Ph.D. in Electronic and Computer Engineering
Dept. of Electrical and Electronic Engineering
University of Cagliari

Interactive search techniques for content-based retrieval from archives of images

Luca Piras

Advisor: Prof. Giorgio Giacinto
Curriculum: ING-INF/05 SISTEMI DI ELABORAZIONE DELLE INFORMAZIONI

XXIII Cycle
March 2011

Ph.D. in Electronic and Computer Engineering
Dept. of Electrical and Electronic Engineering
University of Cagliari

Interactive search techniques for content-based retrieval from archives of images

Luca Piras

Advisor: Prof. Giorgio Giacinto
Curriculum: ING-INF/05 SISTEMI DI ELABORAZIONE DELLE INFORMAZIONI

XXIII Cycle
March 2011

A Iole ed alla mia famiglia
per il loro sostegno

Abstract

Through a little investigation by file types it is possible to easily find that one of the most popular search engines has in its indexes about 10 billion of images. Even considering that this data is probably an underestimate of the real number, however, immediately it gives us an idea of how the images are a key component in human communication. This so exorbitant number puts us in the face of the enormous difficulties encountered when one has to deal with them. Until now, the images have always been accompanied by textual data: description, tags, labels, ... which are used to retrieve them from the archives. However it is clear that their increase, occurred in recent years, does not allow this type cataloguing. Furthermore, for its own nature, a manual cataloguing is subjective, partial and without doubt subject to error. To overcome this situation in recent years it has gotten a footing a kind of search based on the intrinsic characteristics of images such as colors and shapes. This information is then converted into numerical vectors, and through their comparison it is possible to find images that have similar characteristics. It is clear that a search, on this level of representation of the images, is far from the user perception that of the images.

To allow the interaction between users and retrieval systems and improve the performance, it has been decided to involve the user in the search allowing to him to give a feedback of relevance of the images retrieved so far. In this the kind of image that are interesting for user can be learnt by the system and an improvement in the next iteration can be obtained. These techniques, although studied for many years, still present open issues. High dimensional feature spaces, lack of relevant training images, and feature spaces with low discriminative capability are just some of the problems encountered. In this thesis these problems will be faced by proposing some innovative solutions both to improve performance obtained by methods proposed in the literature, and to provide to retrieval systems greater generalization capability. Techniques of data fusion, both at the feature space level and at the level of different retrieval techniques, will be presented, showing that the former allow greater discriminative capability while the latter provide more robustness to the system. To overcome the lack of images of training it will be proposed a method to generate synthetic patterns allowing in this way a more balanced learning. Finally, new methods to measure similarity between images and to explore more efficiently the feature space will be proposed. The presented results show that the proposed approaches are indeed helpful in resolving some of the main problems in content based image retrieval.

Contents

List of Figures

Chapter 1

Introduction

The growing number of digital data such as text, video, audio, pictures or photos is creating the need to find ways to quickly and accurately retrieve information from that data. Whereas the results of traditional text data search methods are quite satisfactory, the same can not be said for visual or multimedia data.

The most so far common method for image retrieval is predicated on adding meta-data to the images as keywords, tag, label or short descriptions, so that retrieval can occur through such remarks. The manual cataloguing of the images, even though it requires expensive work and a large amount of time, often it is not so effective. Describing a picture in words is not always easy and the relevance of the description is strictly subjective. Figure 1.1, for example, could be considered an image depicting a *sunset* or the *sea* or even a *tower* so that it is impossible to give a unique description.

In addition it is necessary take into account that the use of a limited number of words does not always allow clearly describe what an image represents and that the same word can have several meanings. For example, performing a Google search using the word *"palm"*, the result could be very different from that expected (Figure 1.2). In fact, as you can see, only a picture out of twenty is a palm tree (honestly the first thing I thought when I did this research) and again only one out of twenty is the palm of a hand. To find more palm trees you need to scroll pages up to the third and for other palms of hands even up to the eighth. It is surely necessary to say that the Google searches are probably biased by economic factors or at least that its users seek more likely images of Palm Inc. products, however these small examples can help to understand the difficulties in describing an image. The multiplicity of the results of the search is due to the fact that manual indexing of images is independent of its content because it is not directly related to visual information, but only "connected" to them by author, subject depicted, the place of representation, etc.

To overcome these drawbacks, over the years several methods have been proposed. They are predicated on the idea of content based indexing by means the use of low-level features such as color, texture, shape, etc. and nowadays is a very active research field [94, 62, 24, 105]. Systems that make use of this approaches are called *content based image retrieval* (CBIR) systems.

Of course, a description of the images through this kind of features is far from the common perception that the user has of an image. For a human being indeed an image can represent many things: it can be fine or nasty, it can be good or bad, can evoke emotions and

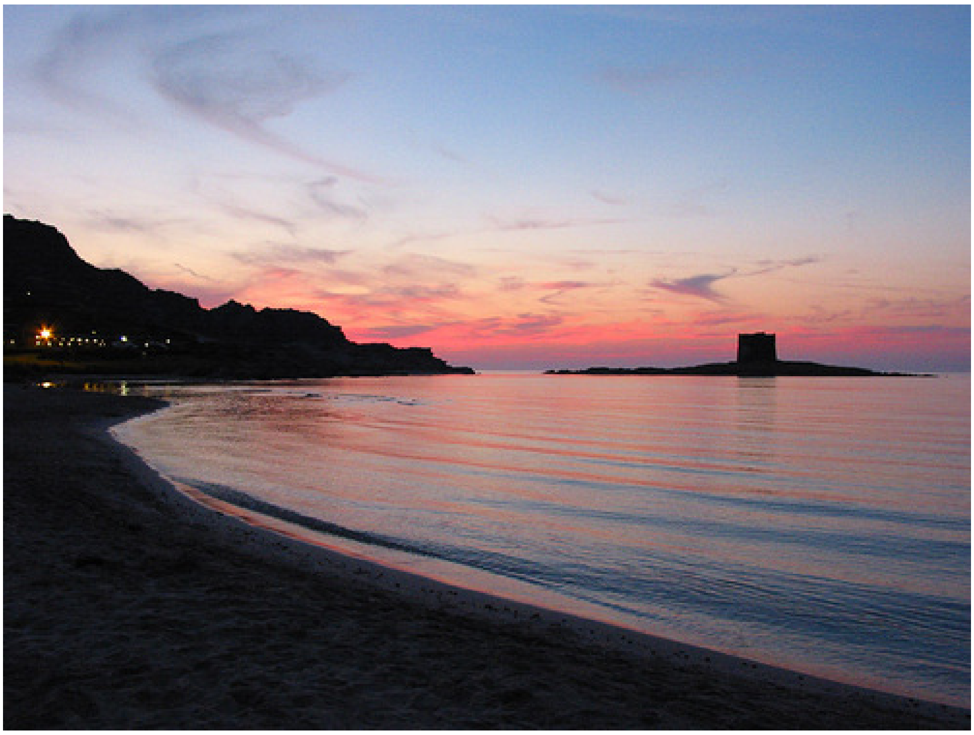
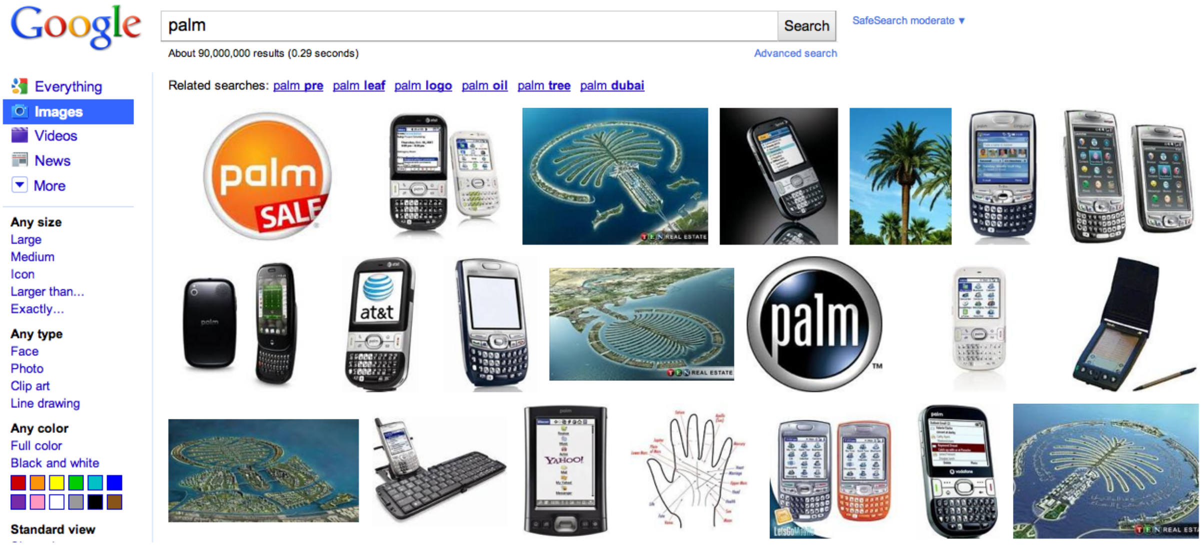


Figure 1.1: Beach *"La pelosa"* - Stintino (SS, Italy)



palm

About 90,000,000 results (0.29 seconds)

Related searches: palm pre palm leaf palm logo palm oil palm tree palm dubai

Figure 1.2: *"palm"* Google Images search

memories, and can also feel sensations that are not really represented (e.g., see Figure 1.3). For a computer is simply a set of pixels with different "color" and different intensities. In the last years there have been many attempts to bridge the gap between the high level features, those perceived by human beings which identify the semantic information related to the images, and the low level ones that are used in the searches. This difference in percep-

Figure 1.3: *“Cold”*

tion is widely known in the CBIR field as *semantic gap*. Among the proposed methods seem particularly effective retrieving images according to their similarity with respect to an image given as query (*Query by Example*) and to refine the search using feedbacks of the user that judges as *relevant* or *non relevant* (*Relevance Feedback*) the images that the system retrieves iteration by iteration.

Having said this, it might seem a now solved problem, however, there are still many obstacles to face. As already mentioned the pictures for a computer are sequences of numbers, so it is natural that low-level features are represented as vectors of numbers in a N-dimensional space. One of the first problems to address it is to make sure that relevance or semantic similarity can be measured and represented in this N-dimensional space. The problem is aggravated by the fact that, in general, the images that the user is interested in are just a small part of the totality of the images in the dataset and in addition the size of the space in which images are represented can be very large. In practice she is facing a search with a few items, in an very large area, populated mainly by objects in which she is not interested in. Fortunately some techniques from the field of *Pattern Recognition* and *classification* give us a helping hand, however, they carry with them some solutions as well as some weaknesses.

1.1 Research Overview

During the three-year period of the Ph.D. my research has been focused mainly on addressing some of the open issues concerning the relevance feedback and in general CBIR. In particular, they have been investigated *unbalanced learning, high dimensional feature space, low informative training set* and *low discriminative capability of the feature set* problems. In addition some innovative solutions have been developed to evaluate a similarity score between images.

unbalanced learning: Usually during the first few iterations the number of the images that the user considers non relevant is much larger than the number of those considered relevant. This is due mainly to two factors. First of all even if the user was “collaborative” the system can not submit to her an excessive number of images, say no more than a few dozen. Second, in very large databases the amount of images that interest the user, respect to the whole pictures, is usually very limited. Our main contribution in the *unbalanced learning* problem concerns a technique, called *K-Nearest Neighbors Directed Pattern Injection* (DPI K-NN), that consists of adding to the training set in the feature space some representations of "synthetic" images.

high dimensional feature space: In order to try to describe accurately the pictures, feature spaces are becoming larger and larger. This event, if in some ways increases the descriptive capability of feature vectors for other makes them difficult to handle. In addition, when sizes became very high often it happens to deal with sparse vectors, i.e. with a lot of components equal to zero, for some queries but not for others. This implies the need to find ways to reduce the feature spaces maintaining the pattern distribution or even better, trying to keep the relevant images close each other and far from the non relevant ones. On this topic, in this thesis, two techniques will be shown. The first is a preliminary work and consists in reconstructing all the pictures from an N-dimensional space in M-dimensional and P-dimensional spaces (with $M, P < N$) as a linear combination of relevant and non relevant images, respectively. It is possible to evaluate the relevance of each image by the error committed during in the reconstruction phase.

The second is based on attributing weights to the various components of the feature vectors. The weight evaluation that will be proposed is performed following two different approaches: the first is founded on the relative distance of images relevant and not relevant with respect to a certain component of the feature space, the second on the minimization of the classification error.

low informative training set: In the field of image retrieval usually, the images, which the user is interested in, belong to the same class but those that are not interesting for her may belong to many different classes. This behaviour sometimes makes the use of classifiers based on nearest neighbor not the most effective. Based on the concept that similar images are located in adjacent areas of space, this type of classifier retrieves relevant similar images but also non-relevant images similar to each other losing the generalization capability. In order to address this problem in this thesis it has been proposed a methodology that provides to combine the nearest neighbor paradigm (*Exploitation*) with a phase of exploration of the immediate neighbourhoods of the area occupied by the relevant images (*Exploration*). In this way it is possible to properly choose images that allow a greater ability to generalize instead of choosing only images that are the most similar to the examples.

low discriminative capability of the feature set: It is possible to extract from an image a huge quantity of low-level features: color moments, color and edge histogram, directivity and co-occurrence descriptors, etc. but probably the use of just one of them is not effective enough to achieve excellent performance. For this reason, very often it has been tried to combine information from each feature space in order to exploit the different characteristics that they are able to emphasize. The main problem is to find a way to increase the accuracy in the search but not the processing time. Another way to make the most of the features that have been extracted is not to combine the descriptors but different classifiers on the same set of features. Our contributions on this topic is twofold. One is the proposal of an approach that represents the images in a "(dis)similarities" space where each component is associated with a different feature space and that represents a similarity measure of the image with respect to the given query. In this way it is possible to obtain a space whose dimension is equal to just the number of spaces to be combined.

The second contribution has been the idea to combine two very different techniques for image retrieval in order to obtain good results in different datasets and representations rather than high performance in a specific dataset. The techniques that have been chosen are a nearest neighbor based approach and the SVM.

relevance evaluation: Finding a method which allows to transform the semantic similarity between two images into something measurable is a hoary open issue. Over the years several methods have been proposed that were more or less successful, depending on the context in which they were used. Under this perspective two new techniques to assess the similarity between images have been also proposed: one related on graphs and clustering, the other on on-line learning. In the first the image dataset is represented as an undirected weighted graph, where the weight of the edge reflects the similarity between pairs of vertices. In this graph it should be possible form a cluster of relevant images where the nodes are similar among themselves and at the same time dissimilar from the other nodes that do not belong to the cluster. Starting from the relevant images chosen by the user and following the concept of *Dominant Set* it is possible iteratively to modify the weight of the graph and to find new relevant images.

The other approach proposed on this topic is predicated on the *on-line learning*. More specifically the idea consists in exploiting the on-line nature of the relevance feedback in order to evaluate a classification function from a vector of coefficients. These are then computed iteratively by paying attention that they do not vary too quickly, staying in some way "linked" to the value that took in the previous iteration (*passive*) but at the same time change proportionally to the earlier committed classification error (*aggressive*).

1.2 Organization

The rest of this thesis is structured as follow.

Chapter 2 gives the required background on Content Based Image Retrieval, presents the widely used evaluation metrics and the most common databases.

Chapter 3 presents an overview of the state of the art of this thesis, showing the most popular techniques in the field of relevance feedback focusing particularly on those that have been proposed to address the open issues tackled in my research.

Chapter 4 discusses the solution of he main part of the problem introduced in the previous chapter that it has been proposed for retrieval techniques working in a single feature space. For each issue will be described the proposed solution followed by the result obtained during the experimental phase.

Chapter 5 focuses on two problem whose solutions are in some way related. Both problems will be faced separately. At the end some interesting result will be compared.

Chapter 6 introduces the purposes behind the idea to combine two different CBIR techniques and show why and how could be a "winning" idea in content based systems that work with very different databases.

Finally, Chapter 7 briefly summarizes the topics covered in this thesis and some future improvements that can extend our work.

Chapter 2

Content Based Image Retrieval

The men, ever since it came on the world, have always used images to communicate, to report or simply as a form of expression. Cave paintings of tens of thousands of years ago have even come down to us (Figure 2.1). As time went by men have maintained this way to communicate gradually finding more and easier ways to make these representations. The first photo dates back to almost two centuries ago (Figure 2.2[1]) and ever since the spread has been unstoppable. By now, everyone has a camera in own mobile phone and with the

Figure 2.1: Cro-Magnon, *Image of a horse* (Upper Paleolithic, 17,000 BC-15,000 BC, Lascaux caves, Limousin, France)

[1] *View from the window at Le Gras*, the first successful permanent photograph (WIKIPEDIA)

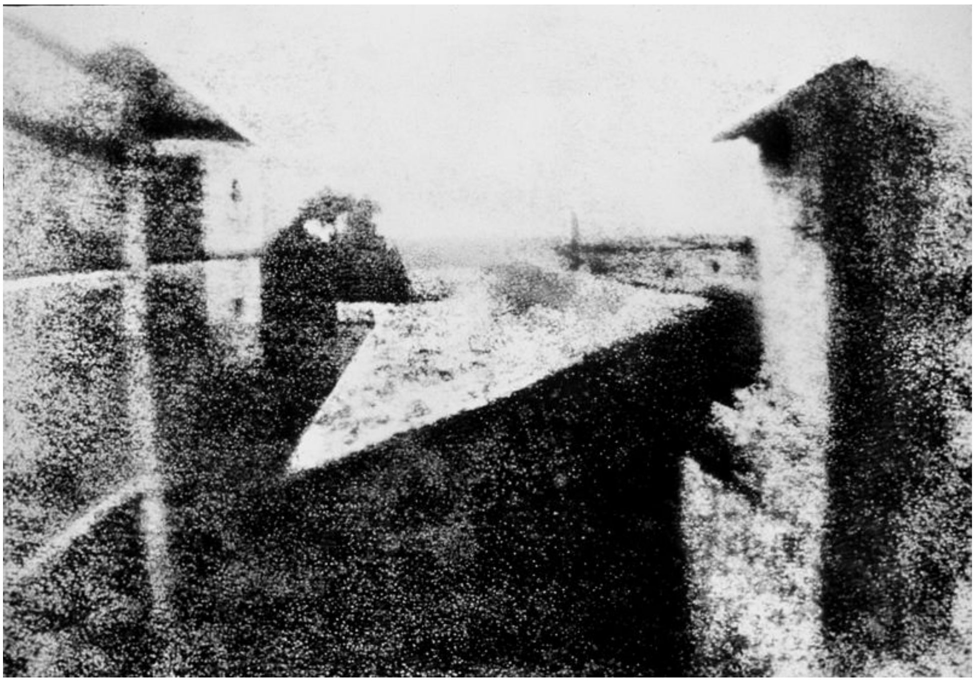


Figure 2.2: J. N. Niépce, *Vue de la fenêtre du domaine du Gras* (1826, Saint-Loup-de-Varennes, Bourgogne, France)

advent of Internet, social networks and storage space almost “unlimited”, the exchange of photos and digital images has become frenetic, to say the least. Faced with this new scenario it has become increasingly urgent need to find a way to manage this heap of data. One needs only to reflect on the amount of photos brought at home after a short vacation, if you do not hurry to save them on your computer, taking care to divide them at least by place and date, very soon by looking them again, “difficult” questions will arise: *Where we took this picture? What building is that?*

This is true, without taking care to also consider the persons represented in the images. In that case, the cataloguing of the photos will take more time than the holiday itself. It is therefore evident that a manual archiving of all the images that one possesses is to categorically rule out. Everyone has surely needed, for various reasons, to search an image on the web and has struggled with the inefficiencies of searches by keyword with the consequent wasting time. For this reason, since the early nineties, the scientific community has interested in the study of content based image retrieval [94].

In the rest of chapter it will be made a brief treatment on the different levels of meaning of an image and then it will be described the features that are extracted from the image to be processed by a retrieval system. In addition it will be described the way used to evaluate a rough but effective kind of similarity measures. Finally, the chapter will conclude with a description of the performance measures used in the rest of the thesis and the used dataset.

