach categorization according to the amount of spatial information captured

is large [46]. This problem of computational cost is not inherent to kernel-based methods. It also occurs when using other classification algorithms like k-NN. In conclusion, the problem of classifying graphs requires the use of a fast but yet effective graph distance. In this objective, we used in our experiments the SubGraph Matching Distance ($SGMD$) defined in subsection 2.2.2.

To conclude, figure 3 depicts the different CBIR approaches considering the incorporation of spatial information as a criterion.

1.3 Recent trends on CBIR

To complete this overview on CBIR, we expose some recent trends which investigate three directions for improving visual object-retrieval performance : a) the first direction is related to clustering algorithm improvements b) a second direction is linked to the reduction of the amount of objects to be compared considering spatial information. c) Finally, a third orientation deals with spatial constraint modelling.

1. **Improving the visual vocabulary.** In [29], authors improve the clustering method by using an approximate k-means algorithm. In typical k-means, the vast majority of computation time is spent on calculating nearest neighbors between the points and cluster centers. Philbin et Al replace this exact computation by an approximate nearest neighbor method, and use a forest of 8 randomized k-d trees [47] [48] built over the cluster centers at the beginning of each iteration to increase speed. The algorithmic complexity of a single k-means iteration is then reduced from $O(NK)$ to $O(Nlog(K))$, where N is the number of features being clustered and K the number of clusters.

2. **Pruning the search space by late incorporation of spatial information.** The output from performing a query is a ranked list of images for a significant section of the corpus. In each image, features have been until now considered as a visual bag-of-words and have ignored the spatial configurations of features. Philbin et al [29]investigates re-ranking the top-ranked results using spatial constraints. The spatial verification procedure estimates a transformation between the query

region and each target image, based on how well its feature locations are predicted by the estimated transformation. They then re-rank target images based on the discriminability of the spatially verified visual words.

3. **Modelling spatial representation.** The part-based model initiated by Fischler et al. [49], which later spawned the star-graph [50], [51] and constellation models [52], [53] are examples of that trend that consists in modelling many interactions between image parts. Though we must emphasise that the part-based model considerably differs with our approach, we feel compelled to describe it here because it efficiently models object parts organised in a graphical structure somewhat similar to ours. Part-based model names are given according to the shape of the graph that describes the interactions between object parts. The star-graph [50], [51] considers that each object has a central part to which auxiliary, smaller scale parts are connected. Felzenszwalb et al. [51] combine the star model with histogram of gradient (HOG, [54]) features and a generalisation of SVM to predict quickly and accurately the location of objects and object parts in the challenging PASCAL VOC 2006 dataset [55]. In [51], authors have managed to formulate the problems of model training and part detection as a variant of support vector machine (SVM), coined latent SVM (LSVM). Consequently, both their training and testing phases are relatively fast. Moreover, one of the contributions of [51] over the spring-model of [49] is that the coefficients that quantify the interaction between two different parts are allowed to take negative values, as they are in fact weighting coefficients of an SVM.

4. **Positioning our paper.** Points 1 and 2 are beyond the scope of this paper, here the focus is given to content image representations with respect to spatial constraints. Regarding to point 3, we have favoured conceptually weaker models than the part-based models. The -arguably questionable- reason behind this strategic decision is that databases of today labelled images are likely to grow in size and diversity in the future. Thus, we believe that methods that rely on the comparison to many training samples bears good promises. It should not be necessary to explicitly model parts appearance and relationships, because the amount of data should allow us to make up for intra-class variability. However, several essential comparisons can be drawn between our work and part-based models: first, we rely on local visual features to populate the nodes of our visual graphs. Second, we employ connections between visual features and capture information about these connections for classification. But, this information is not directly stored inside a model, but indirectly, through the image representation. The visual features we extract are arranged into graphical structures from which we infer the image labels. Finally, our work should be firmly distinguished from part-based models, as we do not model the pairwise connections themselves.

1.4 Our local approach

A standard CBIR data flow process relies on three phases. The first one, the extraction of local information aims at finding relevant areas in the image and then to extract features in these regions. All these features are grouped into clusters using a partitional algorithm. Those clustered features form a visual vocabulary that can be used to express the content of an image. Hence, often an image is transformed into a bag of words, and the comparison of two images turns into a distance between histograms

of words. Here in this paper, another point of view is adopted by trying to take into account the spatial relationship between regions of interest. Therefore, a given image is no longer reduced to a set of words but more likely, a graph based representation is built from the image to enrich the model.

Contributions Here, the paper explores the possibility of adding structural information for image retrieval. In our case, the topological question is taken into account early in the system by the use of a Graph-Based Representation. Our representation is invariant to scale, rotation and translation. The contribution of the paper is twofold: Firstly, we propose a combination of existing techniques (for segmentation and feature description) in order to obtain a new image description based on regions. We call this descriptor Invariant Feature From Segmentation IFFS (IFFS). On the other hand, we propose to investigate how structural representations which are invariant to scale and rotation can impact a CBIR system ? Our image descriptor is evaluated in two well-known datasets and compared against a reference method, such as the bag-of-words approach.

1.5 Paper Organization

The next section (section 2) is dedicated to the description of our proposal called IFFS descriptor. Accordingly, blob extraction and organization are described. In these descriptions, explanations about visual features and structured objects comparison are detailed. Section 3 is dedicated to our experimental results, comparing BoW, Tree and Graph based approaches. Finally, a conclusion is given and future works are brought in section 4.

2 Invariant Feature From Segmentation (IFFS) and spatial constraints

An important question is how can one obtain an image vocabulary. In other words, how does one represent every image in the collection using a subset of items from a finite set of items. An intuitive answer to this question is to segment the image into regions, cluster similar regions and then use the regions as a vocabulary. The hope is that this will produce semantic regions and hence a good vocabulary. In our approach, an information extraction stage called Invariant Feature From Segmentation (IFFS) is proposed. The partition into regions is based on a recent statistical region merging algorithm [56] while standard Color, Shape and Texture features are extracted from each region to characterize them. In content-based image retrieval the use of simple features like color, shape or texture is not sufficient. Instead, the ultimate goal is to capture the content of an image via extracting the objects of the image. Usually images contain an inherent structure which may be hierarchical. Once regions and features are extracted, there is still the question of how to organize them to perform a classification stage. We describe two models for image representation and similarity measurement, which take into account content features like color, texture, shape. A CBIR decomposition is proposed in figure 4.

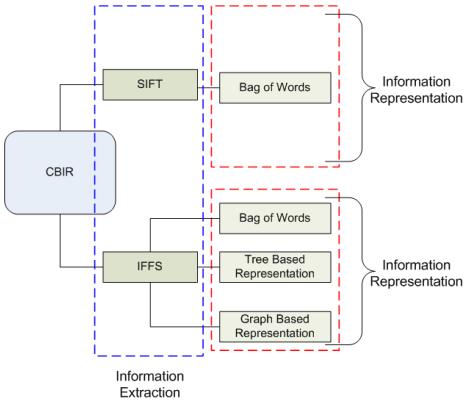


Figure 4: CBIR taxonomy

2.1 Blob Extraction

The blob extraction stage is composed of two phases : Segmentation and visual features extraction.

2.1.1 Segmentation algorithm

Recently, thanks to the increasing speed and decreasing cost of computation, many advanced techniques have been developed for segmentation of color images. In particular we used the Statistical Region Merging [56] algorithm that belongs to the family of region growing techniques with statistical test for region fusion. SRM is based on the following model of image: I is an image with $|I|$ pixels each containing three values (R, G, B) belonging to the set $1, 2, ..., g$. The model considers image I as an observation of perfect unknown scene I^* in which pixels are represented by a family of distributions from which each color level is sampled. In particular, every color level of each pixel of I^* is described by a set of Q independent random variables with values in $[0, g/Q]$. In I^* the optimal regions satisfy the following homogeneity properties:

- inside any statistical region and for any color channel, statistical pixels have the same expectation value for this colour channel;
- The expectation value of adjacent regions is different for at least one color channel.

From this model Nielsen and Nock obtain the following merging predicate:

$$P(R, R') = \begin{cases} true & \text{if } \forall\, a \in R, G, B, \mid \overline{R'_a} - \overline{R_a} \mid \leq b(R) + b(R'); \\ false & otherwise. \end{cases} \tag{1}$$

$$b(R) = g\sqrt{\frac{1}{2Q \mid R \mid}(ln\frac{\mid R_{\mid R\mid} \mid}{\delta})} \tag{2}$$

$\overline{R_a}$ denotes the observed average for color a in region R whereas $R_{|l|}$ is the set of regions with l pixels

The order in which the tests of merging were done follows a simple invariant A:

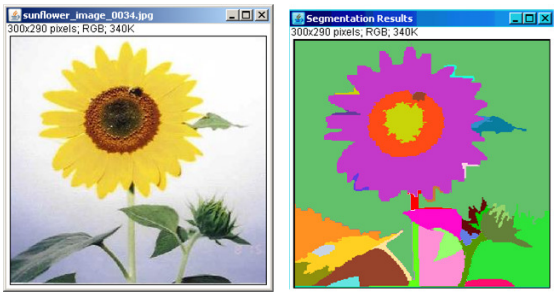


Figure 5: A segmentation result. Processing SRM took 1625ms and 26 features were identified and processed

- When any test between two true regions occurs, that means that all tests inside each region have previously occurred.

In the experiments, A is approximated by a simple algorithm based on gradient of nearby pixels. In particular Nielsen and Nock consider a function f defined as follow:

$$f(p, p') = \max_{a \in R,G,B} f_a(p, p') \tag{3}$$

A simple choice for f_a is:

$$f_a(p, p') = \mid p_a - p'_a \mid \tag{4}$$

The set of the pairs of adjacent pixel (S_I) is sorted according to the value of equation 3. Afterwards the algorithm takes every couple of pixels (p, p') of S_I and if the regions to which they belong $(R(p)$ and $R(p'))$ were not the same and satisfactory equation 1, it merges the two regions. Some image examples segmented by SRM algorithm are shown in figure 5.

2.1.2 Features for visual classification

1. Color features: Color Histograms $< H >$. We discretize the color space of the image such that there are n distinct (discretized) colors. A color histogram H is a vector $< h_1, h_2, ..., h_n >$, in which each bucket h_j contains the number of pixels of color j in the image. Typically images are represented in the RGB color space, and a few of the most significant bits are used from each color channel [57]. The 2 most significant bits of each color channel are considered, for a total of n = 64 buckets in the histogram.

2. Texture features: Co-occurrence matrices $< T >$. Statistical methods use second order statistics to model the relationships between pixels within the region by constructing Spatial Gray Level Dependency (SGLD) matrices [58]. From SGLD matrices, a variety of features may be extracted. The original investigation into SGLD features was pioneered by Haralick et al. [59]. From each matrix, 14 statistical measures are extracted including: angular second moment, contrast, correlation, ... Feature values in all four directions are averaged to build a vector $< T >$ of $4 \times 14 = 56$ components.

3. Shape features: Zernike Moments $< S >$. Zernike polynomials are widely used as basis functions of image moments. Since Zernike polynomials are orthogonal to each other, Zernike moments can represent properties of an image with no redundancy or overlap of information between the moments. Zernike moments are orthogonal and rotation invariant. T. Taxt in [60]. Moments of orders up to 8-11 are needed to achieve a reasonable shape classification. According to this result, our shape feature vector $< S >$ is composed the 13^{th} first Zernike moments.

The complete feature vector $< F >$ is made up of the three feature descriptors defined above. This lead us to a vector of dimension 133:

$$\mid F \mid = \mid H \mid + \mid T \mid + \mid S \mid = 64 + 56 + 13 = 133$$

2.1.3 Motivation of our choices

1. **Segmentation algorithm.** About the segmentation algorithm, SRM [56] is a linear-time fast and simple (yet effective) region growing segmentation algorithm based on an adaptive statistical threshold merging predicate on color channels. It runs fast and handles nicely occlusion and noise.

2. **Color features.** Many color features could be used, however color histograms are frequently used to compare images. Examples of their use in multimedia applications include scene break detection [61], [62] and querying a database of images [63], [64]. Their popularity stems from at least three factors : a) Color histograms are computationally trivial to compute. b) Small changes in camera viewpoint tend not to effect color histograms. c) Different objects often have distinctive color histograms.

3. **Texture features.** On texture classification contests, the co-occurrence matrix is a popular texture method, which was assessed successfully on the publicly available Meastex database [65], [66].

4. Shape features. The first thirteen Zernike invariant moments [67] give us global point of view of segmented regions. They provide sufficient information of shapes which are not too specific to shape details. Zernike moments often describe pretty well shapes. Undoubtedly, they remain on top of shape descriptors, they always achieve good results in shape contests [20].

2.2 Blob Organization

This part is dedicated to image representations and complex object measurements. Two models are described a Containment Tree (CT) and a Region Adjacency Graph (RAG). These paradigms are illustrated figure 6. When dealing with structured objects the question of dissimilarity measure between objects arises. Here a discussion is brought about the compromise between computational complexity and accuracy.

2.2.1 Tree Based Representation (TBR) and Similarity Measure

One way to model images for content-based retrieval is the use of trees representing the structural and content information of the images. To utilize the inherent structure of images for content-based retrieval, we model them as so called containment trees.

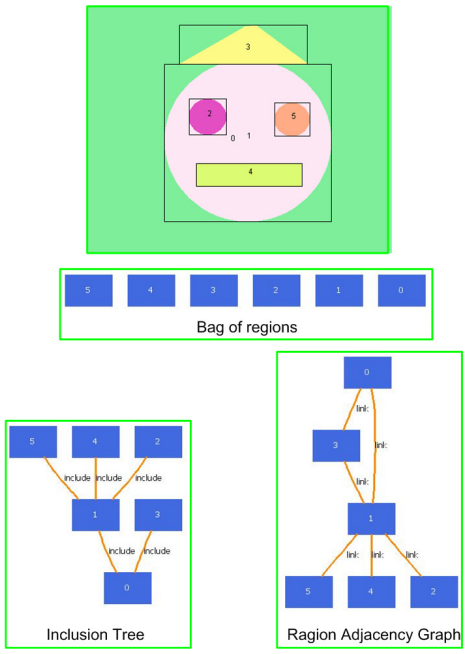


Figure 6: Multiple representations

Containment trees (CTs) model the hierarchical containment of image regions within others. The containment hierarchy is extracted from the set of segments by determining which regions are completely contained in other regions. In this context, a region R_{in} is said to be contained in a region R_{cont} if for every point $p \in R_{in}$ and every straight line $L \ni p$ there exist two points $o_1, o_2 \in R_{cont}$ with $o_1, o_2 \in L$ and o_1, o_2 are on opposite sides of p.

Measuring the distance between two Containment Trees To measure the similarity of containment trees, special similarity measures for attributed trees are necessary. A successful similarity measure for attributed trees is the edit distance. Well known from string matching [68], [69], the edit distance is the minimal number of edit operations necessary to transform one tree into the other. The basic form allows two edit operations, i.e. the insertion and the deletion of a node. In the case of attributed nodes the change of a node label is introduced as a third basic operation. A great advantage of using the edit distance as a similarity measure is that along with the distance value, a mapping between the nodes in the two trees is provided in terms of the edit sequence. The mapping can be visualized and can serve as an explanation of the similarity distance to the user. However, as the computation of the edit-distance is NP-complete [70], constrained edit distance like the Zhang and Shasha edit distance [71] has been introduced. They were successfully applied to trees for web site analysis [72], structural similarity of XML documents [73] or shape recognition [74].

Zhang introduced the constrained edit distance between two trees ($T1$,$T2$) denoted by δ_c, which is defined as an edit distance under the restriction that disjoint subtrees should be mapped to disjoint subtrees. Formally, $\delta_c(T1, T2)$ is defined as a minimum cost mapping $(Mc, T1, T2)$ satisfying the additional constraint, that for all $(v1, w1), (v2, w2), (v3, w3) \in M_c$.

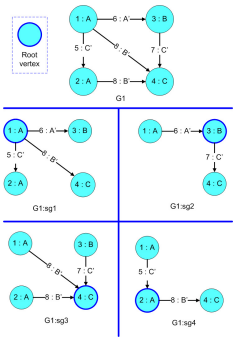


Figure 7: Graph decomposition into subgraph world

- $(v1, v2)$ is a proper ancestor of $v3$ iff $(w1, w2)$ is a proper ancestor of $w3$.

In [75], Zhang presents algorithms for the computing the minimum cost constrained mappings. For the ordered case he gives an algorithm using $O(|T1|.|T2|)$ time.

2.2.2 Graph Based Representation (GBR) and Similarity Measure

Here, an extension of the BoW and the CT methods is proposed. Blobs are structured into an attributed related graph (ARG) in order to take into account spatial relationships between blobs.
An ARG is a graph where its vertices correspond to regions and edges correspond to relationships between regions of images. Both vertices and edges are labeled by attributes corresponding to properties (features) of objects and relationships respectively. To retrieve the similarity of images by using ARGs, it is required to perform a distance measure or a graph matching. Graph matching tolerant to noise and variation is a complicated process with high complexity. In this paper, a graph distance that compromises between accuracy and time consumption is presented. We chose an approximation of the well-known graph edit distance [76], [77] called SubGraph Matching Distance (SGMD) [78]. This sub-optimal solution has the merit to be pretty accurate while keeping the time complexity quite low. Hereafter, we provide the guidelines of this graph distance.