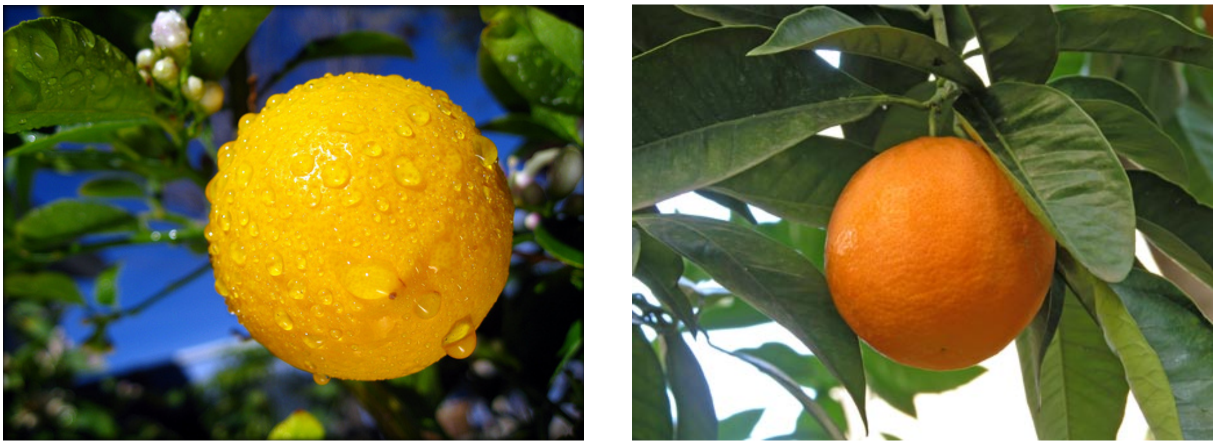


Figure 2.3: (Dis)similarity between features

2.1 What is the image content?

The first papers on CBIR are, by now, twenty years old [60] but we are still facing with problems not yet fully solved. The largest of these is that currently the only way to handle the "content" of an image is to extract what are called low-level features, i.e. to extract numerical values of the colors, shapes, edges of the images . This means that when two images are compared, the related "features" are compared while a human being compare the "concepts" expressed by them [101]. In the same light it is possible to distinguish between different "levels" [29, 38] of image content retrieval:

Level 1: it is the lowest level and is the one where the comparisons between the images can be made considering the intrinsic features of an image: find images with round objects, finds pictures with yellow objects, etc.. Of course, being the one needing less information has also the lowest effectiveness. For example, if you tried to evaluate the similarity between a orange and a lemon you might not always be satisfied by the result [61]. A retrieval system based on this level of content may respond with either a very high value of similarity or very low. If you look at the Figure 2.3 it is not difficult to see how a retrieval shape-based would consider the two images as similar, one based on color does not.

Level 2: it is characterized by the properties derived from objects in the picture. The queries can be the retrieval of images of specific objects (such as the Colosseum) or a specific type (like a bicycle). For searches at this level is needed a certain knowledge base, for example, that a particular building with a certain shape is known as the *Colosseum*.

Level 3: it is the highest level and requires a good deal of abstraction. At this level the search aims at concepts that go beyond what is physically represented. These may be particular events (ceremony of installation of the President of the United States) or feelings and emotions (cold, joy, etc.) and are definitely the hardest to find so much so that even a human being may be wrong when dealing with the identification of these concepts.

The most significant difference between levels is surely the one between the first and the second two (for which we speak of *semantic retrieval*). In fact it is not uncommon that two

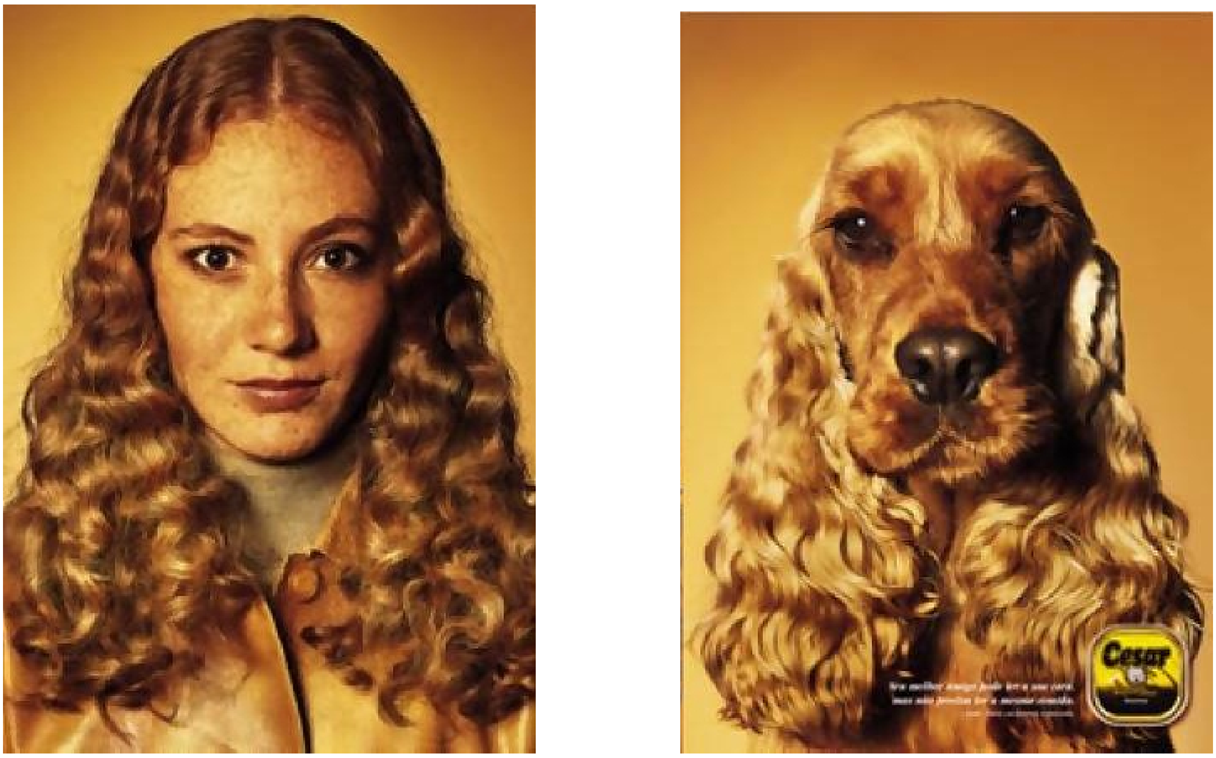


Figure 2.4: (Dis)similarity between concepts

images, even though similar in terms of colors and shapes, can represent completely different objects. An example of this is the Figure 2.4 [7] taken from a dog food advertisement of a few years ago, although the colors, the background and the shape of the object in the foreground are almost identical, the semantic difference is considerable. This difference of similarity perception is called *semantic gap*.

2.2 Image features

Before to provide a description of the features it is necessary to divide them into two categories, namely global and those local. Among the first ones those that are widely used are: color, texture and edges and are extracted from the image in its entirety. The local feature on the contrary, as the name suggests, are involved in the assessment of the most significant areas of the image [8, 25].

2.2.1 Color

The color is certainly one of the most important low-level feature for an image, first of all because marks it out strongly and it is robust with respect to translation and rotation. Secondly, a picture can be zoomed in or zoomed out without that its color distribution is significantly altered. Finally it is immediately perceived by human beings. It is therefore probably that a similarity between images perceived by the user corresponds to a similarity of color features. The human visual system is such as the retina is more sensitive to three particular wavelengths: Red, Green and Blue. This has led to the emerging of trichromatic model (RGB), in which all colors can be represented as a combination of the three primary ones in the right proportions, in the functioning of all the monitors and televisions. To maintain compatibility between the old black and white monitors and the color ones it has been preferred, however, encode the **RGB** information through the **YC_bC_r** signal, where $Y = (R+G+B)$, $C_b = (B-Y)$, and $C_r = (R-Y)$. Another model, closer to human perception of color, is the **HSV** model in which the three components represent the Hue, the level of color purity (Saturation) and the brightness value (Value).

A very simple description of the colors of an image can be done using a vector $\mathbf{H} = (h[1],\dots,h[i],\dots,h[N])$ of a vector space of dimension equal to the number N of colors in the image and where $h[i]$ is the percentage of pixels in the image of the i-th color. The graphical representation of the vector **H** is called ***color histogram*** [96]. However, the simplicity of representation of color histograms is accompanied by an information lack. In fact, histograms capture the global color distribution in an image but do not preserve spatial information., For example, a red figure on a blue background has the same histogram of one with the same number of pixels, but random arranged (see Figure 2.5). In order to overcome this drawback is used a "layout version" of the histograms: before to extract the color values, the images is split in sub-images and for each an histogram is evaluated. Some color histogram descriptors are included in the standard MPEG-7, which sets the color spaces that can be used to calculate them. The most common are the ***Scalable Color Descriptor*** and the ***Color Layout Descriptor*** [1] The main difference between the two descriptors are that the first is a color histogram in the **HSV** color space encoded by Haar Transform, the second is extract splitting the images in 8×8 sub-images, evaluating the the average color of each sub-image in the **YC_bC_r** color space and transforming them in coefficients by performing a Discrete Cosine Transform. In the following of the thesis this feature with the color histogram and color histogram layout [96] will be used as color descriptors.

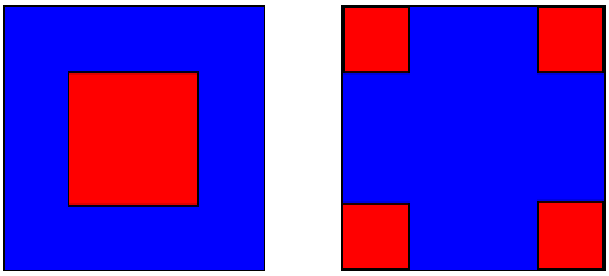


Figure 2.5: Indistinguishable figures for a color histogram

2.2.2 Texture

The texture identifies the recurrence of patterns with similar geometric properties. The techniques used for its description are typically statistical and spectral. Spectral methods are based on the Fourier spectrum analysis of the images in order to identify the recurrence that are present. These reveal themselves as peaks in the spectrum and carry with them a high energy content. The statistical methods are used instead to obtain information about the relative positions of the pixels. These are obtained by considering not only the intensity distribution but also the position of pixels with identical or very similar intensity values.

The best known texture feature set is that proposed by ***Tamura*** and that is called after him [97]. The set is composed by *coarseness, contrast, directionality, line-likeness, regularity*, and *roughness* and are designed according to human visual perception. Over the years the first three have been the widely used: the *coarseness* gives information about the size of the texture elements, the higher the coarseness value is, the rougher is the texture. The *contrast* is designed to evaluate changing a picture quality, not picture structure, considering the stretching or the shrinking of its grey scale. The grey level distribution influence also the sharpness of edges in the images, in fact a higher contrast imply sharper edges. The *directionality* measures the presence of orientation in the texture; does not matter the orientation of the patterns in the figures but, simply, the total degree of directionality; i.e., two patterns which differ only in orientation should have the same degree of directionality.

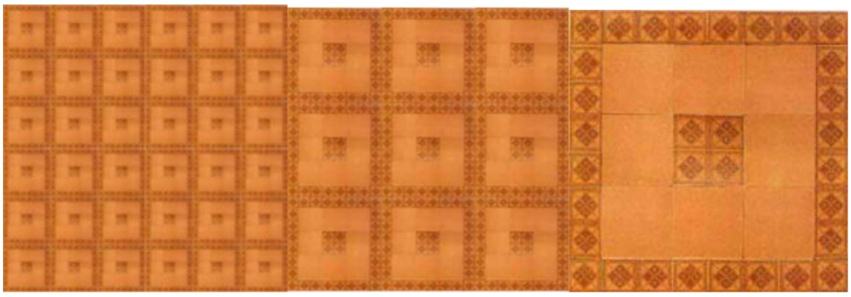


Figure 2.6: Examples of different scales textures

In different way works another texture feature, the ***co-occurrence texture matrix*** [39]. The entries of this matrix are evaluated counting how often in the image occurs a intensity variation between two adjacent pixels along a certain direction. This permits to taking in to account how much and following which orientation the edges change in the picture.

As for the color it is possible to extract a texture histogram, the ***Edge Histogram Descriptor*** [1] in fact, represents the spatial distribution of five types of edges: vertical, horizontal, 45°, 135°, and non-directional. The feature vector is obtained dividing the image into 16 (4 × 4) sub-images and generating a 5-bin histogram for each one.

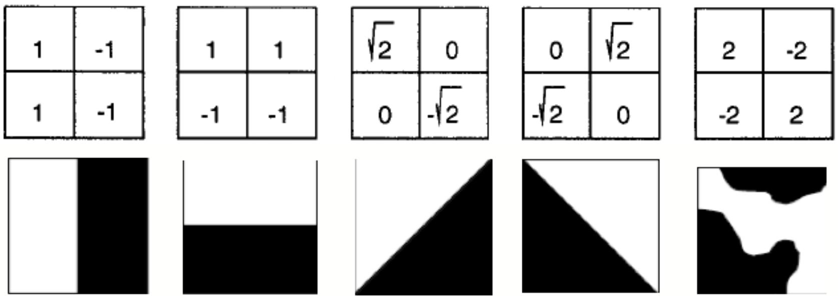


Figure 2.7: Different types of edges represented by the Edge Histogram Descriptor

In order to increase the discriminative capability of the feature it has been proposed to enclose in a unique vector different type of characteristics. In this light ***Color and Edge Directivity Descriptor*** [13] incorporates, as the name says, color and texture information in

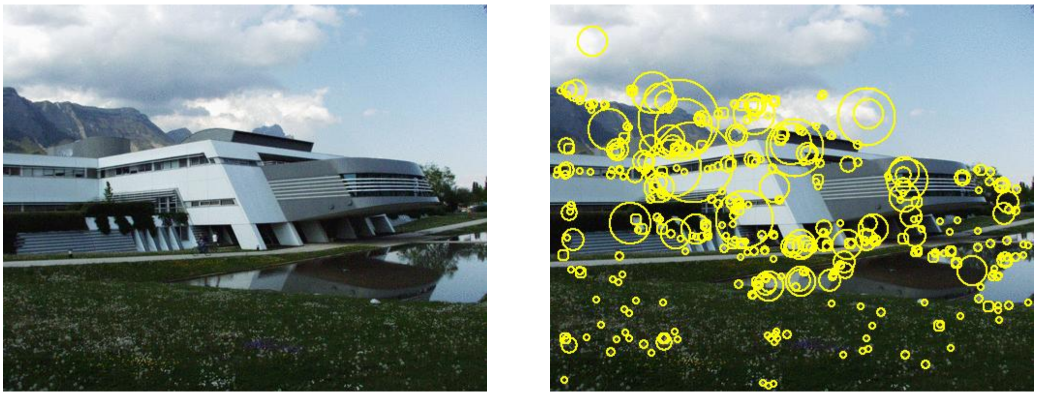


Figure 2.8: SIFT points

a histogram requiring low computational power for its extraction. The CEDD histogram is constituted by 6 regions from which the texture features are evaluated and each region is constituted by 24 individual regions from which the color feature are emanated.

2.2.3 Interest points

In the last few years interest points attracted increasing interest in the scientific community. With this term it is indicated a locations that, due to sudden changes in brightness or because they are on the edges of represented objects in the images, characterizes in a particular way the image [40]. The advantage of this feature is that it is invariant to changes of scale, rotation and translation. In fact, considering only the main points of the represented object it is negligible where they are located in the figure. For this reason from images that depict similar object it is possible to extract similar interest points. These interest points are then represented by means of numerical vectors. In the literature have been proposed many types of descriptors[67, 68], but that has emerged over the years is the *Scale-invariant Feature Transform* (**SIFT**) [65]. In the following the use of expression “local feature” it will be meant as SIFT descriptors extracted at Harris interest points (see Figure 2.8[2]) [27, 25].

2.3 Metrics

Over the years a large number of different similarity measures have been proposed by the researcher community. The choice of similarity measure depends on the chosen image descriptor, in fact some of them can be used with “standard” metrics some other requires particular measures suitably designed. In this section it will be discussed a selection of widely used “general-purpose” measures and some metrics designed for the feature described above [86].

A distances (or metric) in a space F is any function

$$\begin{array}{rcl} d : F \times F & \longrightarrow & \mathbb{R} \\ \mathbf{x} \times \mathbf{y} & \longmapsto & d(\mathbf{x}, \mathbf{y}) \end{array}$$

such as, $\forall$ **x**, **y**, **z** in F

[2] http://lear.inrialpes.fr/people/dorko/downloads.html

- $d(\mathbf{x},\mathbf{y}) \geq 0$,
- $d(\mathbf{x},\mathbf{y}) = 0 \iff \mathbf{I}_1 = \mathbf{I}_2$,
- $d(\mathbf{x},\mathbf{y}) = d(\mathbf{y},\mathbf{x})$,
- $d(\mathbf{x},\mathbf{y}) + d(\mathbf{y},\mathbf{z}) \geq d(\mathbf{x},\mathbf{z})$.

Given any two images and their representations in a N-dimensional feature space

$$\mathbf{x} = \left[x_1, \ldots, x_f, \ldots, x_N\right]; \\ \mathbf{y} = \left[y_1, \ldots, y_f, \ldots, y_N\right];$$

the most common metrics are:

- **Euclidean metric** (or L$_2$)

$$d(\mathbf{x},\mathbf{y}) = \sqrt{\sum_{f=1}^{N} (x_f - y_f)^2}; \tag{2.1}$$

- Weighted Euclidean metric

$$d(\mathbf{x},\mathbf{y}) = \sqrt{\sum_{f=1}^{N} w_f (x_f - y_f)^2} \tag{2.2}$$

  where w_f weights each component in different way according to its relevance. If w_f is equal to one the distance becomes as Eq. (2.1);

- Generalized Euclidean metric

$$d(\mathbf{x},\mathbf{y}) = \sqrt{(\mathbf{x}-\mathbf{y})^T\, W\, (\mathbf{x}-\mathbf{y})^2} \tag{2.3}$$

  where W is a $N \times N$ correlation matrix. If W is diagonal the distance becomes as Eq. (2.2), if it is the identity matrix as Eq. (2.1).

These metrics are usually used for *Tamura* and *Co-occurrence Matrix* descriptors

- **Manhattan metric** (or L$_1$)

$$d(\mathbf{x},\mathbf{y}) = \sum_{f=1}^{N} \left|x_f - y_f\right|; \tag{2.4}$$

- Weighted Manhattan metric

$$d(\mathbf{x},\mathbf{y}) = \sum_{f=1}^{N} w_f \left|x_f - y_f\right| \tag{2.5}$$

  where the weight w_f has the same function as in Eq. (2.1).

Manhattan metric is probably the widely used similarity measure, the distances for a lot of descriptors are evaluated according to it. In particular in this thesis will be used for *Color Histogram*, *Color Layout Histogram*, *Scalable Color*, and SIFT descriptors.

- **CEDD metric**
  For the measurement of the distance of CEDD between the images, Tanimoto coefficient [18] is used [13].

$$d(\mathbf{x},\mathbf{y}) = \frac{\mathbf{x}^T\mathbf{y}}{\mathbf{x}^T\mathbf{x}+\mathbf{y}^T\mathbf{y}-\mathbf{x}^T\mathbf{y}} \tag{2.6}$$

  In the case absolute congruence of the vectors, the distance takes the value 1, while in the maximum deviation it tends to zero.

- **Color Layout metric**
  The distance between two Color Layout descriptors is calculated as follows [1, 59]:

$$d(\mathbf{x},\mathbf{y}) = \sqrt{\sum_{i\in(Y)} w1_i\left(x_i-y_i\right)^2} + \sqrt{\sum_{i\in(C_b)} w2_i\left(x_i-y_i\right)^2} + \sqrt{\sum_{i\in(Cr)} w3_i\left(x_i-y_i\right)^2}; \tag{2.7}$$

  where Y, C_b, C_r denote the set of elements of the Color Layout vector that are associated with the **Y**, $\mathbf{C}_\mathrm{b}$, and $\mathbf{C}_\mathrm{r}$ components, respectively, and $w1_i$, $w2_i$, and $w3_i$ are the weights for the i-th vector's component.