Graph decomposition The subparts for the matching problem can be expressed as follows:

Let G be an attributed graph with edges labeled from the finite set $\{l_1, l_2, ..., l_a\}$. Let SG be a set of subgraphs extracted from G. There is a subgraph sg associated to each vertex of the graph G. A subgraph (sg) is defined as a structure gathering the edges and their corresponding ending vertices from a root vertex. In such a way, the neighborhood information of a given vertex is taken into account. A subgraph represents a local information, a "star" structure from a root node. The mapping of these subparts should lead to a meaningful graph matching approximation. The subgraph extraction is done by parsing the graph which is achievable in linear time through the joint use of the adjacency matrix. The subgraph decomposition is illustrated in figure 7.

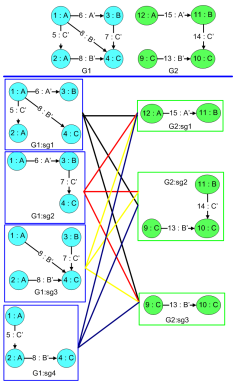


Figure 8: Subgraph matching : A bipartite graph

Subgraph Matching Let $G_1(V_1, E_1)$ and $G_2(V_2, E_2)$ be two attributed graphs. Without loss of generality, we assume that $|SG_1| \geq |SG_2|$. The complete bipartite graph $G_{em}(V_{em} = SG_1 \cup SG_2 \cup \triangle, SG_1 \times (SG_2 \cup \triangle))$, where $\triangle$ represents an empty dummy subgraph, is called the subgraph matching graph of G_1 and G_2. A subgraph matching between G_1 and G_2 is defined as a maximal matching in G_{em}. We define the matching distance between G_1 and G_2, denoted by $SGMD(G_1, G_2)$, as the cost of the minimum-weight subgraph matching between G_1 and G_2 with respect to the cost function c'. This optimal subgraph assignment induces an univalent vertex mapping between G_1 and G_2, such as the function $SGMD : SG_1 \times (SG_2 \cup \triangle) \rightarrow \Re_0^+$ minimized the cost of subgraph matching. If the numbers of subgraphs are not equal in both graphs, then empty "dummy" subgraphs are added until equality $\mid G_1 \mid = \mid G_2 \mid$ is reached. The cost to match an empty "dummy" subgraph is equal to the cost of inserting a whole unmapped subgraph ($c'(\emptyset, sg)$). The approximation lies in the fact that the vertex mapping is not executed on the whole structure, but more likely for subparts of it. The node matching is only constrained by the assumption of "close" neighborhood imposed by the subgraph viewpoint of a vertex. This paper adopts a "Divide and Conquer strategy" and an example of graph matching is proposed in figure 8.

3 Experiments

In this section, our graph based approach was benchmarked and measured up to a conventional BoW methods using both $SIFT$ and $IFFS$ as information extraction systems. In a two-step mechanism, we started to analyze the vocabulary size impact choosing the best parameters and finally we compared both Bag of Words and graph based representation solutions. A pattern recognition stage was undertaken to analyze the behavior in classification. The database images are ranked in the ascending order of their distance to the query image, with the top k images returned. Two publicly available databases, Coil-100 and Caltech-101, are used to achieve our benchmark (ie. section 3.2).

In this practical work, the tree distance approximation was provided by Stephen Wan, Macquarie University in Australia (Reference [79]) and the SIFT algorithm is an ImageJ plug-in publicly available [80] while others methods where re-implemented by us from the literature. The methods were implemented in Java 1.5 and run on a

Notation	Method	Representation	Distance
$SIFT_{BoW}$	SIFT	Bag of Words	Euclidean
$IFFS_{BoW}$	IFFS	Bag of Words	Euclidean
$IFFS_{Tree}$	IFFS	Containment Tree	Zhang&Shasha tree distance
$IFFS_{GBR}$	IFFS	Region Adjacency Graph	SubGraph Matching Distance

Table 1: Distance between images.

2.14GHz computer with 2G RAM. A prototype version can be downloaded ont the projetct website[1] . For the comprehension of theses tests, we first introduce notations that will make the reading much simpler. A dissimilarity measure between images is a function :

$$d : X \times X \rightarrow \Re$$

where X is an image. We report in table 1, the notations derived from this general form.

3.1 Protocol

- In the first experiment, an image classification stage was carried out. Let $X_{tr} = \{x_1, .., x_n\} \ni R^P$ a crispy labeled set of training data. Our presumption is that X_{tr} contains at least one point with class label $j,\ 1 < j < C$. Let x be an unlabeled object that we wish to label as belonging to one of C classes. The standard nearest-neighbor (1-NN) classification rule assigns x to the class of the $most\ similar$ prototype in a set of labeled training data (or reference set). Why do we use a nearest prototype classifier? Because the graph classification problem is defined in a dissimilarity space, the 1-NN classifier can be used to categorize objects in such a space, in addition, it is intuitive, simple, and often, pretty accurate. Hereafter, $E_{np}(X_{tr}; X_{test})$ denotes the test error committed by the 1-NN rule that uses X_{test} when applied to the training data. For a better understanding of the time consumption and the classification behavior, the number of classes influence is evaluated. Each data set is split up into 6 subsets containing from 5 to 100 classes (Number of classes: 5,10,20,40,80,100). These 6 folds allow us to extend our benchmark. It makes feasible, for each approach, an estimation of the generalization power over small or large data sets.

- The last experiment consists in a Content-Based Image Retrieval process. Images are ranked in the ascending order of their distance to a given query image. All these responses ($|X_{tr}|$ responses) to the query are returned to compute two measures of performance, named, Precision and Recall. Precision and recall are two widely used statistical classifications. Precision can be seen as a measure of exactness or fidelity, whereas Recall is a measure of completeness. Algorithm 1 states clearly how to obtain the values.

3.2 Data set descriptions

In this paper, we consider two different labeled image databases. The well-known caltech-101 database [81]. Pictures of objects belonging to 101 categories (figure 10).

[1] http://alpage-l3i.univ-lr.fr/

Algorithm 1 Precision and Recall computation

Require: For the i^{th} query x_{ij} belonging to the class j from X_{test}.
Ensure: There exists exactly $|X_{tr}|$ pairs of precision and recall measures.
1: **For** $k = 1$ **To** k= $|X_{tr}|$ by **Step**=1 **Do**
2: Get the k top responses and put them into a list called O
4: Within the list O compute the precision and recall values.
5:

$$precision_{ik} = \begin{cases} \frac{|\{\textit{Relevant Documents}\} \cap \{\textit{Retrieved Documents}\}|}{|\{\textit{Retrieved Documents}\}|} \\ \\ \frac{|\textit{Correctly Labelled Documents}|}{k} \end{cases}$$

6:

$$recall_{ik} = \begin{cases} \frac{|\{\textit{Relevant Documents}\} \cap \{\textit{Retrieved Documents}\}|}{|\{\textit{Relevant Documents}\}|} \\ \\ \frac{|\textit{Correctly Labelled Documents}|}{|\textit{ Documents of class } j \textit{ }|} \end{cases}$$

7: **End For**

About 40 to 800 color images per category. Most categories have about 50 images. The training images were hand labeled to create a consistent ground truth. Note that we consider completely general lighting conditions, camera viewpoint, scene geometry, object pose and articulation. Our database was split randomly into roughly 75% training, 25% validation sets, while ensuring approximately proportional contributions from each class. More information about this data set is presented in table 2. The COIL-100 database [82] consists of images of 100 different objects. The objects were placed on a motorized turntable against black background. The turntable was rotated through 360° to vary object pose with respect to a fixed color camera. Images of the objects were taken at pose intervals of 5°. This corresponds to 72 poses per image. Figure 9 shows an example image of each class. Randomly, for each class of object, 18 images are withdrawn from the initial set to constitute a test set. This leads us to a training set of 5400 images and a test base of 1800 items.

These two sets of data are fairly different and represent an heterogeneous environment to prove the merit of our systems. The Coil-100 database is known to be relatively simple since no backgrounds are considered and images within the same class are derived from a single original object, on the contrary, the caltech-101 set is more complex, a same concept gathers different kind of images from different sources.

3.3 A classification context

Back on track, we keep in mind that the final purpose is to perform a classification stage in order to evaluate the relevance of the image models. Based on the data sets described in section 3.2, a 1-NN rule is applied to obtain the number of correctly classified instances (CCI) and the corresponding classification rate. Firstly, the number of words impact is investigated and the results are illustrated in figures 11 and 12 for the Coil-100 and Caltech-101 databases respectively. Then a comparison between the four image distances is brought considering the best number of words for each method. These results are shown in figures 13.

Figure 9: Columbia University Image Library

Table 2: Characteristics of the data set used in our computational experiments

	Caltech-101	Coil-100
$\mid Training \mid$	6821	5400
$\mid Test \mid$	2323	1800
IFFS: Feature Length	133	133
IFFS: Average number of nodes	31.34	14.10
IFFS: Average number of edges	72.15	25.905
SIFT: Feature Length	128	128
SIFT: Average number of interest points	121.57	40.02

Figure 10: Image Samples from the Caltech-101 Data set. The 101 object categories and the background clutter category. Each category contains between 45 to 400 images. Two randomly chosen samples are shown for each category. The categories were selected prior to the experimentation, collected by operators not associated with the experiment, and no category was excluded from the experiments. The last row shows examples from the background dataset. This dataset is obtained by collecting images through the Google image search engine (www.google.com). The keyword "things" is used to obtain hundreds of random images. Complete datasets can be found at http://vision.caltech.edu

Complete results figures:

- Figure 11 gathers four histograms, one for each method exposed to our evaluation framework ($SIFT_{BoW}$, $IFFS_{BoW}$, $IFFS_{Tree}$, $IFFS_{GBR}$). This complete test aims at underlying the influence of the vocabulary size parameter on the recognition rate. The scalability question is also addressed by increasing progressively the number of classes. In this way, the behavior of each approach is depicted as the problem becomes more and more complex. These tests were run on the Coil-100 set.
- Figure 12 reflects the recognition rate evolution according to the number of words and the number of classes for the Caltech-101 database.

Summary results figures:

- Figure 13 expresses the best results in classification obtained for the most suited number of words. It is the quintessence of the results over the two databases, hence, it makes the comparison more readable and clearer.
- Figure 14 presents how many words are needed for each method to provide their best accuracy level. It shows how sensitive and greedy are the methods about this question of the number of clusters.

Number of words impact Tests on the number of words were carried out. Performance in classification according the number of words (*w*) are presented in figure 14. The question of the vocabulary size is an important issue. Here, a decision of tuning the parameter *w* from 4 to 1024 was taken. In this way, we expect to cover a wide range of possibilities. A first comment states that structural approaches reach their maxima with a smaller number of clusters than BoW methods. Reducing the vocabulary size put more weight on the graph data structure while a large number of words is highlighting the information carried by each regions. A compromise between the feature expressivity and the importance given to the spatial organization has to be found. As an example, too many words may turn the representation very sensitive to noises and small variations, on the other hand, if no feature is extracted from the regions then only the structure is taken into account. Those extrema are representative of how the vocabulary size can impact the classification process.

The histograms presented in figure 14 corroborates the following hypothesis, when the number of classes increases the vocabulary size should be extended too. Bigger is the problem more words are needed to describe it. However, our experiments pointed out that $IFFS_{GBR}$ and $IFFS_{Tree}$ needed a smaller set of words than BoW for the same configuration to reach their best performances.

Recognition rate comparison

GBR vs BoW

In the meantime, each database were divided into six subsets to analyze the number of classes influence. The figure 13 denotes a straightforward fact, a high number of classes leads to a decrease of the performances as the problem becomes more complex. Contrarily to our first thoughts, structural based representation did not overcome the BoW methods in terms of accuracy. Over the two databases, results of BoW systems

Table 3: Average results over the two databases according to the accuracy criterion and time consumption.

Criterion	$SIFT_{BoW}$		$IFFS_{BoW}$		$IFFS_{Tree}$		$IFFS_{GBR}$	
	Coil	Caltech	Coil	Caltech	Coil	Caltech	Coil	Caltech
Accuracy(%)	76.25	50.61	95.03•	46.86	78.86	36.68◦	90.68•	44.28
Time(s)	17604	14457	24358	19069	645567◦	162889◦	898279 ◦	250570◦

• Statistically significantly better than the reference system $(SIFT_{BoW})(\alpha = 0.05)$

◦ Statistically significantly worse than the reference system $(SIFT_{BoW})(\alpha = 0.05)$

outperform the structured ones. This leads us to the question: does structure really matter when indexing natural scene images? The main advantage of a description of patterns by graphs instead of vectors is that graphs allow for a more powerful representation of structural relations. However, in natural images, it appears that the structure may not be stable enough and this variability might be misleading. Nevertheless, the use of a GBR method is recent in CBIR, and we can say that they achieve reasonable results for a "new born" solution. They can obtain similar or slightly under performance than BoW. When at the same time, BoW methods are mature, they have been introduced decades ago in CBIR, they have the age benefits. Furthermore, these encouraging recognition rates reached by GBR methods can be improved, it does exist a rich literature dealing with the insertion of spatial information into graph edges. We can mention GBR methods using Bi-dimensional Allen Algebra (Ref. [83]) or Delaunay triangulation (Ref. [84]). In addition, the Region Adjacency Graph could be swapped for a neighboring graph or a visibility graph for instance, but all these variations on the same theme are beyond the scope of this paper. Here, the objective was to expose that our results are encouraging enough and it leave is plenty of rooms for progress in this direction. Finally, graphs lead to new kind of services, the graph matching problem can be used to locate sub parts of an image from a crop image as a query. All these points converge to state the worth of investigating the graph tools in a CBIR context. A comprehensive comparison is provided in table 3. This table sums up the information according the following metric (Eq.5). The mean value of the best results over the 6 subsets.

$$\overline{E_{np}} = mean\left(\sum_{i=1}^{6} \min_{w} \left(E_{np}\left(X_{tr_i}; X_{test_i}\right)\right)\right) \tag{5}$$

It turns out that classification accuracy can be improved by $IFFS_{GBR}$ compared to the reference system, that is to say $SIFT_{BoW}$, and this, on all number of classes levels. Note that 2 out of 3 improvements are statistically significant.

Independence inter methods

In this experiment, we aim at understanding whether the methods make the same mistakes or not; if the methods decide wrong at the same time or not. On the Caltech-101 database, we perform a χ^2 test of independence. A test of independence assesses whether paired observations on two variables, expressed in a contingency table, are independent of each other -for example, whether $IFFS_{BoW}$ differs in the decision with $IFFS_{GBR}$. The contingency table, in a context of classification, is also called confusion matrix. Each column of this matrix represents the number of occurrences of an estimated class, while each line denotes the number of occurrences of a real class. From the confusion matrix, we derive the construction of what we call a dependence matrix.

	χ^2 test	df	$p-value$
$IFFS_{BoW}$ *vs* $IFFS_{GBR}$	9922	10000	0.209

Table 4: χ^2 independence test between a Graph-based method and a Bag of Words approach.

This latter reflects the dependence of two classifiers based on different representations. In our case, each column of this matrix represents the number of occurrences of an estimated class by the method one, while each line denotes the number of occurrences of an estimated class by the method two.

In this case, an "observation" consists of the values of two outcomes and the null hypothesis is that the occurrence of these outcomes is statistically independent. Each outcome is allocated to one cell of a two-dimensional array of cells (called a table) according to the values of the two outcomes. The "theoretical frequency" for a cell, given the hypothesis of independence, is

$$\chi^2 = \frac{(O_{ij} - E_{ij})^2}{E_{ij}}$$

In our case, the observed value "O" corresponds to the value of the dependence matrix whereas the theoretical occurrence is defined by the average. In each cell, the expected value E_{ij} is equal to the sum of each element of the line i multiplied by the sum of the elements of the column j, divided by N.

$$E_{ij} = \frac{x_i \times y_j}{|X_{test}|}$$

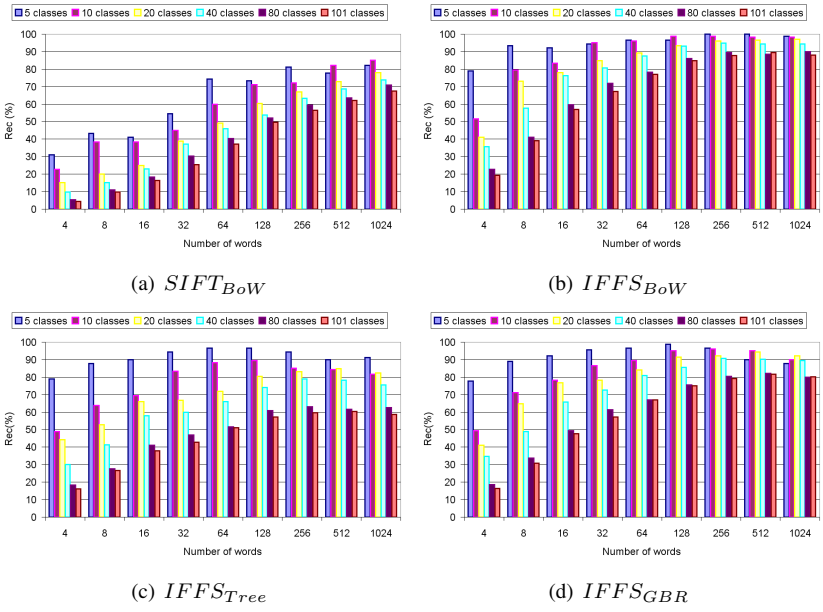
The expected value (E) can be seen as the wanted value in case of independence.