- **Edge Histogram metric**
  Because the 80 bins may not be sufficient to yield efficient image matching, in order to evaluate a similarity measure they are generated an additional global edge histogram and some semi-global edge histograms directly from the original 80 bins. The global edge histogram has 5 bins and each bin value is obtained by accumulating the bin values of the corresponding 5 edge types. Similarly, for the semi-global edge histograms, it is possible to group in some subsets of the original histogram. They are defined 13 different subsets of the image and for each subsets they are extracted the edge distributions for the five different edge types from the 80 histogram bins (see Figure 2.9). Consequently, there are a total of 150 bins (80 bins (original) + 5 bins (global) + 65 bins (13×5, semi-global)) for the similarity matching. The distance between two Edge Histogram descriptors is calculated as follows [1, 115]:

$$d(\mathbf{x},\mathbf{y}) = \sum_{i=0}^{79}\left|x_i^{local}-y_i^{local}\right| + 5\times\sum_{i=0}^{4}\left|x_i^{global}-y_i^{global}\right| + \sum_{i=0}^{64}\left|x_i^{semi\text{-}global}-y_i^{semi\text{-}global}\right| \tag{2.8}$$

  where x_i^{local} represents the i-th component of the original vector of the image **x**, x_i^{global} represents the i-th component of the global Edge Histogram for the same image, and finally $x_i^{semi\text{-}global}$ is the i-th bin value for the semi-global edge histograms of image **x**. Because the number of bins of the global histogram is relatively smaller with respect to the local and semi-global a weighting factor of 5 is applied.

2.4 Evaluation

To evaluate CBIR, several performance evaluation measures have been proposed [69]. The widely used are surely the *precision* and the *recall*.

Precision

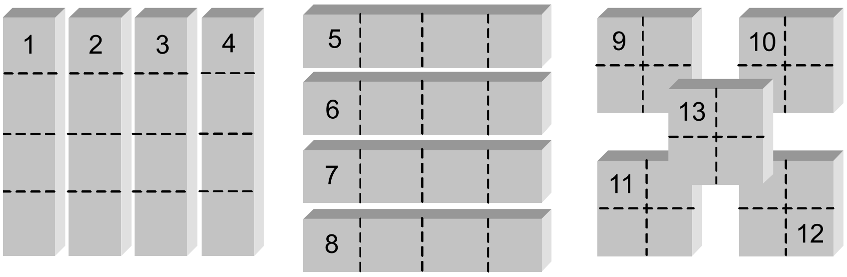


Figure 2.9: Subsets for semi-global histograms

The retrieval precision p_q for a given query q is the ratio between the number of relevant images found by the system at a given iteration i and the total number of images retrieved in the same iteration.

$$p_q = \frac{\textit{Number of relevant images retrieved}}{\textit{Total number of images retrieved}} \tag{2.9}$$

As you can see from Eq. (2.9), P_q measures, the capability of correct retrieval of the system in a single iteration. The mean percentage of the precision on the total number Q of queries is:

$$p_{(\%)} = \frac{1}{Q}\sum_{q=1}^{Q} p_q \cdot 100. \tag{2.10}$$

Recall
The recall r_q is instead defined as the ratio between the number of relevant images retrieved up to a certain iteration i for a query q from the system and the number of relevant images for the same query that are in the database.

$$r_q = \frac{\textit{Number of relevant images retrieved}}{\textit{Total number of relevant images}} \tag{2.11}$$

Through r is possible to evaluate the system's ability to retrieve images relevant to each new iteration. It is also possible a variation in recall measure obtained changing the denominator of Eq. (2.11) with the minimum between the *number of relevant images in the dataset*, and the *total number of images evaluated by the user*. In this way it is possible to take into account the maximum number of relevant images that can be actually retrieved. The mean percentage of recall on the total number of queries is:

$$r_{(\%)} = \frac{1}{Q}\sum_{q=1}^{Q} r_q \cdot 100. \tag{2.12}$$

F-measure
It is also possible to combine the precision and the recall in a unique performance measure: the *F-measure* [51]

$$F = 2 \cdot \frac{p \cdot r}{p + r}. \tag{2.13}$$

Average precision

The average precision for some aspects summarize the precision and the recall, in fact given a query q is the mean over the precision after each retrieved relevant image.

$$AP_q = \frac{1}{N_R} \sum_{i=1}^{N_R} p(i) \tag{2.14}$$

where N_R is the total number of relevant images for the query q. The mean average precision (MAP) is the mean of the average precision over all queries:

$$MAP = \frac{1}{Q} \sum_{q=1}^{Q} AP_q. \tag{2.15}$$

2.5 Datasets

The technique that will be shown in the following chapter has been tested using several dataset; in the next subsection they will be presented.

2.5.1 Corel *SMALL*

It is a subset of the Corel dataset obtained from the UCI KDD repository[3]. It consists of 19511 images that have been manually subdivided into 42 semantic classes with a large variability in the size: from 96 to 1544 (see Figure 2.10). Even if the Corel dataset is never used in its entirety, different subsets of the complete stock are very common in the scientific community.

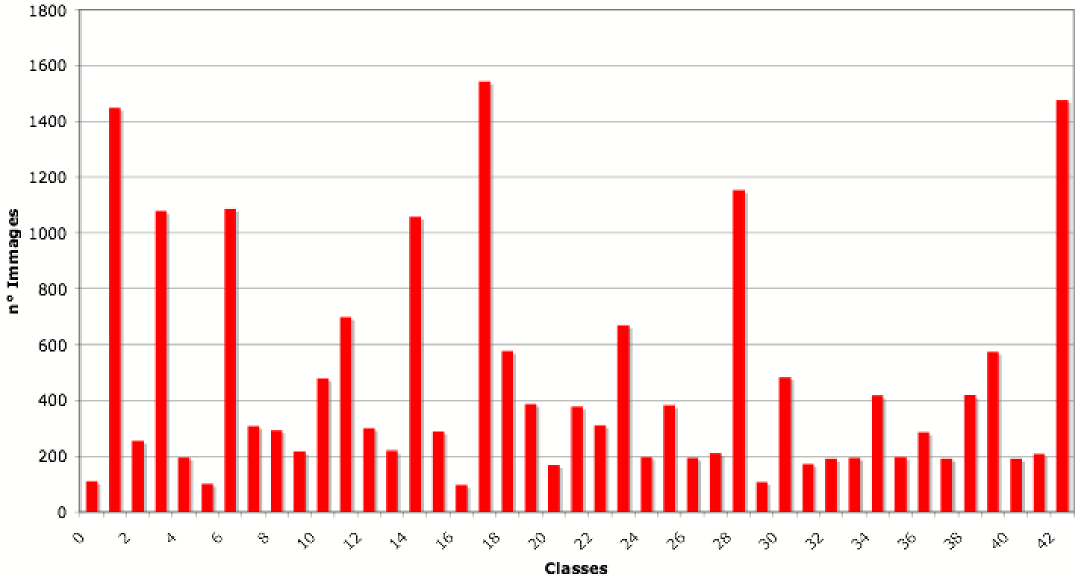


Figure 2.10: Images per class in Corel *SMALL* datset

[3] http://kdd.ics.uci.edu/databases/CorelFeatures/CorelFeatures.html

2.5.2 WANG dataset

WANG dataset[4] is another subset of Corel and consists of a subset of 1000 images of the Corel stock photo database which have been manually selected and which form 10 classes of 100 images each [112] (see Figure 2.11). The WANG database is usually considered as a easy task in the Image Retrieval context.

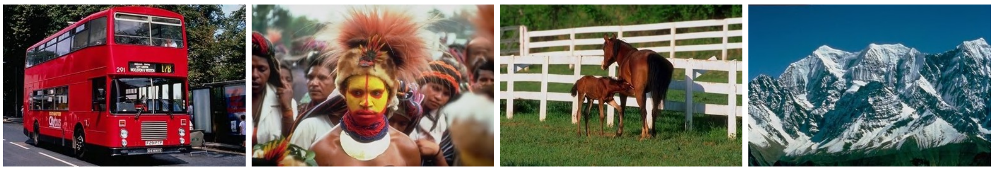


Figure 2.11: Example images from WANG dataset

2.5.3 MSRC dataset

The Microsoft Research Cambridge Object Recognition Image database[5] (in the following referred to as MSRC) contains 3007 images subdivided into 17 “main” classes, each of which is further subdivided into subclasses, for a total of 33 semantic classes [114]. It is mainly used in object recognition, in the pictures there is usually one object “captured” from a particular point of view (front, side, rear, etc.) or at most more object of the same type. An example of object from different classes coming from two of the “main” classes (car and flower) is given in Figure 2.12.

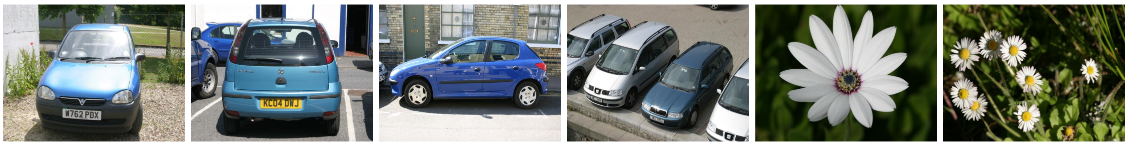


Figure 2.12: Example images from MSRC dataset

2.5.4 Caltech-256 dataset

The Caltech-256 dataset, from the California Institute of Technology[6] consists of 30607 images subdivided into 257 semantic classes [37] as MSRC is widely used in object recognition and it is considered a difficult task. In Figure 2.13 it is possible to see some examples of difficult classes. In the first row the picture belong to the *cactus* class, in the second to the *birdbath* one. It is possible to see how much, in the same class, the images differ each other.

[4] http://wang.ist.psu.edu/docs/related.shtml

[5] http://research.microsoft.com/downloads

[6] http://www.vision.caltech.edu/Image_Datasets/Caltech256/

Figure 2.13: Example images from Caltech-256 dataset

2.5.5 MIRflickr-25000 dataset

The MIRflickr-25000 collection [52] consists of 25000 images downloaded from the social photography site Flickr through its public API. It is a multi tagged dataset with a average number of tags per image of 8.94. All images include the tags that the original photographer has assigned to them and in the collection there are a total of 1386 tags which occur in at least 20 images. Moreover for the images also some manual annotations are available. MIRflickr has been proposed only few years ago but is enjoying great success by the scientific community. Recently it has been proposed an updated dataset with 1,000,000 of images [53].

Chapter 3

Semantic gap and other open issues

As the years go by, it is ever-easier to have access to a ever-greater amount of electronically archived images. As a consequence there is an increasing need for tools enabling the semantic search, classification, and retrieval of images. As above-mentioned the use of meta-data associated to the images solves the problems only partly, as the process of assigning meta data to images is not trivial, is slow, and closely related to the persons who performed the task. This is especially true for retrieval tasks in very high dimensional databases, where images exhibit high variability in semantic. It turns out that the description of image content tends to be intrinsically subjective and partial, and the search for images based on keywords may fit users' needs only partially. The main reason for the difficulty in devising effective image retrieval and classification tools is caused by the vast amount of information conveyed by images, and the related subjectivity of the criteria to be used to assign labels to images [94, 62, 24, 105].

This kind of problem is called *semantic gap* and it is precisely due to the different ways in which human beings and machines interpret the images. For the humans, these arouse emotions, memories or also reflections; for a computer are simple sets of pixels from which to extract numerical values. In order to capture such subjectivity, image retrieval tools may employ the so called *relevance feedback* [82, 121]. In the relevance feedback techniques the user is involved in the process of refining the search. In a CBIR task in which the RF is applied (see Figure 3.1), the users submit to the system a query image, that is an example of the pictures of interest (Fig. 3.1 **(1)**); the system starting from the query give to the images in the database a score related to a similarity measure between the images and the query. A certain number of best scored images are returned to the users (Fig. 3.1 **(2)**) that label them as relevant or not (Fig. 3.1 **(3)**) and that re-submit these images as new "query" (Fig. 3.1 **(4)**). With this new information the system can improve the search and give a more accurate result in the next iteration.

The main distinctions can be made between different relevance feedback approaches concern the degree of relevance that the user can express and type of learning system: *long* or *short-term*, and *active* or *passive*. Regarding the type of judgement that the user can express on an image it can be expressed either by assigning a score, i.e. expressing "how much" a certain image is similar to the one she desire, or by a definite judgement of relevance or non

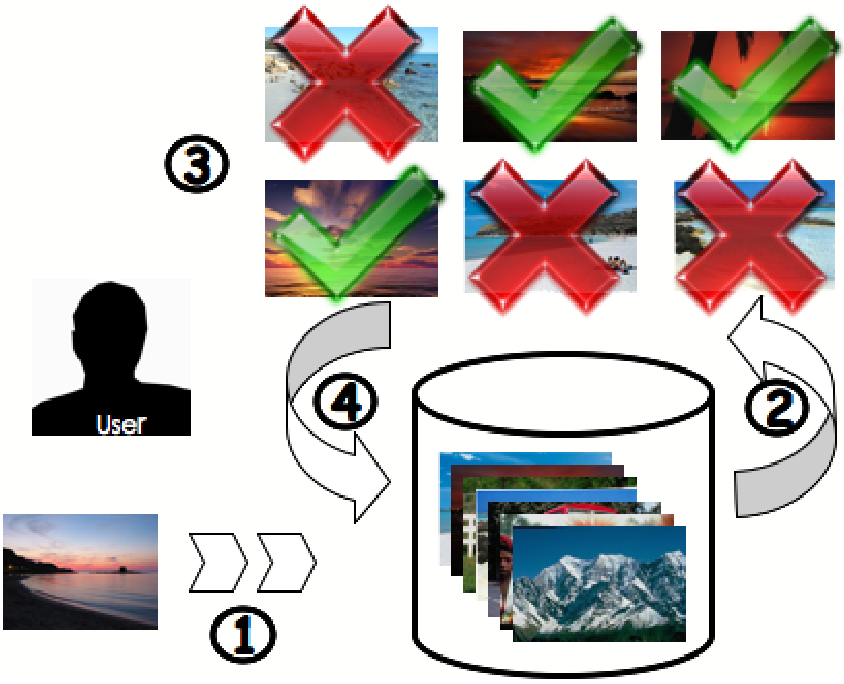


Figure 3.1: Relevance Feedback steps: **(1)** Query by example; **(2)** Returned best similarity scored images; **(3)** User's judgement; **(4)** New query examples

relevance. Even if with the latter type of choice all the nuances that the human is able to capture in the similarity evaluation are inevitably lost, it is the most widely used method. In fact, it is that allows a reduced complexity in the retrieval systems and requires less work for the user.

The type of learning, as already mentioned, can be either *passive* or *active* type. In the first case, the system returns to the user the images that "judges" the most relevant, in the second case, relying on the cooperation of the user, the system returns those on which has greater uncertainty and therefore that, once assessed by user, get more information. This second type usually obtains better results but requires a greater effort of the user who almost never is willing to repeat the evaluation process for more than four or five times. In the following it will be considered mainly the passive learning. Active learning techniques will be discussed in Section 3.5.1.

A further distinction in the type of learning is between *short* and *long-term*. The first involves that the system, to show to the user the most significant images, takes into account only the RF iterations just made by that user. The second one takes into account all the queries of the same type carried out in the past by different users. This second type of learning, in spite of the potential for better performance, has the disadvantage that, since each user expresses its judgement in a subjective manner, not always, the opinions given by a user may then bring a real "help" in a search of another one.

The core of the retrieval systems, obviously, is the algorithm that learns which images in the database the user is interested in by analysing the query image and any feedback. Some methods assign directly a relevance score to each image in the database [31], others find an hyperplane that separates the relevant images from the non relevant ones [118], and others map the images as a weighted graph [56] from which extracting a cluster of relevant images. The methods also can differ in the way they approach the problem of classification (broadly speaking) of images. In fact from the most common *two-class* approach, where a model is built that either classifies an image as positive or as negative, it has been also proposed *one-class* approaches, where the model is built only for the relevant class [99]. These techniques are well known in pattern recognition field, but are not always very suitable in application for content based image retrieval. Starting from the consideration that *"all positive examples are alike; each negative example is negative in its own way"* [120] a *(1+x)-class* approach has been proposed.

In the next section will be shown a brief review of some techniques of relevance feedback at the state of the art as *query shifting*, *support vector machine*, *(dis)similarity spaces* and *nearest neighbor* approach. In the following some of the most common problems currently investigated by the researchers and faced in this thesis will be described. In particular, *unbalanced learning*, *high dimensional feature space*, the *low capability to evaluate similarity between all relevant images*, *low informative training set* and the *low discriminative capability of the feature set* problems have been investigated. In addition will be shown some solutions proposed to resolve them by the scientific community and some our innovative proposals.

3.1 Relevance Feedback techniques

3.1.1 Query Shifting Techniques

As mentioned in the Chapter 2 the similarity measure can be expressed by means of distances in the feature space, a family of techniques to improve the performance have adopted the so called query shifting paradigm. A simple approach to finding more relevant images from the user's feedback is based on a well known technique in *Information Retrieval*: the Rocchio's formula [80]. This method, initially developed for the retrieval of text documents has been then adapted to image retrieval techniques [83]. This technique is based on the concept that the relevant images and the non relevant ones should form, in the feature space, distinct group. Then moving the queries in these areas more densely populated by relevant images should be easier to retrieve them. The approach is based on the minimization of a function that approximated the relevance of all the images of the database. Given a query $\mathbf{q}$, the new query $\mathbf{q}'$ is evaluated as

$$\mathbf{q}' = \alpha \cdot \mathbf{q} + \beta \cdot \left(\frac{1}{r} \sum_{\mathbf{x}_i \in R} \mathbf{x}_i \right) - \gamma \cdot \left(\frac{1}{n} \sum_{\mathbf{x}_j \in N} \mathbf{x}_j \right) \tag{3.1}$$

where $\mathbf{x}_i$ and $\mathbf{x}_j$ represent the images in the set of the relevant (R) and non relevant images (N), respectively, that have cardinality equal to r and n, respectively. α, β, and γ are parameter to fix.

In [33] the authors proposed a modification of the original approach, they move the query according to the position of the mean point of the relevant and non relevant images. Let $\mathbf{x}$ and $\mathbf{q}$ be two feature vectors representing an image in a d-dimensional feature space and the initial query provided by the user to perform the k-NN search, respectively. Let $N^r(\mathbf{q})$ and $N^{nr}(\mathbf{q})$ be the sets of relevant and non relevant images, respectively, contained in the neighbourhood of $\mathbf{q}$ $N(\mathbf{q})$. The mean vectors of relevant and non relevant images of $N(\mathbf{q})$, $\mathbf{m}_r$ and $\mathbf{m}_n$, can be computed as follows:

$$\mathbf{m}_r = \frac{1}{r} \sum_{\mathbf{x} \in N^r(\mathbf{q})} \mathbf{x}, \quad \mathbf{m}_n = \frac{1}{n} \sum_{\mathbf{x} \in N^{nr}(\mathbf{q})} \mathbf{x} \tag{3.2}$$

where r and n are the sizes of relevant and non relevant image sets, respectively $(r + n = k)$. The two normal distributions of relevant and non-relevant images are characterized with means $\mathbf{m}_r$ and $\mathbf{m}_n$, respectively, and equal variance σ^2 computed as follows:

$$\sigma^2 = s_W \cdot s_B \tag{3.3}$$

where

$$\begin{aligned} s_W^2 = {} & \frac{1}{k}\left(\frac{r}{r-1} \sum_{\mathbf{x} \in N^r(\mathbf{q})} (\mathbf{x} - \mathbf{m}_r)^T (\mathbf{x} - \mathbf{m}_r) + \right. \\ & \left. + \frac{n}{n-1} \sum_{\mathbf{x} \in N^{nr}(\mathbf{q})} (\mathbf{x} - \mathbf{m}_n)^T (\mathbf{x} - \mathbf{m}_n)\right) \end{aligned} \tag{3.4}$$

$$s_B^2 = (\mathbf{m}_r - \mathbf{m}_n)^T (\mathbf{m}_r - \mathbf{m}_n) \tag{3.5}$$

In other words, σ is computed as the geometric mean of the within-scatter s_W and the between-scatter s_B. According to [33] the new "shifted" query can be evaluated as

$$\mathbf{q}_{BQS} = \mathbf{m}_r + \frac{\sigma^2}{\|\mathbf{m}_r - \mathbf{m}_n\|}\left(1 - \frac{r-n}{\max(r,n)}\right)(\mathbf{m}_r - \mathbf{m}_n) \tag{3.6}$$

In the following, between the different versions of the query shifting approaches that it has been proposed in the literature, it will be used the BQS, as proposed in our research group.