We computed the χ^2 for the following setting: $IFFS_{BoW}$ vs $IFFS_{GBR}$. We consider a null hypothesis of independence ($H0$) between the two methods and then, we compute, by means of a one-tailed statistical hypothesis test, the probability (p-value) of getting a value of the statistic as extreme or more extreme than observed by chance alone, if $H0$ is true. Results are presented in table 4. We compare the χ^2 score with the theoretical χ^2 distribution (degree of freedom (k=10000), risk level (α =0.05)), $\chi^2_{\alpha=0.05,k=10000}$=10233.8). $\chi^2 < \chi^2_{\alpha=0.05,k=10000}$, so we can say that the hypothesis $H0$ of independence can be accepted in with a risk of 5%. The calculated p-value exceeds 0.05, so the observation is consistent with the null hypothesis, the deviation from expected outcome is just small enough to be reported as being "not statistically significant at the 5% level".

In fact, we draw the reader's attention to de-correlated methods, they are likely to be combined to perform better. Inspired from [29] and stimulated by these results of independence, an interesting work will come up. It would aim at speeding up the system by computing at first a BoW method and later in a second time, to process a re-ranking stage with the top k responses integrating spatial information through the use of our graph-based approach. To avoid sequential comparison of the query with all items stored in the archive.

IFFS vs **SIFT**

A comment on the good behavior of IFFS as an extraction information system. Hence, figure 13 validates the join use of an efficient segmentation algorithm (SRM) and dis-



(a) $SIFT_{BoW}$ (b) $IFFS_{BoW}$

(c) $IFFS_{Tree}$ (d) $IFFS_{GBR}$

Figure 11: On the Coil-100 database : Recognition rate in function of the number of classes and the number of clusters.

tinctive features. On the Coil-100 database $IFFS_{BoW}$ overrides $SIFT_{BoW}$ with a significant level. Nevertheless, the power of generalization of this statement is limited by the superiority of the SIFT process on the Caltech-101 data sets.

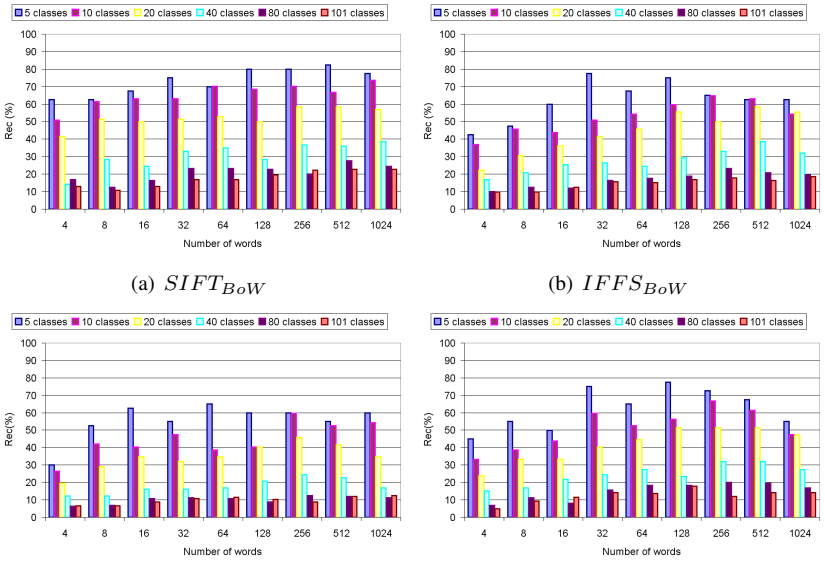
3.4 A CBIR Context

Precision is defined as the ratio of retrieved positive images to the total number retrieved. Recall is defined as the ratio of the number of retrieved positive images to the total number of positive images in the corpus. The precision and recall in a multi-class problem is defined through multi-levels (or j is greater than 1). The overall average precision and recall over all classes j can be evaluated by the macro-average, which first calculates the precision and recall on each class j followed by a calculation of the average information on the C classes.

$$precision = \frac{\sum_{j=1}^{C} precision_j}{C}$$
$$recall = \frac{\sum_{j=1}^{C} recall_j}{C}$$

To evaluate the performance we use the average precision (AP) measure computed as the area under the precision-recall curve. An ideal precision-recall curve has precision 1 over all recall levels and this corresponds to an average precision of 1. The AP scores is used as a single number to evaluate the overall performance. Results are reported in table 5.

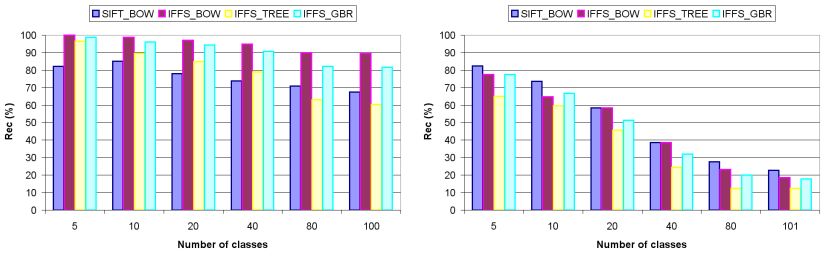
On both databases, precision and recall values are computed and displayed in the figure 15.



(a) $SIFT_{BoW}$ (b) $IFFS_{BoW}$

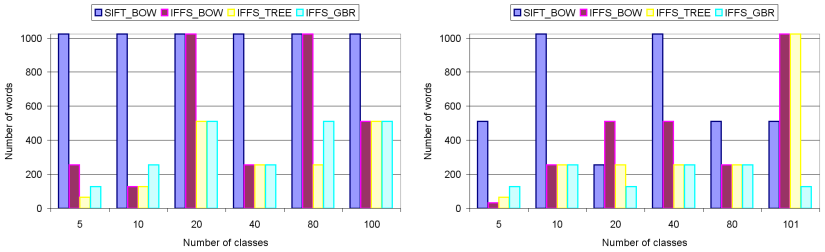
(c) $IFFS_{Tree}$ (d) $IFFS_{GBR}$

Figure 12: On the Caltech-101 database : Recognition rate in function of the number of classes and the number of clusters.



(a) Coil-100 (b) Caltech-101

Figure 13: Comparison between CBIR methods. Summary of results obtained with the best number of words for each method.

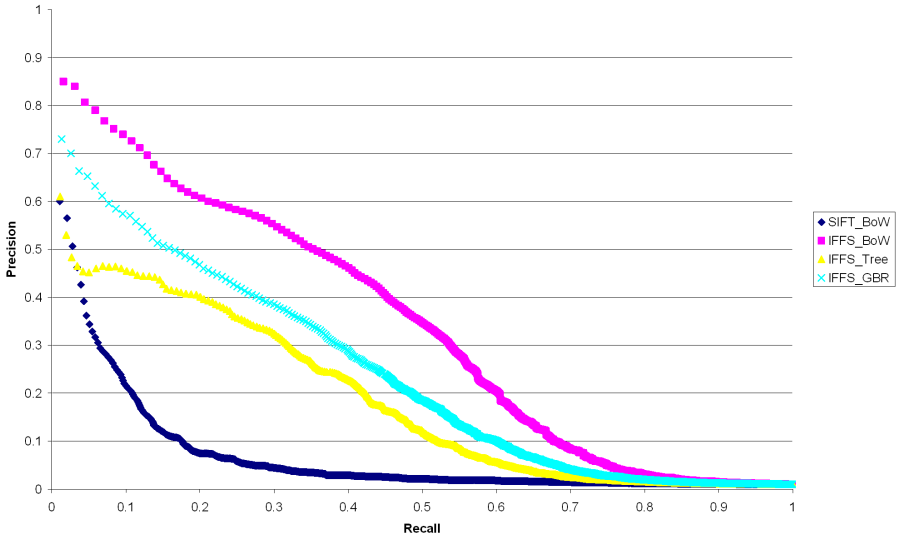


(a) Coil-100 (b) Caltech-101

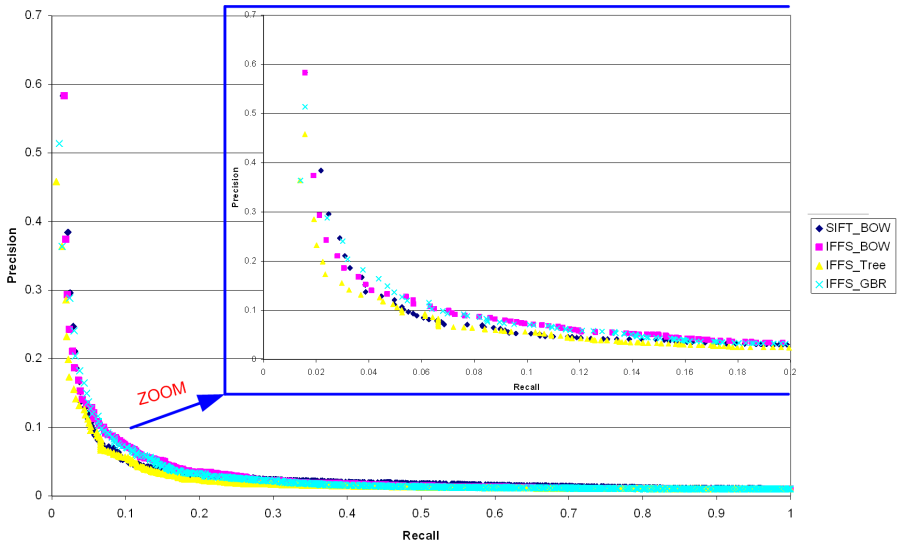
Figure 14: Comparison between the number of words in used by the methods.

	$SIFT_{BoW}$	$IFFS_{BoW}$	$IFFS_{Tree}$	$IFFS_{GBR}$
$Coil-100$	0.0640	0.3242	0.1818	0.2288
$Caltech-101$	0.0314	0.0327	0.0279	0.0335

Table 5: Average Precision (AP) measure. A comparison of the performance of the four methods.



(a) Results on Coil100 database



(b) Results on a 20 classes subset from the Caltech-101

Figure 15: Precision and Recall curves.

3.5 Analysis and discussion

The results are somehow promising with respect to the IFFS approach. It clearly outperforms the $SIFT_{BoW}$ approach in the Coil database. This result could be expected as images are more easily segmented in this database. However, in the Caltech database, where segmentation into regions is more challenging the SIFT obtains better results. On the other hand, the figure 15b puts forward that IFFS does not declare forfeit and tends to get a better precision when the recall is increased. This last comment is re-enforced by measures given in table 5. The AP score of $IFFS_{BoW}$ ($AP_{IFFS_{BoW}}$) is slightly greater than $AP_{SIFT_{BoW}}$.

Concerning the structural representations results are somehow encouraging. They are a bit lower than the other approaches, and only in the Coil database are slightly better than BoW with SIFT, but clearly worse than BoW with IFFS, which could be the reference method in this case, as the graph representation is built on the top of IFFS.

These poor results of the structural approaches seem to refute the main initial hypothesis about the use of this type of graph representation. Nevertheless, the IFFS method is an interesting contribution since it makes possible the organization into graph or tree whereas SIFT is too versatile to be laid out into a complex structure (Too many key-points occur when running sift on an image). Taking into account, the good results on COIL-100, a discussion arises on the kind of images where IFFS method can be useful. In addition, the idea of completing this representation with structural information is also promising. There is not much work in this direction so far. Structural representations stand as a kind of alternative approach with some preliminary results, but to be further investigated. Graph-Based Representation of an image is a rich domain, relations between regions or points of interest can be modeled in many ways, among them, we can cite the representations issued from Delaunay triangulation [84], Allen algebra [83] or a neighboring graph.

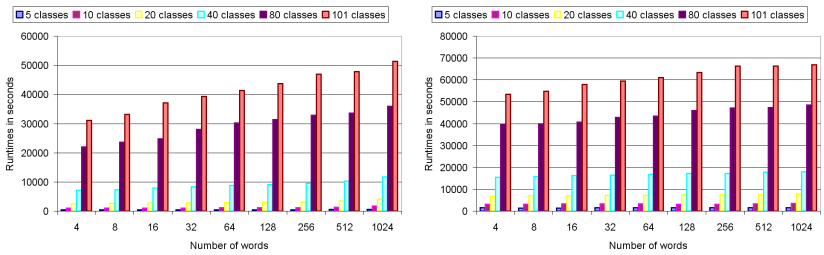
3.6 Time complexity

The graph matching distance ($IFFS_{GBR}$) can be calculated in $O(n^3)$ time in the worst case. To calculate the matching distance between two attributed graphs G_1 and G_2, a minimum-weight matching between the two graphs has to be determined. This is equivalent to determining a minimum-weight maximal matching in the subgraph matching of G_1 and G_2. To achieve this, the method of Kuhn [85] and Munkres [86] can be used. This algorithm, also known as the Hungarian method, has a worst case complexity of $O(n^3)$, where n is the number of probes in the larger one of the two graphs. On the other hand, the histogram distance (used in $IFFS_{BoW}$, $SIFT_{BoW}$) is processed in linear time in function of the number of bins that composes the histogram. A way to compare the computational cost of the different types of distance was to undertake an empirical study on the classification stage. The figure 16 depicts a comparison of the runtime execution according to the kind of distances. This test was performed during the classification phase on the Coil-100 database. It takes into account the computation of regions of interest (IFFS or SIFT) and the distance calculation between image representations.

A first comment aims at illustrating the high time consumption of the graph and tree distances. These techniques are computationally more intensive than others. Structural approaches may fail to face the scalability dilemma in the cases of industrial applications, although their computations remain in polynomial time. Another point illustrated

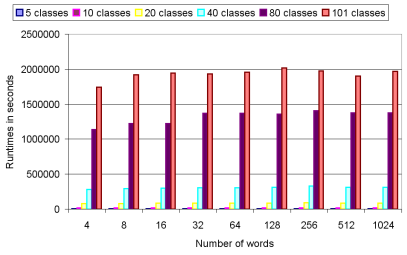
Average response time in seconds	$SIFT_{BoW}$	$IFFS_{BoW}$	$IFFS_{Tree}$	$IFFS_{GBR}$
$Coil-100$ and 1024 words	28	37	1095	1555

Table 6: Average response time of a query. A speed comparison of the four methods at the worst case.

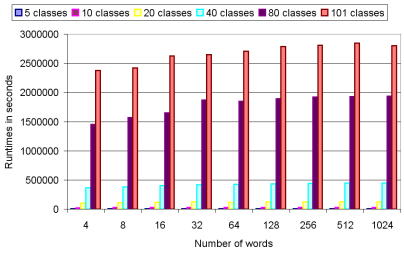


(a) $SIFT_{BoW}$

(b) $IFFS_{BoW}$



(c) $IFFS_{Tree}$



(d) $IFFS_{GBR}$

Figure 16: On the Coil-100 databases : Runtimes in function of the number of classes.

by the figure 16 is the effect of the vocabulary size on the histogram length. Simply, higher is the number of words and larger are the histograms. In average, a linear relation exists between the number of clusters and the time complexity of histogram based methods. Finally, $SIFT_{BoW}$ runs slightly faster than $IFFS_{BoW}$ (at worst case: 51000 seconds vs 67000 seconds). This time gap is low enough to not reject $IFFS$ as a suitable solution considering the significant accuracy gain it can imply. This little loss of speed does not discourage the application of a color segmentation algorithm to extract blobs. For the sake of clarity, we provide in table 6 the average response time of a query. This is relevant to the time the system takes to react to a given input.

4 Conclusion

In this paper, a graph based representation was proposed in a CBIR context. From a partition into regions processed by an efficient segmentation algorithm, a Region Adjacency Graph was built to consider spatial relationships between regions. Each region is characterized using a set of features based on the Color, Texture and Shape. A K-means clustering algorithm is applied to cluster the regions on the basis of these features. These clusters which we call ”blobs” compose the vocabulary for the set of images. Each blob is assigned to a unique integer to serve as its identifier. An efficient and yet fast dissimilarity measure between structured data was presented to compare

attributed relational graphs. The whole method was compared to conventional Bag of Words strategies and to another structural approach based on Containment Trees. The Graph Based Approach overcame the Tree Based one, however it gave similar or slightly under results than BoW methods. BoW systems have been introduced a decades ago into CBIR applications while GBR are quite new in this field of science. Nevertheless, experiments showed that a structural approach requires a fewer number of words to reach its best performance.

A closer look should be given to the relation between regions. For instance, a future promising work concerns the enrichment of the graph representation by the use of a bi-dimensional Allen Algebra. This description inserted on the edge labels should provide a better representation of the region layout.

In addition, we want to express the special interest given to Graph Based Representation in CBIR context, as a final goal, GBR could offer the possibility to spot sub-parts of images from an image portion of the query image. The flip side of coin is an over-load of complexity which leads to a higher time consumption.