3.1.2 Support Vector Machines

Differently from the previous methods that are essentially based on the density estimation there is another family of relevance feedback techniques that is based on discriminative training, i.e. methods that learn from a set of positive and negative labelled images how to classify the unlabelled ones. In this family, surely have a leading role the **Support Vector Machines** (SVM) [111, 23]. The idea behind the SVM is to find a hyperplane, in the feature space, that divides it into two subspaces. The first populated by positive samples, the second one by negative samples. The patterns closest to the hyperplane are called *support vectors*. The greater is the distance (margin) between these vectors and the hyperplane the greater is the reliability of the classifier. Given a set of linear separable training samples $\mathbf{x}_i \in \mathbb{R}^n$, the general form of linear classification function is

$$f(\mathbf{x}) = \mathbf{w} \cdot \mathbf{x} + b \tag{3.7}$$

which corresponds to a separating hyperplane $\mathbf{w} \cdot \mathbf{x} + b = 0$. Solving a constrained quadratic programming problem it is possible to find the vector $\mathbf{w}$ such that maximize $\frac{1}{\|\mathbf{w}\|}$ that is the distance from the closest point to the hyperplane. The solution $\mathbf{w}$ has an expansion $\sum_i \alpha_i \mathbf{x}_i$

in terms of a subset of patterns that are the nearest to the hyperplane and are called *support vector*. The function in the Eq. (3.7) can be now written as

$$f(\mathbf{x}) = \sum_i \alpha_i \mathbf{x}_i \cdot \mathbf{x} + b \tag{3.8}$$

the sign of this function permits to classify the pattern $\mathbf{x}$. In its "plain" form the SVM is a linear classifier but can be used also in non linear problem using the so called *kernel trick*. The kernel trick permits to transform the feature space to a higher dimensional space through the function $\phi(\cdot)$ [23] and it is so possible to find a hyperplane that separates the patterns [87]. The dot product in Eq. (3.8) can be represented as

$$\phi(\mathbf{x}_i) \cdot \phi(\mathbf{x}) = K(\mathbf{x}_i, \mathbf{x}) \,. \tag{3.9}$$

Finally, the classification function can be now written as

$$f(\mathbf{x}) = \sum_i \alpha_i K(\mathbf{x}_i, \mathbf{x}) + b \tag{3.10}$$

The SVM has been widely used in image retrieval [118] also in *one-class* problems [16]. In order to address its low performance in case of asymmetric training set it has been also proposed a solution where several classifiers have been learnt with a bag of balanced number of positive and negative example[98].

3.1.3 Nearest Neighbor Approach

Other approaches estimate the relevance of an image according to the relevant and non-relevant images in its neighborhood. The use of the nearest neighbor paradigm is motivated by its use in a number of different pattern recognition fields, where it is difficult to produce a high-level generalization of a class of objects, but where neighborhood information is available [3, 28]. In [31] a score is assigned to each image of a database according to its distance from the nearest image belonging to the target class, and the distance from the nearest image belonging to a different class. This score is further combined to a score related to the distance of the image from the region of relevant images. The combined score is computed as follows:

$$rel(\mathbf{x})_{stab} = \left(\frac{n/t}{1+n/t}\right) \cdot rel_{BQS}(\mathbf{x}) + \left(\frac{1}{1+n/t}\right) \cdot rel_{NN}(\mathbf{x}) \tag{3.11}$$

where n and t are the number of non-relevant images and the whole number of images retrieved after the latter iteration, respectively. The two terms rel_{NN} and rel_{BQS} are computed as follows:

$$rel_{NN}(\mathbf{x}) = \frac{\|\mathbf{x} - NN^{nr}(\mathbf{x})\|}{\|\mathbf{x} - NN^{r}(\mathbf{x})\| + \|\mathbf{x} - NN^{nr}(\mathbf{x})\|} \tag{3.12}$$

where $NN^r(\mathbf{x})$ and $NN^{nr}(\mathbf{x})$ denote the relevant and the non relevant nearest neighbor of $\mathbf{x}$, respectively, and $\|\cdot\|$ is the metric defined in the feature space at hand,

$$rel_{BQS}(\mathbf{x}) = \frac{1 - e^{1 - d_{BQS}(\mathbf{x}) \Big/ \max_i d_{BQS}(\mathbf{x}_i)}}{1 - e} \tag{3.13}$$

where e is the *Euler's number*, i is the index of all images in the database and d_{BQS} is the distance of image $\mathbf{x}$ from a modified query vector computed according to the Bayes decision theory (Bayes Query Shifting, BQS) [33] (see Section 3.1.1).

3.1.4 Some other techniques

Strictly related to the SVM is the *on-line* binary classification problem, where as classification function is used the a vector of weight [9]. **On-line learning** is particularly suitable for facing RF problems due to the on-line nature of such a problem. Moreover with the on-line approach it is possible to obtain a very fast and compact RF method and this property is very important in a real CBIR scenario. In Section 4.2 will be shown a new application of this technique proposed in relevance feedback for content based image retrieval field.

Despite its wide use, achieving high performance by SVM can be not easy, in fact it depends on the tuning of a number of parameters that often have been set after a long series of tests. Rather than splitting the feature space into subspaces populated only by relevant / non relevant images or find out the distributions of the two images' "classes", sometimes what it is required by a retrieval system is to provide a list of images in order of relevance (passive learning) or in order of informative capacity (active learning), i.e. according to the importance assigned them by the user for that specific kind of query [100]. In order to evaluate the importance it is necessary knowing what the user is looking for and exploit her feedback to find other "similar" images.

What does "similar" mean? This is a question that only the user can answer and for this reason over the years a lot of similarity measure have been proposed. As mentioned in Section 2.3 the easiest way to evaluate the similarity is introducing a metric in the feature space [86]. The pair-wise distances between patterns have been used in pattern recognition field also to build a **(dis)similarity space** [74] where the distances between patterns of the same class are smaller than those of patterns of different classes. A similar approach has been proposed [32] and used [70] to exploit relevance feedback in in content based image retrieval field. Different techniques use the user feedback in order to find a distance metric such that the distance in low-level features is consistent with the users' relevance judgements [92]. This metric should try to minimize the distance between similar images and meanwhile maximize the distance between the feature vectors of dissimilar images.

3.2 Unbalanced learning

3.2.1 Artificial patterns generation

One of the most severe problem in the design of the classifier, is the imbalance between the number of samples of the class the user is interested in, and all other images of the database that share some characteristics with that class. In the machine-learning literature the imbalance problem has been widely investigated and solution based either on under-sampling the majority class, or on over-sampling the minority class have been proposed [41]. However, it is easy to see that these solutions may produce a distortion of the "real" distribution of the classes. In the image retrieval domain, some solutions proposed so far involve the reduction of the set of images that are non-relevant to user's interest by the creation of bootstrap samples from the set of non-relevant images [98]. An ensemble of balanced training sets is thus created, each ensemble being made up of all the available relevant images and one of the bootstrap samples. This solution is computationally quite expensive, as an ensemble of classifiers has to be created, and the choice of the most appropriate set of parameters for the various parts of the systems is far from being a trivial task. Recently, some papers addressed the imbalance problem by proposing the generation of artificial patterns as in

[107, 106] where the authors propose the generation of synthetic images in the *Karhunen-Loeve Transform* feature space (KLT, e.g. [104]). The authors synthesized the images by the standard method of linear reconstruction using the eigenvectors of the KLT representation and extracted the low level features of this synthesized images considering the images as any regular image and applying a common feature extractor. Through these "new" images the system receives more informative feedback from the user. Others interesting solutions have been proposed over the years, in [2] the authors suggest to obtain a object-level view of the query image using image segmentation and permit to the user to selects segment(s) of interest. After this first step a set of modified images is automatically generated by the system and the initial user perception is learned based on the user feedback on the set of modifications. In [58] it has been proposed to use the segmentation and assembling all the segmented regions of positive examples together and resizing the regions to emphasize the latest positive examples, formed a composite image as a new query. In [50] instead the authors increase the number of relevant and non-relevant samples using a two-stage Bayesian classifier. In the first step the system eliminate the vast majority of negative example and submit to the user those that have passed the first selection. When the user labels an image as negative, all sub-images within that image that passed the first-stage classifier are considered as "new" negative examples for the second-stage classifier, and any positive regions identified by the user are used as positive examples for both classifiers. In spite of the previous methods that propose the generation of synthetic images for the sake of receiving informative feedback from the user, in [14] it has been proposed a technique (SMOTE) that addresses the imbalance problem by creating new random artificial patterns of the class of interest in the feature space to improve the performances of learning mechanisms. In Section 4.1 a similar approach will be proposed, where the creation of new patterns is made by exploiting k-NN relations and that improve the performance obtained by SMOTE.

3.3 High-dimensional feature spaces

3.3.1 Local manifold

Strictly related to the small sample problem is the so called *curse of dimensionality* due to the now ever more common use of high-dimensional feature spaces [10]. In these spaces the lack of relevant samples is even more felt, in fact usually as the dimensionality increases, the performance of the retrieval systems drastically decreases. In the literature it is possible to find a lot of different solutions that address this problem; recently the researchers focused their attention on methods that reconstruct the data in lower dimensional spaces while maintaining the patterns' distribution [89].

In [102, 103] the author assumed that the data lie on an unknown manifold embedded in the high-dimensional observation space and by means of a procedure, that he called ISOMAP, generated a mapping function from the observation space to a low-dimensional Euclidean feature space. This transformation is obtained preserving the distances between patterns measured along locally shortest paths. This kind of mapping permits that the highly non-linear paths in the observation space became straight lines in the Euclidean space. The idea of finding a subspace where the relevant images are close and the non-relevant one are far away it is further developed over the years, in [81] it has been proposed that each data point and its neighbors lie on or close to a locally linear patch of a manifold and characterize the local geometry of these patches by linear coefficients that reconstruct each data point from its neighbors. The idea of reconstructing the data with a k-nearest neighbor graph, using a weight matrix that indicates the weights on each of the edges, has been also adopted in [63] where a similarity relational graph is constructed by exploring the neighborhood of each image, and two feedback relational graphs are created to depict the relevant and irrelevant relations in the feedback.

While the similarity property is used in order to preserve the local geometry, the relevant and irrelevant feedback information has been used to gathering together the relevant pairs and keeping away irrelevant ones after the embedding. In [44] the authors propose a method to learn a manifold when number of sample images is very small exploiting the user's feedback. They use geodesic distances to approximate the distances between image pairs along the manifold and construct a semantic matrix where the entries are the distances between images seen by the user. In order to find the semantic representation for each considered image and preserving the distances, Laplacian Eigenmaps [6] are used. Finally to map from low-level feature space to semantic space also never seen images, they proposed to exploit a RBF neural network that approximates the optimal mapping function. In [42] the author proposed also a semi-supervised learning scheme based on *locality preserving projection* (LPP) [45] that could be considered a linear variant of the Laplacian Eigenmaps [6] finding the optimal linear approximation of the intrinsic data manifold. More recently the same author in [43], instead, propose to construct a nearest neighbor graph to model the local geometrical structure of the underlying manifold. This graph is then split into a within-class graph and a between-class graph by using class information and neighborhood information. The within-class graph connects two data points with the same label or if they are close to each other, whereas the between-class graph connects data points having different labels. In this way is taken in account both of the local geometrical and discriminant structures of the data manifold. Then, through a linear transformation matrix the images are mapped to a sub space maximizing the margin between relevant and non relevant images (*Maximum Margin Projection*, MMP).

In [7] it has been proposed an algorithm, *Biased Discriminant Euclidean Embedding* (BDEE), that exploits also the low-dimensional representation of the unlabelled samples combining two different linear projection matrices. The first linearly maps the relevant and non relevant images from the high-dimensional space to a low-dimensional space modelling both the intraclass geometry and interclass discrimination, the second reconstructs each unlabelled sample by its neighboring unlabelled samples preserving the local geometry. Exploiting the idea proposed in [81] it is possible to "reconstruct" a point in a N-dimensional space through a linear combination of M points (M < N) with a certain reconstruction error ϵ.

On this topic in this thesis it has been proposed a new interesting idea: using Relevance Feedback to obtain a set of relevant images and one of non relevant image and through a linear combination of both of them reconstruct each image of the dataset. The more an image is close to its "positive" reconstruction and far from its "negative" reconstruction, the more that image should be relevant (see Section 4.5).

3.3.2 Global dimensionality reduction approaches

Differently from these new dimensionality reduction algorithms that has been designed for discovering the local manifold structure, over the years several techniques have been proposed to reduce high dimensional space taking in account the feature space taken as a whole. The most popular surely include *Principal Component Analysis* (PCA) and *Linear Discriminant Analysis* (LDA) [28]. Both algorithms are well known in pattern recognition field and have been widely used in content based image retrieval. LDA is a supervised learning algorithm that finds the directions that maximize the ratio of between-class scatter to within-class scatter. The main issues in the application of this technique are that negative feedback is treated as belonging to a single class and that the lack of sample has a large effect on the effectiveness of the algorithm.

In [120] have been faced these problems proposing a *Biased Discriminant Analysis*. PCA, on the contrary, is an unsupervised method whose aim is to project the data points into a lower dimensional subspace, in which the sample variance is maximized. It computes the eigenvectors of the covariance matrix for the full data set and approximates the original data by a linear combination of the largest eigenvectors that minimizes the mean sum-squared error. The problems of PCA are mainly two. The firs is that the dimension of the low subspace is beforehand fixed, the second is that is not considered the users' subjectivity. In [95] this problem is addressed using different feature spaces and thanks to the user feedback a goodness measure for each feature type is evaluated. According to this measure a different number of component in each feature set is determined.

3.3.3 Feature weighting

Instead of transforming the feature space to discover hidden relation between relevant images, another solution studied by researchers has been the feature selection or more generally the feature weighting that can be considered to implicitly perform it. In fact, following this paradigm, the non-discriminative features will receive a weight near to zero. As above mentioned the idea comes from the principle that the effectiveness of CBIR techniques strongly depends on the choice of the set of visual features. However, no matter how suitably for the task at hand the features have been designed, the set of retrieved images often fits the users needs only partly. This is because in general the exact intent of the user's query cannot be fully captured even when multiple images. As a consequence, it is not possible to choose "a priori" the subset of features that is best suited to a user's query. The basic idea behind weighting mechanisms is that the feedback from the user implicitly defines which images should be considered as neighbors of each other (i.e., the relevant images), and which images should not (i.e., non-relevant images should not be in the neighborhood of relevant images).

The majority of the papers proposed about this topic evaluated the weight in "probabilistic" way. In [83] the authors proposed to use simply the inverse of the standard deviation considering that if a certain component of the feature vector assumes similar value for all relevant images, it means that the component is relevant to the query; on the contrary if all relevant images have different values for that component, then the component is not relevant. In [75] it has been instead used as weights a local relevance measure (*Probabilistic Feature Relevance Learning*, PFRL) evaluated as least-squares estimation, that is, a certain feature is more relevant for the query if it contributes more to the reduction in prediction error. A different approach is used in [116] where have been selected the features with maximum *balanced information gain* obtained from the entropy of the set of labelled images. The same concept could be clearly used as well to weigh not only the components of a single feature space but also subsets of components. The idea stems from the fact that the feature vectors can be composed of “sub-vectors” each describing a different part of the image or a specific characteristic. Therefore giving a greater or a lower weight to one of them it is possible to bring out more clearly what the user is looking for.

In Section 5.1.3 it will be shown, instead, a different point of view from the usual probabilistic way. A different weighting approach will be proposed where the weights associated to a given feature reflect the capability of representing nearest neighbors relations according to the user’s choices. This method is tailor-made for retrieval techniques based on nearest neighbor and as further advantage it is possible to use the same algorithm to weight every component of one feature space, different subset of the feature space or different feature spaces. Borrowing the concept of “classification error” and “Area Under the ROC curve”, well known in pattern recognition, another new different technique provides to iteratively generate weights that minimize this error using the information provided by the user and that reward the features that best interpret the will of the user. This technique will be described in Section 5.2.1.

3.4 Low capability to evaluate similarity between all relevant images

3.4.1 Graph approaches

Nearest neighbor relations are also used in the clustering and neighborhood graphs domain [56]; an interesting proposal that join (co-)clustering and feature selection paradigms has been done in [15]. In the paper the authors exploit the user’s feedback to improve the the co-clustering methodology [26] and iteratively determined the optimal features and weights. Over the years graph-theoretic techniques have been increasingly used for clustering; in fact it is natural to model the similarity between images by a graph whose vertices correspond to images and weighted edges give the similarity between vertices.

There is a large body of literature that deals with this topic, a place of respect is certainly due to *k-means* [28, 54] that, despite its age, is still very popular today because of its simplicity and speed. In accordance with the algorithm, the feedback images are used as seeds for the creation of the clusters, with the relevant images forming the “positive” one and the non relevant images the “negative” one. The main drawback in this approach is that the non relevant images usually does not form unique semantic class so is not suitable modelling them as a cluster. Could be more useful to locate more then two cluster in order to model a (1+x) classes problem but is not possible only with the *relevance/non relevance* judgement of the user’s feedback.

Another one of the most common clustering algorithms is the DBSCAN (Density-Based Spatial Clustering of Applications with Noise) [30] that thanks to the density of the data and the evaluation of the proximity of the patterns, discovers if an image belong or not to a cluster. In [90] the authors proposed a completely different approach to partition the graph: the *normalized cut* method. Once a matrix, whose entries are the measures of the similarity between the nodes of the graph, has been defined, by means of the use of the eigenvalues of the similarity matrix the optimal cut is obtained. The cut, unlike the classical *min-cut* algorithm, is normalized in order to minimize the disassociation between the groups and maximize the association within the groups. In similar way worked the authors of [119] that evaluate a weighting measurement of each image of the database in terms of both relation with respect to the query and the non relevant images, as well as pair-wise information with neighbourhood images. A cost function is obtained for individual images and images are classified by applying the max-flow/min-cut algorithm. The *min-cut* approach has been used also in [85] but differently from the previous, the authors took into account the unlabelled data in the learning process. In addition they proposed to show the images to the user in the Relevance Feedback stage choosing from time to time those that can refine the estimate of the decision boundary or the ones that lie in a uncharted area that could be more meaningful for the next iteration.

More recently, in [113] it has been proposed a content based image retrieval system based on *Dominant set clustering* [73] that exploits the correspondence between dominant set and the extrema of a quadratic form over the standard simplex to discover the images belonging to the different clusters. In their approach the authors used the DSC algorithm for a post-processing step after that a first relevance evaluation has been given to the images by a SVM. The analogies between the concept of a cluster and that of a dominant set of vertices of a similarity graph it is interesting from different points of view, in fact it is possible to use them in the image retrieval field not only as post-processing step but as a real retrieval technique. Following the idea that, if a weight of the cluster node has a small value, then the corresponding image is weakly associated with the cluster and if it has a large value, then the image is strongly associated with the cluster, it is possible to use the node weight as relevance measure. A retrieval approach designed according to this paradigm it will be described in Section 4.4.

3.5 Low informative training images

3.5.1 Active learning

During a query to a database is not uncommon after the first iterations, where the number of relevant images retrieved increases quickly, that suddenly the system does not get more new relevant results. The reason lies in the way in which images are presented to the user, in fact usually they are shown the best ranking images that are returned after each round of feedback and this implies that the search converges towards a local optimum without considering the pattern distribution of the surrounding areas. In order to address this kind of problem it has been proposed a strategy, called *active learning* [19], that requires the system chooses the most informative images to show to the user. The key issue is how to choose the most informative images. Usually this approach has been used in systems based on discriminative functions, i.e. system that builds a decision function which classifies the unlabelled data.