Inspired from [29] and stimulated by the results of independence between BoW and Graph methods, an interesting work will come up. It would aim at speeding up the system by computing at first a BoW method and later in a second time, to process a re-ranking stage with the top k responses integrating spatial information through the use of our graph-based approach. A sequential comparison of the query with all items stored in the archive could be avoided. Our last perceptive is to avoid discarding information regarding the visual features themselves by quantisation of the feature space. This could be envisaged through inexact graph matching techniques such as graph edit distance.

References

[1] J. Eakins, Retrieval of Still Images by Content, Lectures on Information Retrieval (2001) 111–138.
URL http://dx.doi.org/10.1007/3-540-45368-7-6

[2] I. L. C. I.K. Sethi, Mining association rules between low-level image features and high-level concepts, Proceedings of the SPIE Data Mining and Knowledge Discovery vol. III (2001) 279–290.

[3] S. H. L. S.K. Chang, Picture indexing and abstraction techniques for pictorial databases, IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI) 6 (4) (1984) 475–483.

[4] M. F. J. H. W. N. D. P. W. E. C. Faloutsos R. Barber, Efficient and effective querying by image content, Journal of Intelligence Information System 3 (3-4) (1994) 231–262.

[5] S. S. A. Pentland R.W. Picard, Photobook: content-based manipulation for image databases, International Journal on Computer Vision 18 (3) (1996) 233–254.

[6] A. Gupta, R. Jain, Visual information retrieval, Commun. ACM 40 (5) (1997) 70–79. doi:http://doi.acm.org/10.1145/253769.253798.

[7] S. F. C. J.R. Smith, VisualSeek: a fully automatic contentbased query system, Proceedings of the Fourth ACM International Conference on Multimedia (1996) 87–98.

[8] N. W.Y. Ma B. Manjunath, A toolbox for navigating large image databases, Proceedings of the IEEE International Conference on Image Processing (1997) 568–571.

[9] G. W. S. J.Z. Wang J. Li, Aemantics-sensitive integrated matching for picture libraries, IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI) 23 (9) (2001) 947–963.

[10] D. D. F. F. Long H.J. Zhang, Fundamentals of content-based image retrieval, Multimedia Information Retrieval and Management 1390 (2003) Springer, Berlin.

[11] Y. Rui, T. S. Huang, S.-F. Chang, Image retrieval: current techniques, promising directions, and open issues, Journal of Visual Commun. Image Representation 10 (4) (1999) 39–62.

[12] M. De Marsicoi L. Cinque, S. Levialdi, Indexing pictorial documents by their content: A survey of current techniques., Image and Vision Computing 15 (1997) 119–141.

[13] Y. Rui T. Huang, S. Chang, Image retrieval Past, present, and future, In International Symposium on Multimedia Information Processing.

[14] Y. Rui T. Huang, S. Chang, Image retrieval: current techniques, promising directions and open issues, Journal of Visual Communication and Image Representation 39-62.

[15] D. B. H. Muller N. Michoux, A. Geissbuhler, A review of content-based image retrieval systems in medical applications clinical benefits and future directions, International Journal of Medical Informatics 73 (2004) 1–23.

[16] R. Sriram J. M. Francos, W. A. Pearlman, Texture coding using a wold decomposition based model, IEEE Transactions of Image Processing 5 (1996) 1382–1386.

[17] D. Marr, H. K. Nishihara, Representation and Recognition of the Spatial Organization of Three-Dimensional Shapes, Proceedings of the Royal Society of London. Series B, Biological Sciences 200 (1978) 269–294.

[18] M. Brady, Criteria for Representations and of shape, Human and Machine Vision Academic (1993) 39–84.

[19] W. K. Pratt, Bibliography, in: Digital Image Processing (Third Edition), 2002, pp. 717–722.
URL http://dx.doi.org/10.1002/0471221325.biblio

[20] E. Valveny, P. Dosch, Symbol Recognition Contest: A Synthesis, Graphics Recognition (2004) 368–385.
URL http://www.springerlink.com/content/65dlhypeg3ha96ex

[21] R. Datta, D. Joshi, J. Li, James, Z. Wang, Image retrieval: Ideas, influences, and trends of the new age, ACM Computing Surveys 39 (2006) 2007.

[22] D. Lowe, Distinctive image features from scale-invariant keypoints, Journal on Computer Vision 60(2):91.

[23] et al E.Nowak, Sampling strategies for bag-of-features image classification, Computer Vision (ECCV).

[24] K. Barnard, D. Forsyth, Learning the semantics of words and pictures, IEEE International Conference on Computer Vision (ICCV) 2 (2001) 408–415. doi:10.1109/ICCV.2001.937654.
URL http://dx.doi.org/10.1109/ICCV.2001.937654

[25] P. Duygulu, K. Barnard, J. de Freitas, D. Forsyth, Object Recognition as Machine Translation: Learning a Lexicon for a Fixed Image Vocabulary, Computer Vision ECCV 2002 (2002) 349–354doi:10.1007/3-540-47979-1_7.
URL http://dx.doi.org/10.1007/3-540-47979-1_7

[26] C. Carson, M. Thomas, S. Belongie, J. Hellerstein, J. Malik, Blobworld: a System for Region-based Image Indexing and Retrieval.
URL http://portal.acm.org/citation.cfm?id=893714

[27] J. Shi, J. Malik, Normalized Cuts and Image Segmentation, IEEE Transactions on Pattern Analysis and Machine Intelligence 22 (8) (2000) 888–905.
URL http://citeseer.ist.psu.edu/shi97normalized.html

[28] J. Jeon, V. Lavrenko, R. Manmatha, Automatic image annotation and retrieval using cross-media relevance models.
URL http://citeseer.ist.psu.edu/jeon03automatic.html

[29] J. Philbin, O. Chum, M. Isard, J. Sivic, A. Zisserman, Object retrieval with large vocabularies and fast spatial matching, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.

[30] E. G. M. Petrakis, C. Faloutsos, K.-I. D. Lin, ImageMap: An Image Indexing Method Based on Spatial Similarity, IEEE Transactions on Knowledge and Data Engineering 14 (2002) 979–987. doi:http://doi.ieeecomputersociety.org/10.1109/TKDE.2002.1033768.

[31] E. G. M. Petrakis, Design and Evaluation of Spatial Similarity Approaches for Image Retrieval, Image and Vision Computing 20 (2001) 59–76.

[32] C.-C. Chang, C.-F. Lee, A spatial match retrieval mechanism for symbolic pictures, Journal of Systems and Software 44 (1) (1998) 73–83. doi:DOI: 10.1016/S0164-1212(98)10044-4.
URL http://www.sciencedirect.com/science/article/B6V0N-3V8M1JH-7/2/d26d59a0a

[33] P.-W. Huang, C.-H. Lee, Image Database Design Based on 9D-SPA Representation for Spatial Relations, IEEE Transactions on Knowledge and Data Engineering 16 (2004) 1486–1496. doi:http://doi.ieeecomputersociety.org/10.1109/TKDE.2004.92.

[34] S. BERRETTI, A. DEL BIMBO, E. VICARIO, Weighted walkthroughs between extended entities for retrieval by spatial arrangement, IEEE transactions on multimedia 5 (1) (2003) 52–70.
URL http://cat.inist.fr/?aModele=afficheN&cpsidt=14748191

[35] J. Z. Li, M. T. Özsu, M. Tamer, Point-Set Topological Relations Processing In Image Databases, in: In First International Forum on Multimedia and Image Processing, 1998, pp. 51–54.

[36] D. S. Guru, P. Nagabhushan, Triangular spatial relationship: a new approach for spatial knowledge representation, Pattern Recognition Letters 22 (9) (2001) 999–1006. doi:DOI: 10.1016/S0167-8655(01)00043-5.
URL http://www.sciencedirect.com/science/article/B6V15-435KJR7-7/2/2cd4e4643

[37] S.-K. Chang, E. Jungert, A spatial knowledge structure for image information systems using symbolic projections, in: ACM '86: Proceedings of 1986 ACM Fall joint computer conference, IEEE Computer Society Press, Los Alamitos, CA, USA, 1986, pp. 79–86.

[38] A. Papadopoulos, Y. Manolopoulos, Structure-Based Similarity Search with Graph Histograms, Structure-based similarity search with graph histograms (1999) 174–178.

[39] E. Bonabeau, Graph multidimensional scaling with self-organizing maps, Information Science 143 (1-4) (2002) 159–180.

[40] M. F. Cox, M. A. A. Cox, Multidimensional Scaling. Chapman and Hall, Quantitative Applications in the Social Sciences.

[41] H. Kashima, Y. Tsuboi, Kernel-based discriminative learning algorithms for labeling sequences, trees, and graphs, Proceedings of the twenty-first international conference on Machine learning.