The simplest method is choosing the patterns closest to the decision boundary as described in [109, 49] where SVM based on active learning is used. Differently from the previous works in [17] the authors proposed to learn two SVM classifiers in two uncorrelated feature spaces as color and texture. The classifiers have been then used to classify the images and the unlabelled ones that have been differently classified, have been chosen to show to the user. Also different criteria have been proposed over the years as the minimization of expected average precision [35] or the maximization of the entropy [57]. In the latter paper the authors learned an SVM on the labelled images, mapped the SVM outputs into probabilities and chose the images with the probability to belong to relevant class nearest to 0.5.

Conventional SVM active learning is designed to select a single example for each learning iteration but, as suggested in [48], usually in a relevance feedback iteration the user label as relevant or non relevant image multiple image examples. In this case it is possible the system selects similar images to learn the SVM. The authors to address this problem proposed a *Batch Mode Active Learning* technique that choose the most suitable unlabelled examples one at a time by a min-max approach. In [64] the authors, instead of using SVM, proposed a selective sampling for nearest neighbor classifiers. In order to choose the most informative patterns they suggest to consider not only the uncertainty of the candidate sample point, but also the effect of its classification on the remaining unlabelled points. Their *lookahead algorithm for selective sampling* considers, for this reason, sampling sequences of neighbouring patterns of length k and selects an example that leads to the best sequence. The best sequence is that whose samples have the highest conditional class probabilities. Also in [55] the authors proposed a probabilistic variant of the k-nearest neighbor method for active learning in multi-class scenarios. After that they defined a probability measure, based on the pairwise distances between data points, they used the Shannon entropy as "uncertain" measure over the class labels in order to maximize the discriminating capabilities of the model.

In Section 4.3 an approach based on the nearest neighbor paradigm that follows the active learning approach will be proposed. The k-NN is usually used for its simplicity and because it is not necessary tuning parameters being based on the concept that similar images are located in adjacent areas of space. One of the main problem with this type of classifier is that it retrieves relevant similar images but also non-relevant images similar to each other losing the generalization ability. A solution can come from a methodology that combines the classical nearest neighbor paradigm (Exploitation) with a phase of exploration of the immediate neighbourhoods of the area occupied by the relevant images (Exploration) through a *max-min* approach.

3.6 Low discriminative capability of the feature set

3.6.1 Combining classifiers

For a given image database, a relevance feedback technique that brings the best retrievals to a certain class of query images may be inferior to other RF approaches for another class of query images. According to this behaviour in [117] it has been proposed combining multiple relevance feedback strategies. They proposed a technique that integrates *Query Vector Modification* [80], *Feature Relevance Estimation* [83, 75], and *Bayesian Inference* [20] and selects the most appropriate for a particular query or even for a particular iteration. In [47], on the contrary, the authors proposed to employ the Support Vector Machine ensembles technique to construct a *group-based relevance feedback* algorithm. In their paper they suggested to assume the data come from multiple positive classes and one negative class, i.e. modelled the problem as a *(x+1)-class* classification problem. As above mentioned also in [98, 79] it has been proposed to use a SVM ensemble to address the unbalanced learning, whereas the authors of [110] suggest to use a set of *one-class* classifiers based on *Information Bottleneck* framework [21]. The main part of works that use a classifier ensemble in content based image retrieval field combine the same approach trained on different classes or on different bags of relevant/non relevant images.

In Section 6.1, on the contrary, it will be shown our proposal to combine two very different techniques in order to obtain good results in different datasets and representations rather than high performance in a specific dataset. The two studied techniques are the Support Vector Machines and an algorithm nearest neighbor based. The reason for the choice of these two techniques is that they work well in different feature space and if used with the same representation retrieve different kind of images.

3.6.2 Representation by multi-feature spaces

Instead of the use of multiple classifiers trained on one set of features, another technique to improve the discriminating capability of a retrieval system, is to use different sets of features. In this way it is possible to capture the differences between the images through their different characteristics in fact color, shape, texture or edges of the figures, could be selected or combined in order to have "descriptors" more adaptable to different types of images [82].

In the same spirit, in [108], a large set of *highly selective visual features* has been used, where each feature only responded to a small percentage of images and at the same time only a few features will responded to the relevant images. In this way after that the most selective feature, for a query, has been chosen, each image in the database can be evaluated very rapidly. In [61] the authors proposed a probabilistic feature set weighting. Their weights have been evaluated privileging those spaces that maintain the cluster of the relevant images more concentrated or, in terms of probabilities, giving more importance to features for which the positive examples have a high likelihood, and less importance to features for which the positive examples have a low likelihood.

More recently in [4] it has been proposed a different probabilistic strategy to combine similarity measures. The authors considered a subjective similarity judgement given by users on a fixed set of image and related it to a measure of similarity, then combined the different values evaluated in different feature spaces. The different feature representations can be combined by fusing all the similarity metric through a weighted sum [82, 76]. The main problem combining features is to find a way to increase the accuracy in the search but not the processing time.

In order to do this in Section 5.2.2 will be shown how to represent the images in a "(dis)-similarities" space where each component is associated with a different feature space that accurately represents a similarity measure of the image with the given user query. In this way it is possible to obtain a space whose dimension is equal to just the number of spaces to be combined. Using this technique it has been possible to obtain better performance than the ones obtained with traditional combination methods, with the further benefit of keeping the computational time very low.

Chapter 4

Relevance feedback techniques in individual feature spaces

This chapter focuses on addressing some of the relevance feedback issues described in the previous chapter with techniques that exploit the characteristics of an individual feature space. More in detail, in Section 4.1 is faced the *unbalanced learning* with a technique of pattern injection. In Section 4.2 will be proposed a on-line approach of a relevance feedback technique for content based image retrieval. In Section 4.3 the problem of *low informative training images* is tackled through the proposal of an nearest neighbor approach based on active learning. Section 4.4 presents a graph solution against the *low capability to evaluate similarity between all relevant images* and finally Section 4.5 faces the *high-dimensional feature spaces* problem.

4.1 Directed Pattern Injection

One of the most severe problems in the semantic classification and retrieval of images from very large repositories is the very limited number of elements belonging to each semantic class compared to the total number of images. As training sets for classification and retrieval are usually made up of a small fraction of images per semantic class, it turns out that the learning problems in which elements from different classes are considered, are heavily imbalanced. Our technique artificially increases the number of examples in the training set in order to improve the learning capabilities, reducing the imbalance between the semantic class of interest, and all other images. The new points in the feature space depend on the local distribution of the available patterns of the class of interest, and they are created using the k-NN paradigm. For this reason, the proposed approach is highly effective for relevance feedback techniques based on the Nearest-Neighbor (NN) paradigm, as it allows increasing the generalization capability of NN techniques, and mitigates the risk of classifier over-training on few patterns. The results on different image datasets show the effectiveness of the proposed approach. The improvement in precision and recall gained in one feature space allows also to outperform the improvement in performances attained by combining different feature spaces [78].

4.1.1 Learning from imbalanced data

As mentioned in Section 3.2, in the machine learning literature, a number of techniques have been proposed to address the problem of learning from imbalanced data. The aim of this subsection is not to provide a summary of the literature (see Section 3.2), but rather, before to present our proposal, to briefly describe another technique that is one of the most widely known mechanisms to oversample the minority class by generating synthetic patterns: SMOTE. This description it is necessary in order to better understand the difference between the two techniques and to show how our proposal makes up for some of SMOTE's drawback.

SMOTE

For each pattern of the minority class $\mathbf{x}_k$, new patterns are generated by taking into account its k-nearest neighbors in F one at a time. Let $\mathbf{x}_i$ be one of the the k-nearest neighbors, then a synthetic pattern $\mathbf{x}_{syn}$ is generated by the following formula

$$\mathbf{x}_{syn} = \mathbf{x}_k + \alpha \cdot (\mathbf{x}_i - \mathbf{x}_k) \tag{4.1}$$

where α is a random number in the range $[0,1]$. Typically, the number k of nearest neighbors is chosen to be larger than the number of points that have to be generated in order to balance the data set. Thus, the patterns in the neighborhood actually used to generate the synthetic patterns are usually chosen randomly among the k neighbors.

Directed Pattern Injection

This technique has been originally proposed to inject noisy samples for neural network learning [93]. Actually, this technique can be used to generate new synthetic samples by taking into account all the patterns in a local region of the feature space defined by the neighborhood around one *reference* pattern. The *reference* pattern is typically chosen among the points belonging to the minority class. The use of other points as reference, such as the point computed as the average of the patterns belonging to the minority class has been also considered [77]. The synthetic patterns are generated by a linear combination of the directions defined by the *reference* pattern and its neighbors in F belonging to the minority class, i.e.,

$$\mathbf{x}_{syn} = \mathbf{x}_{ref} + \lambda \sum_{\mathbf{x}_i \in NN(\mathbf{x}_{ref})} \xi_i \cdot \left(\mathbf{x}_i - \mathbf{x}_{ref}\right) \tag{4.2}$$

where the weights ξ_i are drawn from a normal distribution with zero mean and unit variance, and λ is a normalization factor. This formula implicitly assume that the combination of directions generated by one point and its neighbors can generate a new pattern belonging to the class of interest. The validity of this assumption has not been proven formally, and it relies on the observation that if the data of the minority class locally lies on a low-dimensional manifold of the original feature space, then the generated data will lie on the same manifold.

This line of reasoning is also shared by some techniques aimed at discovering non linear manifold embeddings [81]. These techniques compute the low-dimensional embedding by requiring the preservation of local neighboring relations. For a given pattern $\mathbf{x}_k$ these relations are represented in terms of the weights of the linear combination of its neighboring patterns

$$\begin{aligned}\mathbf{x}_{rec} &= \sum_{\mathbf{x}_i \in NN(\mathbf{x}_k)} w_i \cdot \mathbf{x}_i = \mathbf{x}_k + \sum_{\mathbf{x}_i \in NN(\mathbf{x}_k)} w_i \cdot \mathbf{x}_i - \mathbf{x}_k = \\ &= \mathbf{x}_k + \sum_{\mathbf{x}_i \in NN(\mathbf{x}_k)} w_i \cdot \mathbf{x}_i - \sum_i w_i \cdot \mathbf{x}_k = \mathbf{x}_k + \sum_{\mathbf{x}_i \in NN(\mathbf{x}_k)} w_i \cdot (\mathbf{x}_i - \mathbf{x}_k) \end{aligned} \tag{4.3}$$

where $\sum_i w_i = 1$.

The weights are computed so as to minimize the so-called reconstruction error, i.e., the error occurring if the pattern $\mathbf{x}_k$ is represented by $\mathbf{x}_{rec}$, i.e., the linear combination of its neighbors. It can thus be argued that if a real pattern can be approximated by a combination of its nearest neighbors, then the combination of the nearest neighbors of a given pattern can produce synthetic patterns that can be deemed to belong to the minority class with high probability.

4.1.2 Category Learning with Synthetic Feature Vectors

The new patterns, that it is possible to create using the techniques presented in the previous sections, are not “real” new images but synthetic samples generated in the feature space according to the distribution of the available samples. For this reason it has been necessary to take into account the peculiar characteristics of the task at end, so that the new patterns can provide additional information that is embedded into the available samples.

In particular, in the case of image retrieval, the generation of synthetic patterns can be beneficial for relevance feedback techniques, where very often non-relevant images outnumber relevant images. It is worth noting that the choice of the number of artificial patterns to be created is not a trivial task. In fact if the number of them is too large there is the concrete risk to add noise to the dataset, thus producing a distortion of the “real” distribution of images. For these reasons, it has been proposed to constrain the generation of new patterns so that the ratio between images that belong to the class of interest, and those that do not, is constantly equal to a predefined ratio $1 : m$, where $m = 2, \ldots, 5$. In cases when the ratio in the training set exceeds the above ratio, then the artificial generation of patterns is not executed at all.

The number of synthetic patterns that can be generated by SMOTE is limited by the number of available samples. If the number of samples of the minority class is T, then the maximum number of synthetic images that can be generated is equal to $T^2 - 1$. This can be a limiting factor in cases in which the dataset is heavily unbalanced. For example, if the number of patterns of the target class is equal to 2, and the number of patterns belonging to other classes is equal to 20, thus it follows that the total number of synthetic images that can be generated is equal to 3. If the goal is to generate synthetic patterns so that the ratio between patterns of the target class and patterns belonging to other classes is equal to $1 : 2$, then SMOTE in its original formulation does not allow attaining this goal.

On the other hand, in the case of DPI, for each $\mathbf{x}_{ref}$ and its k nearest patterns, it is possible to generate potentially an infinite number of synthetic patterns by varying the coefficients $\xi_1, \ldots, \xi_k$. Consequently, it is possible to train a classifier with a number of samples as large as needed. In addition, these patterns may lie in a region of the feature space defined by the *reference* pattern and its k neighbors, rather than being constrained to lie on the segment connecting to nearest points of the target class. Figure 4.1 shows an example of how the proposed technique works, where the grey area indicates where artificial patterns can be created. Further details on the implementation of the proposed technique will be provided in the following section.

The value of k, i.e., the number of nearest-neighbors considered for the creation of artificial patterns, should not be large to avoid taking into account pattern that are actually far from the *reference* point [93]. For example, the authors of SMOTE suggests the use of $k = 5$, and the patterns actually used to generate the synthetic patterns are drawn randomly from the neighborhood [14].

The proposed approaches are expected to be quite effective when used with learning techniques based on the nearest neighbor paradigm. In fact, new patterns are generated according to a linear combination of directions depending on the distribution of the k-nearest neighbors of a *reference* image, while the coefficients of the combination are random numbers. Starting from the assumption that the patterns of the same class lie in a subspace of the feature space, these techniques permit to identify the subspace determined by the k-nearest neighbors through the linear combination of known patterns and to generate the new ones in it.

The DPI technique depends on the values assigned to a number of free parameters, that are, the image $\mathbf{x}_{ref}$ used as a *reference*, the number of nearest neighbors considered, the number of artificial feature vectors that are generated, and the scale factor λ. These values cannot be "a priori" chosen. On the other hand, tuning all the free parameters in order to find the optimal configuration is a difficult and computationally expensive search task. These parameters have been thus selected according to some heuristics that are reported in the following section.

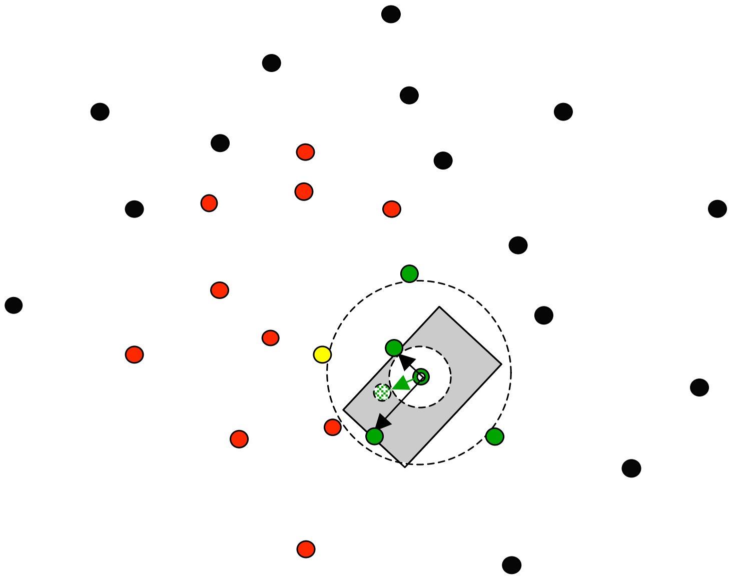


Figure 4.1: Example of k-NN "synthetic" pattern generation in a two-dimensional feature space, where $k = 2$, $\xi_1 = \xi_2 = 0.5$ and $\lambda = 0.5$.

4.1.3 Choice of the Parameters for the DPI technique

There are at least two parameters of DPI that need to be properly adjusted: Should the values of the parameters $\xi_1, \ldots, \xi_k$ be constrained? Which strategy can be used to choose the *reference* point $\mathbf{x}_{ref}$ used to generate the artificial patterns?

Let me provide an answer to the first question. While the proposed approach generates new vectors only in some directions of the feature space (see Eq. (4.2)), the generated patterns may still lie outside the region explored, i.e., the region defined by the known neighborhood of the *reference* point. In my opinion, it is too risky to generate pattern outside that region, as is not possible to have information available about the distribution of images. This can happen depending on the random values of the coefficients $\xi_1, \ldots, \xi_k$. To avoid this risk, it has been proposed to constrain the creation of new patterns in the region delimited by the nearest and the farthest known image of interest w.r.t. the *reference* image $\mathbf{x}_{ref}$ used in equation (4.2). Figure 4.1 shows an example of how the proposed technique works, where the grey area indicates where synthetic patterns can be created, and the two dashed circles bound the area where synthetic patterns are accepted. The green ring represents the *reference* point, chosen as the mean vector of all the images of the target class, while the red, green, and black dots represent non-target, target and unlabelled images, respectively. The green dots pointed by the black arrows represents the two target images nearest to the *reference* point, while the green dot drawn with a broken line and pointed by the green arrow, represents the injected synthetic' pattern. It is easy to see that the new pattern falls in the region containing the samples of the target class, thus increasing the density of this area according to neighborhood information.

Different choices for the *reference* point can be investigated. Let me recall that the directions used to generate new patterns depends on the choice of *reference* point, and the related nearest neighbors. In addition, the choice of the *reference* point may also depend on the application at hand. In an image retrieval problem the *reference* point could be selected as being **the pattern associated to the query image**, as the user asked for images similar to the query. While this can be a reasonable choice, it can also exhibit some drawbacks, as its representation in the low-level feature space may not reflect its representativeness w.r.t. the images the user considers as being relevant. In other words, the so-called "semantic gap" between user perception of similarity and its representation in the low-level feature space may suggest to use a different point in the feature space as a query vector.

As an alternative, the use of **the mean vector of all the known images of the target class** as the *reference* point has been proposed, thus taking into account the distribution of the images of the class of interest in the feature space. This choice, with respect to the first one, takes into account all available information, and thus can be used in different application scenarios, including classification and retrieval. In addition, this choice allows creating synthetic patterns that lie in the region where the images of interests are actually observed .

Other options could be also investigated such as, the use of each known image of the target class as a *reference* image, and the use of a new point computed according to the known target and non-target images. The first option requires some extra parameter to be set, as the total number of artificial patterns that needs to be generated can be smaller than the number of available images, and some target patterns should not participate in the process. The second option requires defining the new point according to some heuristics, e.g., those used for performing query movement in image retrieval tasks [62, 94, 121, 33]. In order to keep the system simple to implement, by reducing the number of parameters which affects the final performance of the system, it has been decided to use the mean vector of all images belonging to the target class as the *reference* point. As above-mentioned, the DPI approach can be effective for techniques based on the nearest neighbor paradigm, in this work it has been resorted to a technique proposed in [31] where a score is assigned according to Eq. (3.11).

4.1.4 Experimental Results

Datasets

This section describes the experimental results related to the use of the proposed technique compared with SMOTE. Experiments have been carried out using two datasets, namely the Caltech-256 dataset and the Microsoft Research Cambridge Object Recognition Image Database (MSRC) (see Section 2.5). Three different kind of features have been extracted, namely the *Tamura* features (18 components), the *Scalable Color* descriptor (64 components), and the *Color and Edge Directivity Descriptor* ($Cedd$, 144 components) (see Section 2.2). The open source library LIRE (Lucene Image REtrieval) has been used for feature extraction [66].

Experimental Setup

In order to test the performances, 500 query images have been randomly extracted for each dataset, covering all the semantic classes. The top twenty best scored images for each query are returned to the user. Relevance feedback is performed by marking images belonging to the same class of the query as relevant, and all other images in the top twenty as non-relevant. It is worth noting that at each round of relevance feedback, the user is asked to mark twenty brand new images never seen before. Performances are evaluated in terms of retrieval precision, recall and the F-measure (see Section 2.4).

Precision is measured by taking into account the top twenty best scored images at each iteration, regardless they have been already labelled by the user. The recall takes into account all the relevant images retrieved so far, including the images labelled by the user to providing the feedback to the system. The recall is evaluated against the minimum between the number of relevant images in the dataset, and the total number of images evaluated by the user (i.e., the maximum number of relevant images that can be actually retrieved). Finally, in order to evaluate the improvement attained by the generation of synthetic patterns (DPI and SMOTE) w.r.t. the nearest neighbor (NN) relevance feedback technique, the following "improvement measure" has been computed:

$$\frac{\left(performance_X - performance_{NN}\right)}{\left(performance_{NN}\right)} \tag{4.4}$$

where X is either SMOTE or DPI, and *performance* is either the precision, the recall or the F measure.

In order to choose the most suitable values of the parameters discussed in Section 4.1.3, a number of preliminary experiments have been performed. Accordingly, the normalization parameter as $\lambda = \frac{1}{k}\left(\xi_1^2 + \xi_2^2\right)^{-2}$ has been computed, and new synthetic patterns at each iteration have been created by taking in account only the information from the last iteration. However, the relevance feedback mechanism takes into account all the images retrieved so far, and all the synthetic images generated so far. Synthetic patterns have been created so that the final ratio between relevant and non relevant images retrieved at the current iteration is equal to $1:2$. Thus it means that artificial patterns are generated only in cases of very imbalanced training sets, and that the number of artificial patterns is kept as small as possible, so that they can provide useful information rather than adding noise. In order to obtain this goal, different values of k for the DPI and SMOTE approaches have been chosen. In the case of DPI, a value of k equal to 2 has been used. In the case of SMOTE, the value of k depends on the number of synthetic patterns needed to attain the ratio between relevant and non relevant images equal to $1:2$. For comparison purposes, relevance feedback has been also computed by a SVM classifier with an *RBF* kernel. Finally, as the three feature representations provide complementary information on image semantic, it has been also evaluated the performances attained by averaging the relevance scores computed separately for each feature representation by the nearest neighbor technique (referred to as NN-Combination).

Results

Figures 4.2, 4.3, 4.4 and Figures 4.5, 4.6, 4.7 show the performances in terms of precision recall and F measure using the two datasets and three feature representations. As expected, the best performances on both datasets are attained by the $Cedd$ representation, as it allows to better capture the different semantic of the classes in the dataset. On the other hand, both the $Tamura$, and the$Scalable\ Color$ representations allow capturing only partially the semantic of the classes. This behavior can be easily seen by comparing the initial retrieval results on both datasets without relevance feedback. It can be also observed that the performance attained by the MSRC dataset are usually higher than those of the Caltech dataset. The reason of this behavior is related to the different semantic of the images contained in the two datasets, and to their subdivision into classes.

Reported results in Figures 4.2, 4.3, 4.4 and Figures 4.5, 4.6, 4.7, show that the artificial generation of patterns allows improving the performance of relevance feedback in all the considered feature spaces, and with respect to the three performance measure. Figures 4.8 and 4.9 shows the average improvements in the three feature sets as described in Eq. (4.4). It can be seen that both DPI and SMOTE allow improving the performance, the recall, and F with respect to the “plain” nearest neighbor relevance feedback technique. In addition, it can be also seen that DPI always outperforms SMOTE. In particular, the main gap between DPI and SMOTE can be observed in the precision figure, where the difference in improvement between DPI and SMOTE is equal to 9% for the Caltech dataset, and 4% for the MSRC dataset. This results can be explained by the different technique employed by DPI and SMOTE in generating synthetic patterns. DPI generates patterns by exploiting the region defined by the mean vector of the patterns of the minority class, and its nearest neighbors, whereas SMOTE generates patterns only on the segment connecting two neighboring patterns. It turns out that DPI allows for a better exploitation of the available information.

In the case of the Caltech dataset, and the $Cedd$ feature representation, Figures 4.5, 4.6 and 4.7 show that the precision attained by the generation of synthetic patterns improves the performances attained by the “plain” nearest neighbor relevance feedback technique, as well as with respect to the use of SVM. In addition, the precision is also higher than that attained by the combination of the three feature representations, starting from the third iteration. At the end of the ninth iteration, the improvement in precision attained by DPI is nearly equal to 3.5%. Thus it can be concluded that the generation of synthetic patterns in one feature representation can be more effective than the combination of information from different feature spaces, when synthetic patterns are generated in an effective feature space for the task at hand. This aspect is particularly interesting from the point of view of the computational complexity. It is worth noting that while the combination of complementary feature representation may allow attaining improvements in performances, the computational overhead of the combination with respect to the most computational demanding feature space (i.e., the $Cedd$ representation) is equal to 77%. On the other hand, the overhead of the proposed techniques based on the synthetic generation of patterns is equal to 15%.

If it is considered the recall and F measures reported in Figures 4.6 and 4.7 , it is possible to observe a behavior similar to the one seen in the case of the precision, apart that the performances of DPI and SMOTE are quite similar each other, the DPI performing slightly better. In particular, significative improvements can be seen since the third iteration.

In the case of the $Tamura$, and the $Scalable\ Color$ representations, it is possible to observe a similar trend as far as the comparison of relevance feedback technique that exploit information on individual feature spaces are concerned. On the other hand, the combination of the three feature sets outperforms all other techniques, as it exploit the information from the $Cedd$ representation. Thus, when the feature spaces are not effective for the task at hand, the generation of synthetic patterns is not competitive with respect to the performances attained by the combination different features.

In the case of the MSRC dataset, it can be seen that using the $Tamura$, and the $Scalable\ Color$ representations, the behavior is the same as the one seen in the Caltech dataset. In particular, the generation of synthetic patterns allows for performance improvements, the DPI providing the best performances in the precision, and F measures (Figure 4.2 and 4.4) since the first iteration. In the case of the recall measure (Figure 4.3), the improvement starts since the fifth iteration, and the performance attained by both DPI and SMOTE reaches the one attained by the combination of the three feature sets at the ninth iteration. In the $Cedd$ feature space, differently from that it happens in the Caltech dataset, the precision attained by generating synthetic patterns is always slightly worse than the one attained by the combination of information from different feature spaces. On the other hand, performances attained for the recall and F measures shows the effectiveness of the generation of synthetic patterns with respect to the “plain” nearest neighbor relevance feedback, the SVM, and the combination of different feature spaces

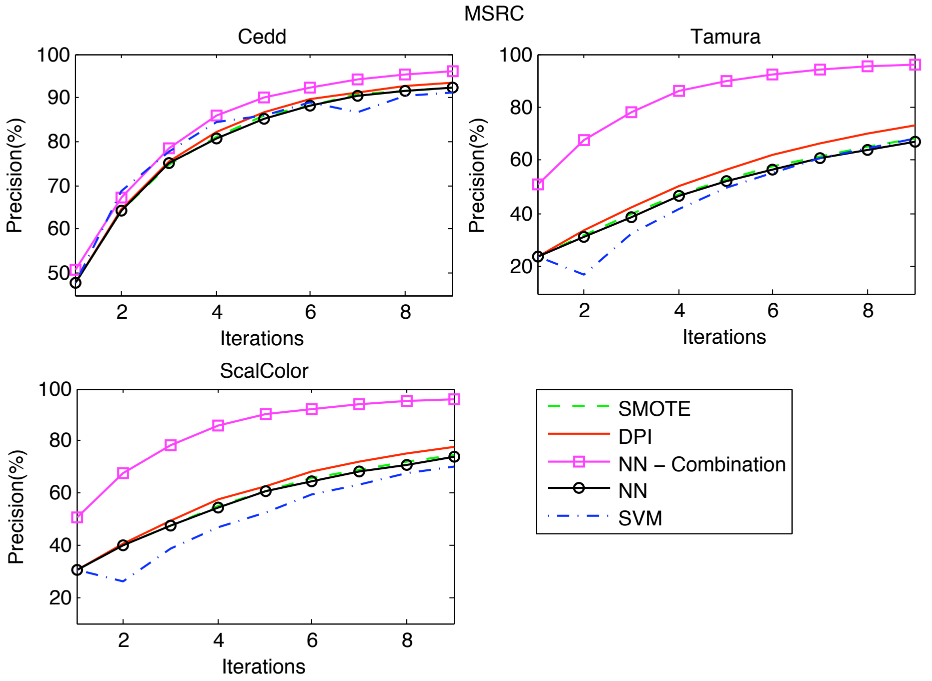


Figure 4.2: MicroSoft Research Dataset - Precision for 9 rounds of relevance feedback.

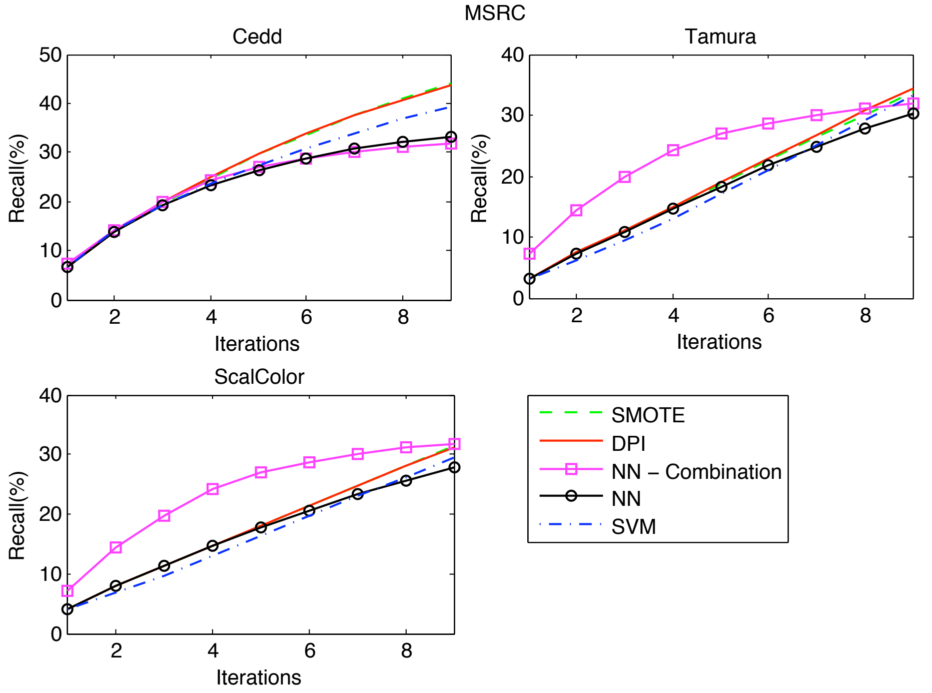


Figure 4.3: MicroSoft Research Dataset - Recall for 9 rounds of relevance feedback.

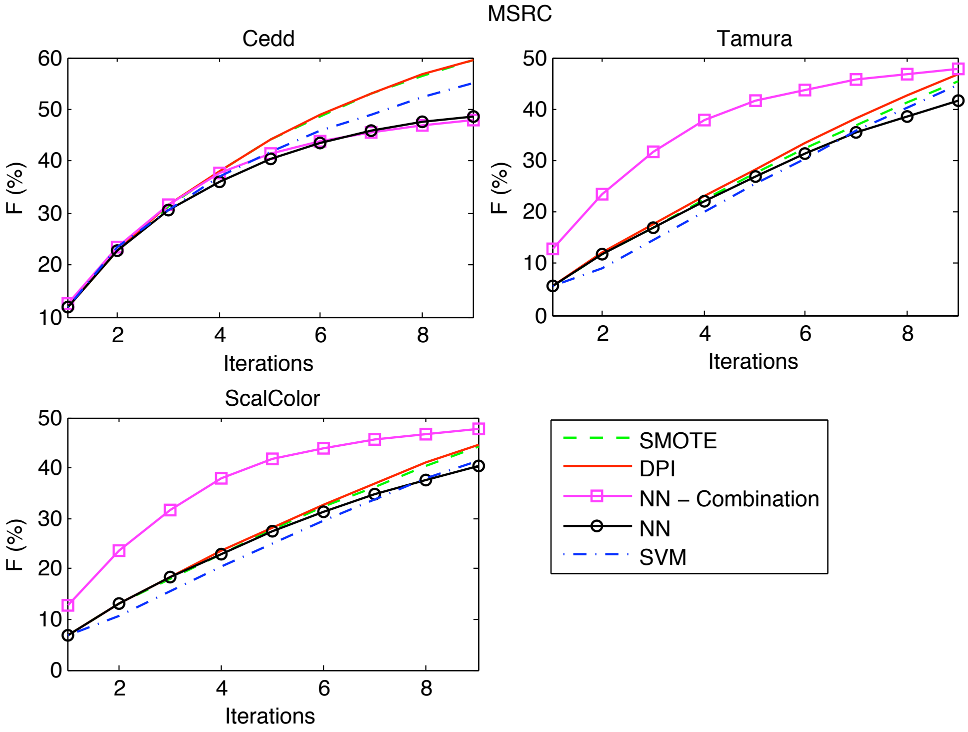


Figure 4.4: MicroSoft Research Dataset - F measure for 9 rounds of relevance feedback.

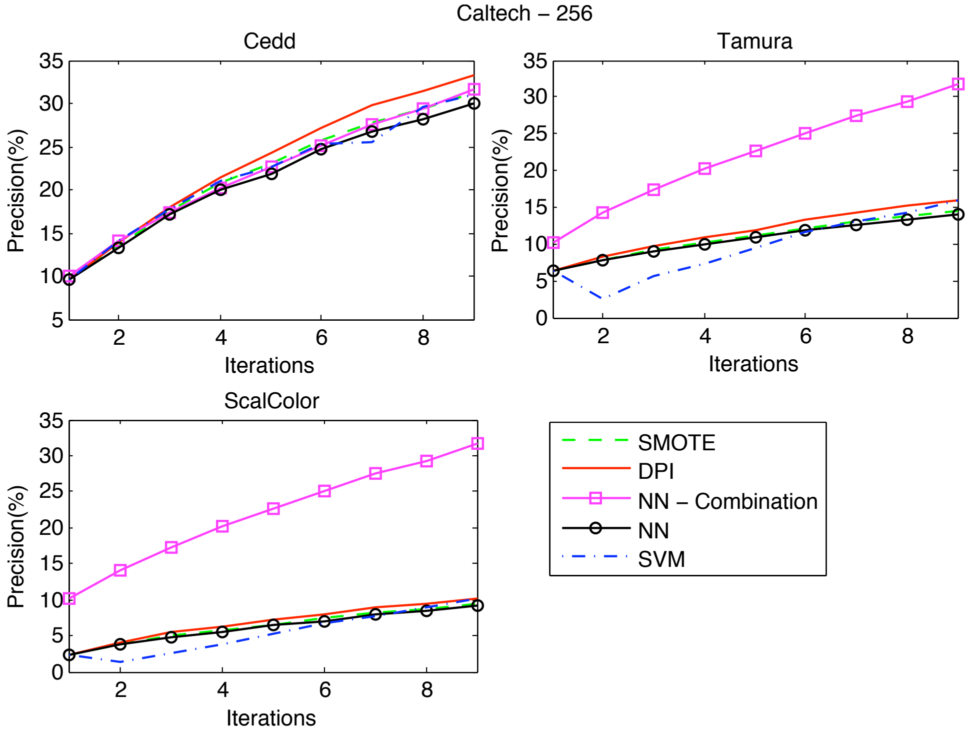


Figure 4.5: Caltech-256 Dataset - Precision for 9 rounds of relevance feedback.

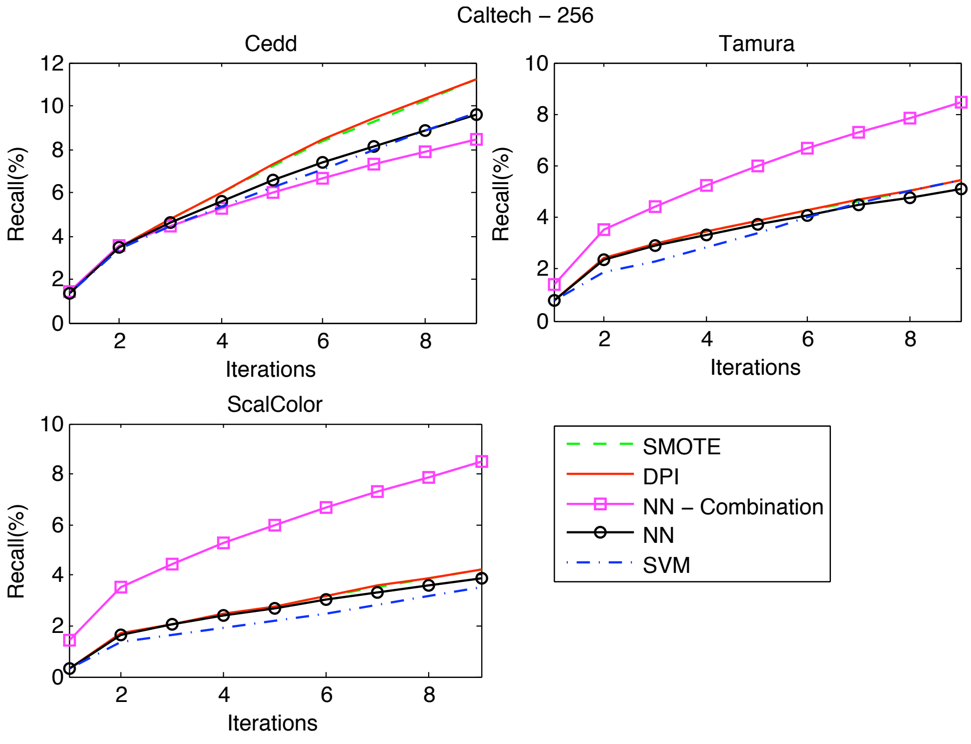


Figure 4.6: Caltech-256 Dataset - Recall for 9 rounds of relevance feedback.

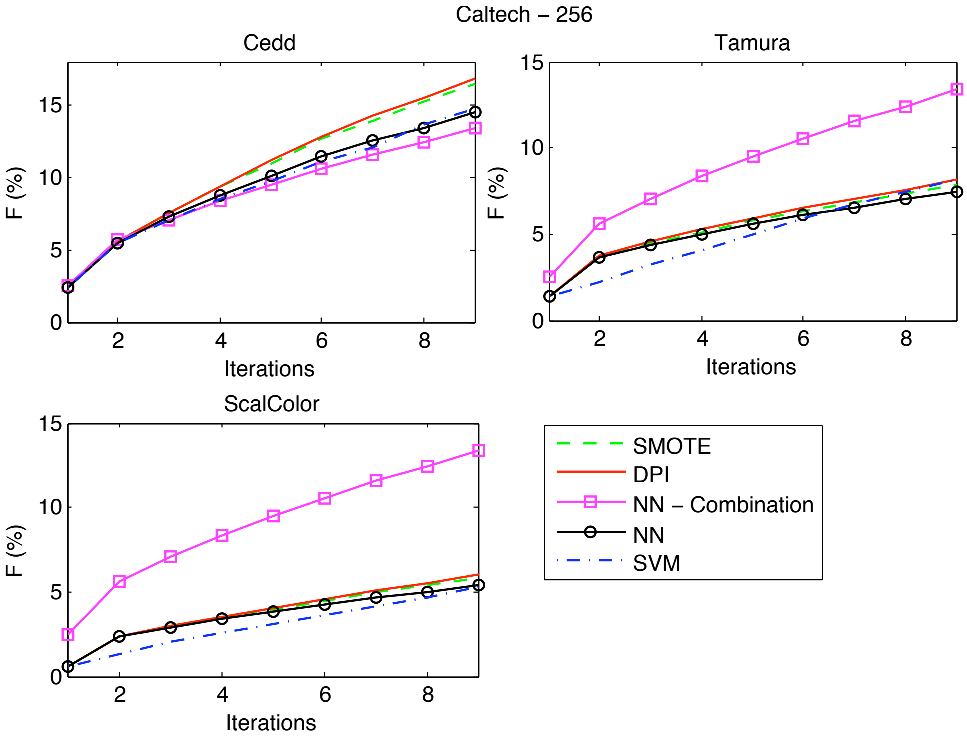


Figure 4.7: Caltech-256 Dataset - F measure for 9 rounds of relevance feedback.