[42] K. M. Borgwardt, H.-P. Kriegel, Shortest-Path Kernels on Graphs, IEEE International Conference on Data Mining (2005) 74–81doi:http://doi.ieeecomputersociety.org/10.1109/ICDM.2005.132.

[43] F. Suard, A. Rakotomamonjy, A. Bensrhair, Object Categorization Using Kernels Combining Graphs and Histograms of Gradients, Image Analysis and Recognition (2006) 23–34.

[44] P. Mahé, N. Ueda, T. Akutsu, J.-L. Perret, J.-P. Vert, Extensions of marginalized graph kernels, In Proceedings of the Twenty-First International Conference on Machine Learning.

[45] P. Mahé, N. Ueda, T. Akutsu, J.-L. Perret, J.-P. Vert, Graph Kernels for Molecular Structure-Activity Relationship Analysis with Support Vector Machines, Journal of Chemical Information and Modeling 45 (4) (2005) 939–951.

[46] R. Raveaux, S. Adam, P. Héroux, E. Trupin, Learning graph prototypes for shape recognition, Computer Vision and Image Understanding 115 (7) (2011) 905–918. doi:10.1016/j.cviu.2010.12.015.
URL http://linkinghub.elsevier.com/retrieve/pii/S107731421100083X

[47] Y. Amit, G. August, D. Geman, Shape Quantization and Recognition with Randomized Trees, Neural Computation 9 (1996) 1545–1588.

[48] V. Lepetit, P. Lagger, P. Fua, Randomized trees for real-time keypoint recognition, in: Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on, Vol. 2, 2005, pp. 775 – 781 vol. 2. doi:10.1109/CVPR.2005.288.

[49] M. A. Fischler, R. A. Elschlager, The representation and matching of pictorial structures, IEEE Trans. Comput. 22 (1) (1973) 67–92. doi:10.1109/T-C.1973.223602.
URL http://dx.doi.org/10.1109/T-C.1973.223602

[50] P. F. Felzenszwalb, D. P. Huttenlocher, Pictorial structures for object recognition, Int. J. Comput. Vision 61 (1) (2005) 55–79. doi:10.1023/B:VISI.0000042934.15159.49.
URL http://dx.doi.org/10.1023/B:VISI.0000042934.15159.49

[51] P. Felzenszwalb, R. Girshick, D. McAllester, D. Ramanan, Object detection with discriminatively trained part-based models, Pattern Analysis and Machine Intelligence, IEEE Transactions on 32 (9) (2010) 1627–1645. doi:10.1109/TPAMI.2009.167.

[52] M. Burl, P. Perona, Recognition of planar object classes, in: In Proc. IEEE Comput. Soc. Conf. Comput. Vision and Pattern Recogn, 1996, pp. 223–230.

[53] R. Fergus, P. Perona, A. Zisserman, Object class recognition by unsupervised scale-invariant learning, in: In CVPR, 2003, pp. 264–271.

[54] N. Dalal, B. Triggs, Histograms of oriented gradients for human detection, in: In CVPR, 2005, pp. 886–893.

[55] M. Everingham, A. Zisserman, C. Williams, L. Van Gool, The Pascal Visual Object Classes Challenge 2006 (VOC2006) Results.
URL http://www.pascal-network.org/challenges/VOC/voc2006/results.pdf

[56] R. Nock, F. Nielsen, Statistical Region Merging, IEEE Transactions on Pattern Analysis and Machine Intelligence 26 (11) (2004) 1452–1458. doi:http://doi.ieeecomputersociety.org/10.1109/TPAMI.2004.110.

[57] HongJiang Zhang Atreyi Kankanhalli, S. W. Smoliar, Automatic partitioning of full-motion video, Multimedia Systems , 1:10{28}.

[58] J. F. B. J.F. Haddon, Co-occurrence matrices for image analysis, IEEE Electronics and Communications Engineering Journal vol. 5, (1993) No 2, pp. 71–83.

[59] K. S. R. M. Haralick, I. Dinstein, Textural features for image classification, IEEE Transactions on System, Man, Cybernetics vol. SMC-3 (1973) 610–621.

[60] Due, A. K. Jain, T. Taxt, Feature extraction methods for character recognition-A survey, Pattern Recognition 29 (4) (1996) 641–662. doi:10.1016/0031-3203(95)00118-2.
URL http://dx.doi.org/10.1016/0031-3203(95)00118-2

[61] Arun Hampapur Ramesh Jain, T. Weymouth, Production model based digital video segmentation, Journal of Multimedia Tools and Applications (1995) 1–38.

[62] Farshid Arman Arding Hsu, M. Y. Chiu, Image processing on compressed data for large video databases, ACM Multimedia Conference (1993) .267–272.

[63] G. J. F. J. K. S. J. M. G. Brown J. T. Foote, S. J. Young, Automatic content-based retrieval of broadcast news, ACM Multimedia Conference.

[64] V. Ogle, M. S. Chabot:, Retrieval from a relational database of images, IEEE Computer (1995) 40–48.

[65] Meastex, http:// www.cssip.elec.uq.edu.au /˜guy/ meastex / meastex.html.

[66] G. Smith, I. Burns, Measuring texture classification algorithms, Pattern Recognition Letters 18 (1997) 1495–1501.

[67] H. Hse, A. R. Newton, Sketched Symbol Recognition using Zernike Moments, Pattern Recognition, International Conference on 1 (2004) 367–370. doi:http://doi.ieeecomputersociety.org/10.1109/ICPR.2004.1334128.

[68] V. Levenshtein, Binary codes capable of correcting deletions, insertions and reversals, .Soviet Physics-Doklady 10 (1966) 707–710.

[69] F. M. J. Wagner R.A., The string-to-string correction problem, Journal of the ACM (1974) 168–173.

[70] S. R. S. D. Zhang K., On the editing distance between unordered labeled trees., Information Processing Letters 42 (1992) 133–139.

[71] K. Zhang, D. Shasha, Simple fast algorithms for the editing distance between trees and related problems., SIAM Journal on Computing 18(6) (1989) 1245–1262.

[72] Z. K. C. G. S. D. Wang J.T.L., Finding approximate patterns in undirected acyclic graphs, Pattern Recognition 35 (2002) 473–483.

[73] J. H. V. Nierman A., Evaluating structural similarity in XML documents, 5th Int. Workshop on the Web and Databases (2002) 61–66.

[74] K. P. N. K. B. B. Sebastian T.B., Recognition of shapes by editing shock graphs., 8th Int. Conf. on Computer Vision. 1 (2001) 755–762.

[75] K. Zhang, A Constrained Edit Distance Between Unordered Labeled Trees, Algorithmica 15 (1996) 205–222.

[76] H. Bunke, On a relation between graph edit distance and maximum common subgraph, Pattern Recognition Letters 18 (9) (1997) 689–694.

[77] M. Neuhaus, H. Bunke, Bridging the Gap Between Graph Edit Distance and Kernel Machines, World Scientific Publishing Co., Inc., River Edge, NJ, USA, 2007.

[78] R. Raveaux, J.-C. Burie, J.-M. Ogier, A graph matching method and a graph matching distance based on subgraph assignments, Pattern Recognition Letters In Press, (2009) –. doi:DOI: 10.1016/j.patrec.2009.10.011.
URL http://www.sciencedirect.com/science/article/B6V15-4XHVGWD-1/2/4bec72a05

[79] T. E. D. Implementation, http://web.science.mq.edu.au/˜swan/howtos/treedistance/.

[80] A. lightweight SIFT-implementation for Java after the paper of David Lowe (2004), http://fly.mpi-cbg.de/˜saalfeld/javasift.html.

[81] R. F. L. Fei-Fei, P. Perona, Learning generative visual models from few training examples: an incremental Bayesian approach tested on 101 object categories., IEEE. CVPR 2004, Workshop on Generative-Model Based Vision.

[82] S. K. N. S. A. Nene, H. Murase, Columbia Object Image Library (COIL-100), Technical Report CUCS-006-96.

[83] M. Aiello, A. M. W. Smeulders, Thick 2D relations for document understanding, Inf. Sci. Inf. Comput. Sci. 167 (1-4) (2004) 147–176. doi:http://dx.doi.org/10.1016/j.ins.2003.05.015.

[84] A. M. Finch, R. C. Wilson, E. R. Hancock, Matching delaunay graphs, Pattern Recognition 30 (1) (1997) 123–140. doi:DOI: 10.1016/S0031-3203(96)00060-X. URL http://www.sciencedirect.com/science/article/B6V14-3SNN1SV-C/2/5da571a46

[85] H. W. Kuhn, The Hungarian method for the assignment problem, Naval Research Logistic Quarterly 2 (1955) 83–97.

[86] J. Munkres, Algorithms for the Assignment and Transportation Problems, Journal of the Society of Industrial and Applied Mathematics 5 (1) (1957) 32–38.