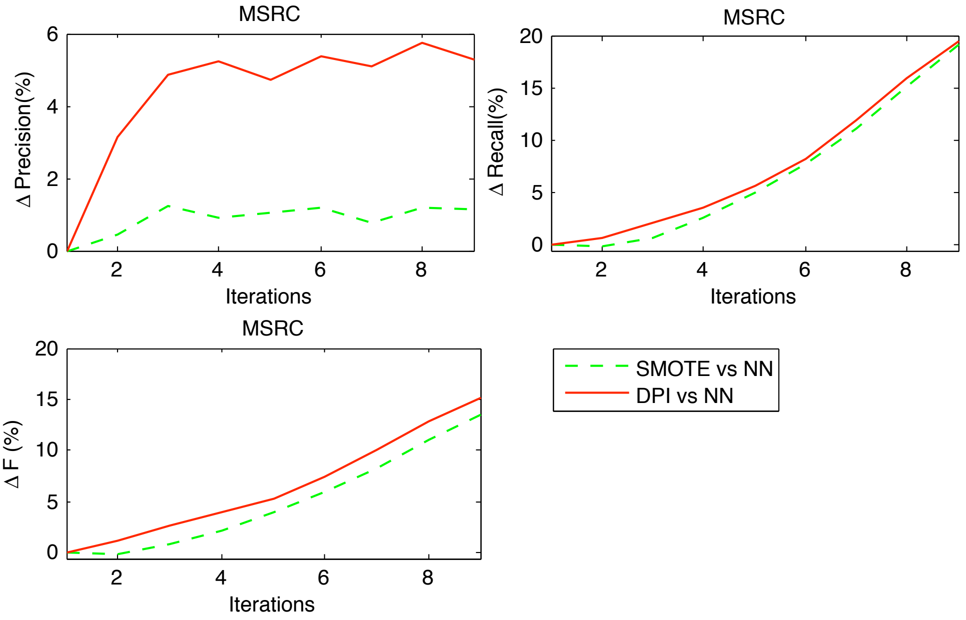


Figure 4.8: MicroSoft Research Dataset - Performance Improvements of SMOTE and DPI against the Nearest-Neighbor technique for relevance feedback, averaged over the three feature spaces.

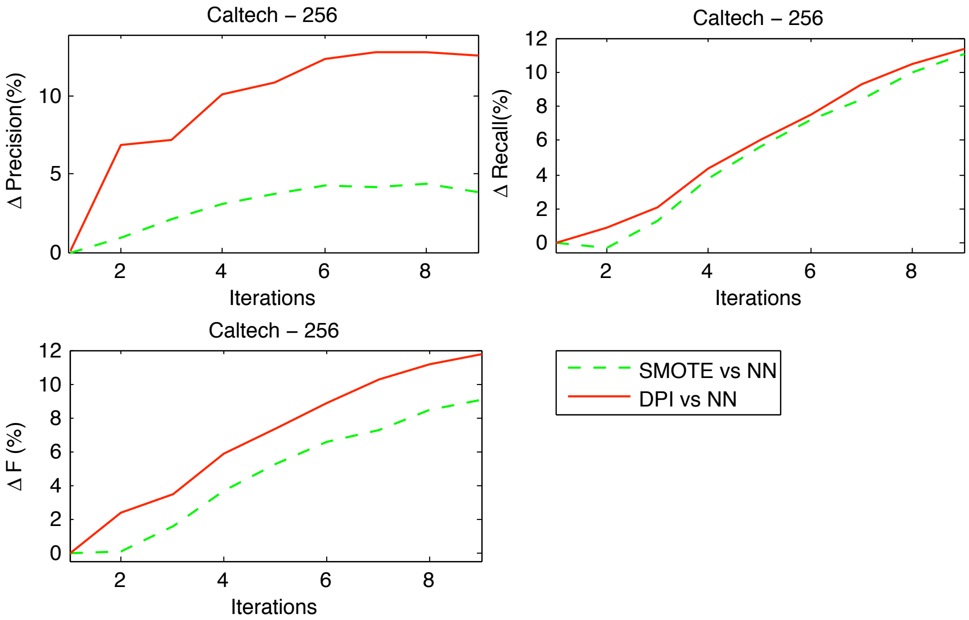


Figure 4.9: Caltech-256 Dataset - Performance Improvements of SMOTE and DPI against the Nearest-Neighbor technique for relevance feedback, averaged over the three feature spaces.

4.2 On-line learning

The increasing consultation of digital images has made necessary to develop more effective systems for their classification and retrieval. The most widely used system for retrieving images by content, nowadays, are based on the SVM or nearest neighbor approaches. Instead strengthening these well know techniques it has been proposed to enhance the RF paradigm by means of some approaches based on on-line learning.

On-line learning is a technique for addressing classification problems [46], binary and multi-class categorization as well regression and sequence prediction problems [22]. It has been successfully used also in problems of images ranking from text queries [36] and video tagging [71]. Our proposal consists in a new application of this technique in the field of Relevance Feedback (RF) for Content Based Image Retrieval. This approach is particularly suitable for facing RF problems due to the *online* nature of such a problem. Moreover with the proposed on-line approach it has been obtained a very fast and compact RF method, this property is very important in a real CBIR scenario. Starting from an idea used in images ranking from text queries problems [36] and in video tagging problem [71], the on-line learning in the content based image retrieval has been exploited .

The problems of the relevance feedback in CBIR will be faced using three approaches based on the on-line learning paradigm [22], one related to a linear predictor and two that extend the the idea through the *kernel trick*. The reported results show that the proposed algorithm is able to achieve results comparable and in many cases higher than those obtained with techniques such as SVM and NN based.

4.2.1 On-line approach

In the classic on-line learning algorithms one pattern at time is evaluated and for each one the algorithm establish if it belongs or not to the class of interest. Every pattern is associated with a unique label $y_t \in \{+1,-1\}$ where $\{+1\}$ indicates the belonging to a certain class and $\{-1\}$ the non belonging. After each judgement the predict class is compared to the true class. According to how the previous prediction was correct or wrong the algorithm improve its prediction rule for the next evaluation.

In the on-line binary classification problem one of the simplest classification approach is the linear classifier [9] where as classification function is used the a vector of weight. Given a vector $\mathbf{w}$ and a pattern $\mathbf{x}$ the sign of $(\mathbf{w}\cdot\mathbf{x})$ classify the pattern as belonging to one or the other class and the magnitude of $|\mathbf{w}\cdot\mathbf{x}|$ is the degree of confidence in the prediction. It is easy to expand the above approach for non linear classifiers, in fact it is possible to replace the product between the two vectors with a Mercer's kernel.

Linear Model for RF

According to the definition of linear classifier, that has been already provided, it is possible to give to an image $\mathbf{x}_i$ a similarity score as follows:

$$score(\mathbf{x}_i) = \mathbf{w}\cdot\mathbf{x}_i, \tag{4.5}$$

where $\mathbf{w}$ is the weight vector to learn. In a CBIR task what a user want is that the relevant images with the query are shown before than the non relevant, when a score ranking is used the desired behaviour is that the first ones have a higher score than the latter, i.e.,

$$\forall \mathbf{x}_r \in R, \forall \mathbf{x}_n \in N, \mathbf{w} \cdot \mathbf{x}_r > \mathbf{w} \cdot \mathbf{x}_n, \tag{4.6}$$

where R and N are the sets of the relevant and non relevant images, respectively. In other words it is necessary to maximize the following expression:

$$\sum_{\forall \mathbf{x}_r \in R} \sum_{\forall \mathbf{x}_n \in N} \mathbf{w} \left(\mathbf{x}_r - \mathbf{x}_n \right). \tag{4.7}$$

Several methods can be applied in order to find a value of $\mathbf{w}$ that maximize the Eq. (4.7). Here it has been followed the same idea proposed in [36] and obtain the weight $\mathbf{w}$ using an on-line iterative algorithm and solving the following equation:

$$\mathbf{w}_t = \underset{\mathbf{w}}{\operatorname{argmin}} \frac{1}{2} ||\mathbf{w}_{t+1} - \mathbf{w}_t||^2 + \mathscr{C} l_t, \tag{4.8}$$

where t is the iteration and l_t is the loss function for the iteration t:

$$l_t = \begin{cases} 0 & \text{if } \mathbf{w}_t \left(\mathbf{x}_r - \mathbf{x}_n \right) > 1 \\ 1 - \mathbf{w}_t \left(\mathbf{x}_r - \mathbf{x}_n \right) & \text{otherwise.} \end{cases} \tag{4.9}$$

From this equation it is possible to see that when the weighted distance between the considered images $(\mathbf{x}_r, \mathbf{x}_n)$ is bigger than 1 the loss is equal to zero, but if it is lower the expression suffers a loss equal to the difference between 1 and the weighted distance. In the Eq. (4.8) it is also possible to see that the loss is “smoothed” by the parameter $\mathscr{C}$ that avoid an excessive fluctuation of the weight between two iterations in succession. The first term of the expression $\frac{1}{2}||\mathbf{w}_{t+1} - \mathbf{w}_t||^2$ plays an important role as well. In fact it forces the values of the weight obtained at consecutive iterations to be close to each other in order to take in account all the information learned in the past iterations. According to [22] the solution of Eq. (4.8) is given by:

$$\mathbf{w}_{t+1} = \mathbf{w}_t + \Gamma_t \left(\mathbf{x}_r - \mathbf{x}_n \right), \tag{4.10}$$

where:

$$\Gamma_t = \min \left\{ c, \frac{l_t}{||\mathbf{x}_r - \mathbf{x}_n||^2} \right\}, \tag{4.11}$$

In order to complete the analysis of the algorithm it is still necessary to discuss how the weights are initialized and how the images $\mathbf{x}_r$ and $\mathbf{x}_n$ are chosen. The algorithm is performed in the following steps:

i) The first action that is performed after a user has provided to the system query image is calculating the distances in the feature space among the input images and all the images in the database and sorting them from the smallest to the largest;

ii) the first k images are labelled by the user as being relevant or not;

iii) henceforth the on-line learning algorithm begins. The weight vector $\mathbf{w}_0$ is initialized to 0;

iv) the images $\mathbf{x}_r$ and $\mathbf{x}_n$ are randomly drawn from the sets of the relevant images (R) and the set of the non relevant images (N) respectively. A new random drawing is performed for each round of updating and the iterative procedure is stopped after a fixed number of round;

v) using the update weight vector a score for each images of the database is evaluated in accordance with the Eq. (4.5);

vi) the images are sorted according to the scores, the first k ones are labelled as in step **ii)**;

vii) a new relevance feedback iteration begins with a new iterative update of the weight $\mathbf{w}$ until the user is not satisfied.

The choice of a predefined number of weight update is motivated chiefly by the number of the retrieved images, in fact if it was evaluated overall pairs and the numerousness of the two sets was high, the procedure would be too computationally expensive. On the other hand if one of the two sets was much smaller than the other (as usual in the relevance feedback field) the weight would be unbalanced toward the bigger set. Using a fixed number of round, on the contrary, permits the images of the smaller class are drawn also more than one time. It is worth noting that the final result is different if the same image is considered at the first update round, or at the second one, or at the last one. In fact the algorithm takes into account, thanks to the first term of Eq. (4.8), all of the previous updates, and thus the same images is considered two or three times in the same update, and an improvement in the weight value can be attained in the same way as considering two or three different images.

Kernel Models for RF

It is possible to easily generalize the algorithm presented in the previous subsection using the so called *kernel trick*. In the following they will be presented two forms of kernel method: a *"pure" kernel* solution and another solution, called *Kernel Pair*, that could be viewed as a middle course between the linear solution and the "pure" kernel solution.

Kernel Trick

In order to better understand the contribution of the kernel in addressing the image retrieval problem it is probably more useful to consider the actual issue as a classification problem instead of a ranking problem. Let me assume that every image $\mathbf{x}_i$ is associated with a unique label $y_t \in \{+1,-1\}$ where $\{+1\}$ indicates the relevant images and $\{-1\}$ the non relevant ones. If it is used the same classification function as described above: the weight $\mathbf{w}$, the sign of $(\mathbf{w}\cdot\mathbf{x})$ could classify the pattern as belonging to one or the other class and the magnitude of $|\mathbf{w}\cdot\mathbf{x}|$ could be considered as the degree of confidence in the prediction. In this case what is desirable is not to maximize the distances between relevant and non relevant images but to maximize the *margin* in the decision. In order to reflect this change it is possible to modify the updating expression (4.10) as follows:

$$\mathbf{w}_t = \sum_{j=0}^{t-1} \Gamma_j \, y_j \, \mathbf{x}_j, \tag{4.12}$$

and therefore

$$score_t(\mathbf{x}_i) = \mathbf{w}_t \cdot \mathbf{x}_i = \sum_{j=0}^{t-1} \Gamma_j \, y_j \, \left(\mathbf{x}_j \cdot \mathbf{x}_i\right). \tag{4.13}$$

In order to use the *kernel trick* it is possible to replace the inner product in the right hand side at Eq. (4.13) with a kernel expression $K\left(\mathbf{x},\mathbf{x}'\right)$ and obtain

$$score_t(\mathbf{x}_i) = \sum_{j=0}^{t-1} \Gamma_j \, y_j \, K\left(\mathbf{x}_j,\mathbf{x}_i\right), \tag{4.14}$$

where according to [22]:

$$\Gamma_t = \min\left\{c, \frac{l_t}{||\mathbf{x}_t||^2}\right\}, \tag{4.15}$$

$$l_t = \begin{cases} 0 & \text{if } d_t > 1 \\ 1 - d_t & \text{otherwise,} \end{cases} \tag{4.16}$$

$$d_t = \sum_{j=0}^{t-1} \Gamma_j \; y_j \; K\left(\mathbf{x}_j, \mathbf{x}_t\right), \tag{4.17}$$

It is interesting to notice that in this approach the vectors $\mathbf{x}_t$ could be seen as *support vectors* as in Support Vector Machine paradigm [23] in fact as in that case the scope is to maximize the *margin* between the relevant and the non relevant images. In this approach only one image per round is randomly drawn from the set of the previous retrieved images and can be drawn more than once. t indicates the number of updating round executed and obviously the evaluated images until that moment.

Kernel Pair

Whereas in the linear approach only the weight is evaluated iteratively and the score is computed one time for each relevance feedback iteration at the end of the weight updating, in the Kernel Pair approach also the score is updated iteratively. For each image $\mathbf{x}_i$ the score is evaluated in the following way:

$$score_t\left(\mathbf{x}_i\right) = \sum_{j=0}^{t-1} \Gamma_j K\left(\left(\mathbf{x}_r - \mathbf{x}_n\right), \mathbf{x}_i\right), \tag{4.18}$$

where: Γ_j is evaluated as in Eq. (4.11), the loss l_t is obtained as in Eq. (4.9) and the weight $\mathbf{w}_{t+1}$ is update as in Eq. (4.10). In this case the weight plays a role in the loss function and so through Γ_j influences the score. The distance between the images of the pair $(\mathbf{x}_r, \mathbf{x}_n)$ instead affects both the weight update and the evaluation of the kernel.

Used Kernels

In this section a briefly review of some of kernels that can be used in the *Kernel Pair* and in the *"pure" kernel* approach will be provided. In particular it will be focused on the RBF kernel, that has been successfully used in a broad range of pattern recognition applications, and on the Histogram Intersection [5] and the Generalized Histogram Intersection [11] kernel that are more suitable for image retrieval.

RBF

The Radial Basis Function is one of the most widespread functions used as kernel and it is expressed in the following way

$$K_{RBF}\left(\mathbf{x}, \mathbf{x}'\right) = \exp\left(-\frac{||\mathbf{x} - \mathbf{x}'||^2}{\sigma^2}\right). \tag{4.19}$$

It easy to see how the expression depends on the distance between the two images but even more it depends on the value of σ that determines the area of influence that the "support vector" has over the space, the bigger is the value of σ the larger is its area of influence.

HI

The Histogram Intersection is a technique proposed in [96] for object recognition and it has been drawn on by Barla in [5] in the image retrieval field. Considering a feature space of dimension m the feature vector that represent an image in that space will be $\mathbf{x} = (x_1, \ldots, x_m)$ and the expression of the kernel:

$$K_{HI}\left(\mathbf{x}, \mathbf{x}'\right) = \sum_{i=1}^{m} \min\left\{|\mathbf{x}_i|, |\mathbf{x}'_i|\right\}. \tag{4.20}$$

GHI

In [11] it has been proposed a generalization of the Histogram Intersection kernel that usually works better in the field of image retrieval. The authors introduce a parameter β greater than 0 that permit to optimize the performance of the task at hand.

$$K_{GHI}\left(\mathbf{x}, \mathbf{x}'\right) = \sum_{i=1}^{m} \min\left\{|\mathbf{x}_i|^{\beta}, |\mathbf{x}'_i|^{\beta}\right\} \text{ with } \beta \geq 0 \tag{4.21}$$

4.2.2 Experimental Results

Datasets

Experiments have been carried out using three datasets, namely the Caltech-256 dataset, a subset of the WANG dataset, and a subset of the Microsoft Research Cambridge Object Recognition Image Database (MSRC) (see Section 2.5). The subset of the WANG dataset (in the following only referred to as WANG) consists of 700 images out of the 1000 that compose the original one. From Caltech-256 two different kind of features have been extracted, namely the *Edge Histogram* descriptor (80 components), and the *Color and Edge Directivity Descriptor* (*Cedd*, 144 components). The open source library LIRE (Lucene Image REtrieval) has been used for feature extraction [66]. The images from WANG are represented by a 512-dimensional *colour histogram* and a 512-dimensional *Tamura* texture feature histogram concatenated in a unique vector, the images of MSRC instead are represented by a vector of 4096 components of SIFT descriptors extracted at Harris interest points (see Section 2.2).

Experimental Setup

In order to test the performances 500 query images from Caltech-256 dataset have been randomly extracted that cover all the semantic classes. For WANG and MSRC datasets each image is used as query. The top twenty best scored images for each query are returned to the user. Relevance feedback is performed by marking images belonging to the same class of the query as relevant, and all other images in the top twenty as non-relevant. Performances are evaluated in terms of precision and mean average precision. The first one is evaluated taking in account the top twenty best scored images at each iteration, in the average precision evaluation all relevant images have been considered (see Section 2.4).

In order to choose the most suitable values of the parameters discussed in the Section 4.2.1 and 4.2.1, a number of preliminary experiments have been performed. Accordingly, the number of updating round t has been fixed at 100, the value σ^2 in the RBF kernel at 0.1, and the parameter β in the exponent of the Generalized Histogram Intersection at 2. In the

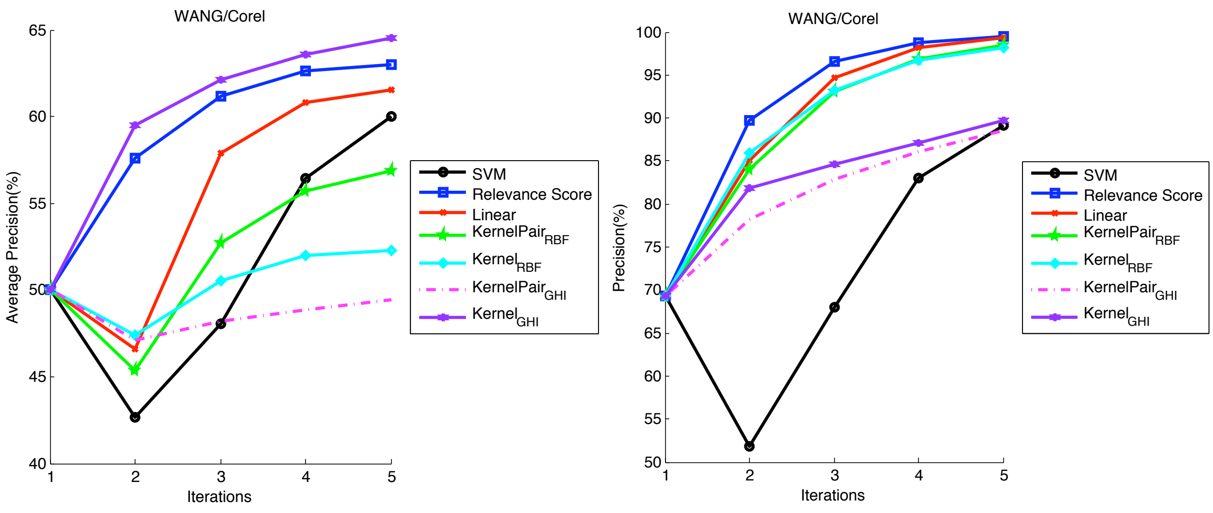
Figures from 4.10 to 4.13, in order to make more clear the graphs, it has been reported only one line for Generalized Histogram Intersection and Histogram Intersection showing only the best of two kernels. In Section 4.2.2 is reported for every dataset the used values of beta. Different values between 0.001 to 1000 have benn also tested for the parameter $\mathscr{C}$ but in this case it is strictly related to the considered dataset.

For comparison purposes, relevance feedback has been also computed by a SVM classifier with an RBF kernel and a nearest neighbor technique based on the computation of a relevance score for each image according to its distance from the nearest relevant image, and the distance from the nearest non relevant image [31].

Results

WANG

Figures 4.10(a) and 4.10(b) show the performance of the proposed approaches compared to the ones obtained by a SVM classifier and a nearest neighbor approach that are the most widely used techniques in CBIR. In Figure 4.10(a) it easy to see how the proposed method overcome the SVM and in particular using the Histogram Intersection kernel ($\beta = 1$) it has been obtained the best result. The RBF kernel instead neither with the "pure" kernel nor the *Kernel Pair* solution is able to obtain values comparable with the SVM. Also the *Kernel Pair* with HI kernel obtains poor result. Regarding the the precision at top 20 images the behaviour of the proposed methods is different: RBF both with the "pure" kernel and the *Kernel Pair* solution reaches a performance comparable with the nearest-neighbor technique and the linear approach. The HI kernel, on the contrary, obtains the performance similar to the ones of the SVM but lower than the other techniques.



(a) Average Precision

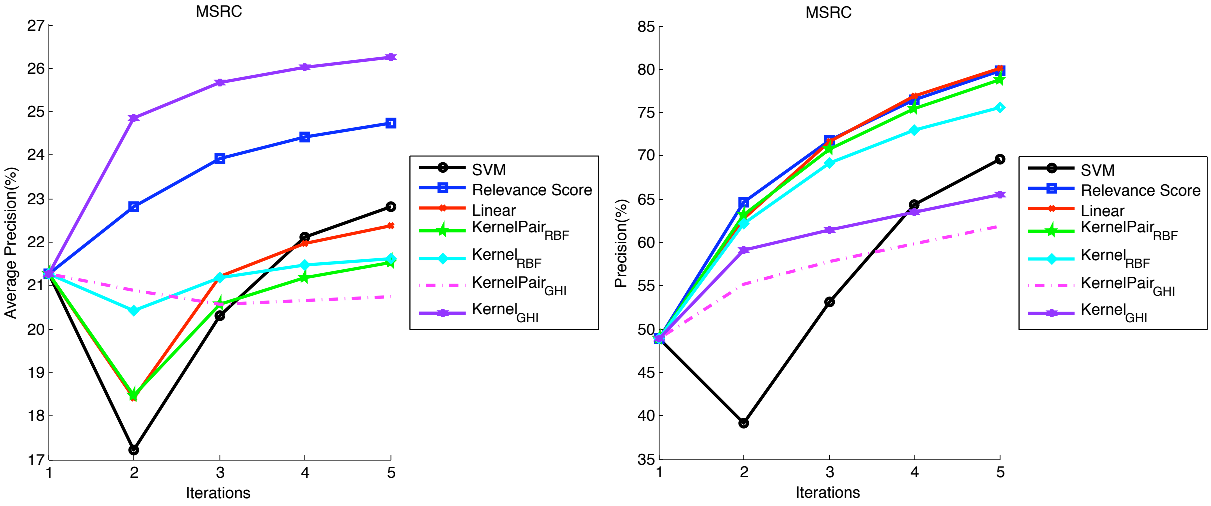
(b) Precision at top 20 pictures

Figure 4.10: WANG Dataset - Average Precision and Precision for 5 rounds of relevance feedback.

MSRC

Figures 4.11(a) and 4.11(b) show the results obtained using the MicroSoft Research Dataset.

In particular Figure 4.11(b) reports the values of the precision at top 20 images and it is possible to see how the trend is the same that one shown for WANG dataset, the linear approach outperforms slightly all the other methods, the RBF "pure" kernel obtains performance comparable with the nearest neighbor approach but higher that the *Pair* approach. Finally, the worst results are attained by the SVM and HI kernel ($\beta = 1$). Regarding he average precision (Figure 4.11(a)) the HI "pure" kernel obtains by far the best performance again, followed by the nearest neighbor approach and the SVM. On the contrary with respect to the WANG dataset the linear approach does not overcome the SVM but its performance is higher than RBF kernel and HI *Kernel Pair* performances.



(a) Average Precision

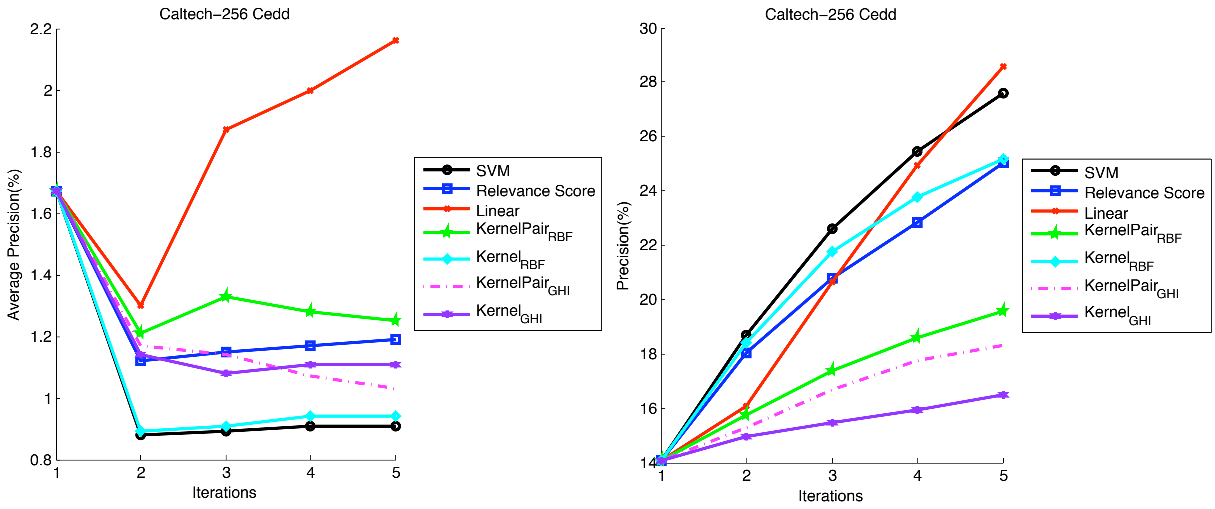
(b) Precision at top 20 pictures

Figure 4.11: MicroSoft Research Dataset - Average Precision and Precision for 5 rounds of relevance feedback.

Caltech-256

For Caltech-256 the experiments have been carried out using two different sets of feature, notwithstanding this the obtained performance is not so different. In figures 4.12(a) and 4.13(a) are reported the results related to the average precision that differently from the performance discussed above show how the linear approach of the proposed method overcome all the other techniques. The second best result is obtained in both the graph by RBF *Kernel Pair* even if using the Edge Histogram descriptors has a better trend. In the third place in the rank of the best results it is possible to find find the nearest neighbor technique that outperforms the two approaches of HI kernel ($\beta = 2$ for both descriptors) and the "pure" RBF kernel. Finally, the worst result is attained by the SVM.

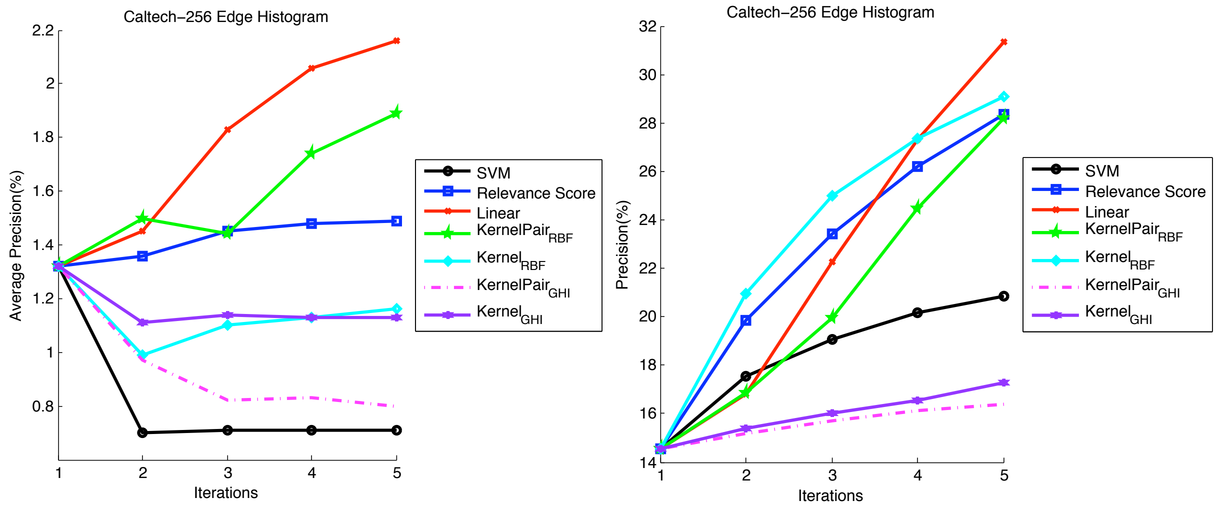
In Figure 4.12(b) are reported the results obtained using Color and Edge Directivity Descriptor and it can be easily seen that the highest performances are provided by the linear approach followed by the SVM. The nearest neighbor approach and the "pure" RBF kernel obtains similar result but better than HI kernel and RBF *Kernel Pair*. Reported results in Figure 4.13(b) show that the precision attained by the linear approach it is also in this case the highest. The nearest neighbor technique and both the RBF kernel approaches, even if with different trend, at the end of the five iterations provide comparable results whereas the worst results are obtained by the HI kernel and the SVM.



(a) Average Precision

(b) Precision at top 20 pictures

Figure 4.12: Caltech-256 Dataset - Average Precision and Precision for 5 rounds of relevance feedback using Color and Edge Directivity Descriptor.



(a) Average Precision

(b) Precision at top 20 pictures

Figure 4.13: Caltech-256 Dataset - Average Precision and Precision for 5 rounds of relevance feedback using Edge Histogram Descriptor.

4.3 Exploitation-Exploration

In this section an active learning approach tailored for the nearest neighbor paradigm will be presented. This approach, over the years, has been widely used with Support Vector Machine, where the choice of the most informative patterns has been driven by its closeness to the decision boundary. The proposed method instead exploits the information obtained from the relevance score computed by a nearest neighbor approach choosing between the best scored images those that lie in an area not too far and not too close to the query. In the following section the proposed method will be presented and some experimental results will be provided.

4.3.1 MAX-min selection

In the classical score based relevance feedback approach after that the user submit a query, the system, according to a similarity measure, scores all images in the database and presents to the user the k best scored ones. This approach could make sure that the system shows to the user several relevant images ever since the first iterations. One of the problem in this kind of behaviour is that in the following iterations the search will be driven by the (probably few) relevant images retrieved and the system could concentrate the search in a limited area of the feature space. Sometime neither the non relevant images, that are also considered in the evaluation of the relevance score by now, can help to get out of this situation. In fact using always the same relevant images iteration by iteration the search can "takes a wrong way". In order to face this problem the proposed method, chosen a fixed number of images that will be the seeds from where it begins the search, selects between a certain number of best scored images those that are "not to close" to the "classical search area". What do "not to close" and "classical search area" mean in the proposed approach will explained in the following.

Let me define k as the number of the images to return to the user and $\lfloor k \cdot \alpha \rfloor$ as the fixed number of the "seed images". Let me also define $(k - \lfloor k \cdot \alpha \rfloor) \cdot \beta$ as the number of images among which they can be chosen the most informative ones. In the previous formulas the parameter α can assume values between $\frac{1}{k}$ and 1, and $\beta \geq 1$. Summing up, with the *Exploitation-Exploration* approach k images are shown to the user, the best scored $\lfloor k \cdot \alpha \rfloor$ are selected beforehand and the others $(k - \lfloor k \cdot \alpha \rfloor)$ are chosen through a *max-min* approach between the $(k - \lfloor k \cdot \alpha \rfloor) \cdot \beta$ best scored images. It is clear that if $\alpha = \frac{1}{k}$ all the images, apart the query, are selected in "active" way, on the contrary when it is equal to 1 they are shown to the user the best k scored images as in the classical nearest neighbor approach. The same happens when $\beta = 1$, in fact in this situation it will be chosen $(k - \lfloor k \cdot \alpha \rfloor)$ images from a set of $(k - \lfloor k \cdot \alpha \rfloor)$ best scored images.

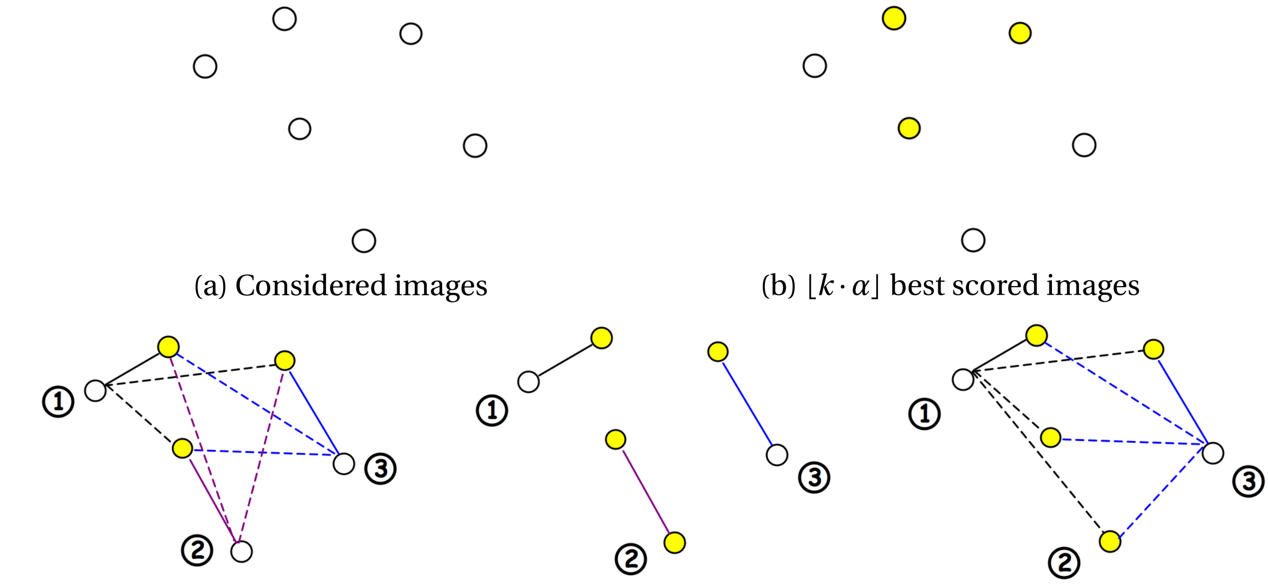
The *max-min* approach selects an image from a set evaluating all the distances between the seed images and the images in the set and choosing for each of them the shortest. The images are then sorted according to these distances and that with the maximum distance it is selected. To better explain the algorithm, in the Figure 4.14 it is reported an example where $k = 4$, $\alpha = 0.75$, and $\beta = 3$

(a) For each images of the database a score is computed according to Eq. (3.11);

(b) the three best scored are used as seems of the search ($\lfloor 4 \cdot 0.75 \rfloor = 3$);

(c) for each of remaining images ($(4 - \lfloor 4 \cdot 0.75 \rfloor) \cdot 3 = 3$) the distances with the seem images are evaluated and the minimum ones are chosen;

(d) the image with the longest minimum distance, the **(2)**, is chosen to be add to the seem images.



(a) Considered images

(b) $\lfloor k \cdot \alpha \rfloor$ best scored images

(c) Evaluation of the all distances

(d) *Max-min* distance

(e) New round of selection

Figure 4.14: Esploration-Exploitation algorithm.

(e) in case of different parameter, e.g., $k = 5$, $\alpha = 0.6$, and $\beta = 1.5$ should be necessary adding another image to show to the user and the algorithm restart from point (c).

4.3.2 Experimental Results

Datasets

Experiments have been carried out using two datasets, namely the WANG dataset and the Microsoft Research Cambridge Object Recognition Image Database (MSRC) (see Section 2.5). The images from WANG are represented by a 512-dimensional *colour histogram* and a 512-dimensional *Tamura* texture feature histogram concatenated in a unique vector, the images of MSRC instead are represented by a vector of 4096 components of SIFT descriptors extracted at Harris interest points (see Section 2.2).

Experimental Setup

In order to test the performances each image has been used as query. The top twenty best scored images for each query are returned to the user. Relevance feedback is performed by marking images belonging to the same class of the query as relevant, and all other images in the top twenty as non-relevant. Performances are evaluated in terms of precision and mean average precision. The first one is evaluated on twenty images taking in account the top best scored images at each iteration and the relevant images retrieved in the previous iterations, in the average precision evaluation all relevant images have been considered (see Section 2.4).

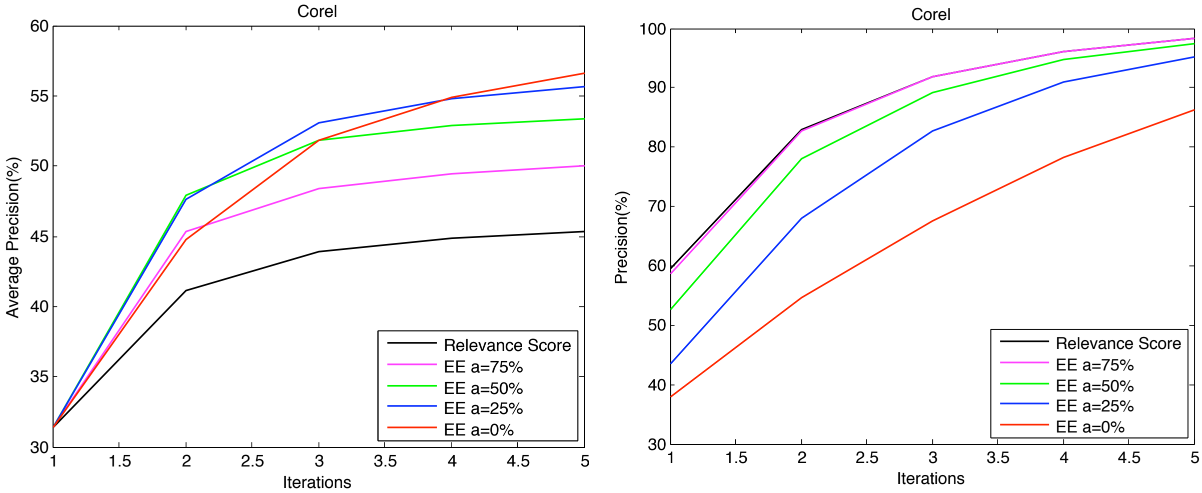
In order to choose the most suitable values of the parameters α and β, a number of preliminary experiments have been performed. Accordingly, it has been fixed the value of β at 10, whereas the different values of α it has been shown for comparison purposes.

The relevance score at each image ha been assigned according to Eq. (3.11) and in the graph it has been referred to as *Relevance Score*, whereas the case $\alpha = \frac{1}{k}$ it has been referred to as $a = 0\%$

Results

Figures 4.15(a) and 4.15(b) show the average precision and the precision rate, respectively, evaluated using the WANG dataset. Observing the image two images it is immediately clear how the two graph have an opposite trend, in fact the lower is the value of α in Figure 4.15(a), the better is the measured average precision. In the precision graph (Figure 4.15(b)), on the contrary, the performance is proportional to α.

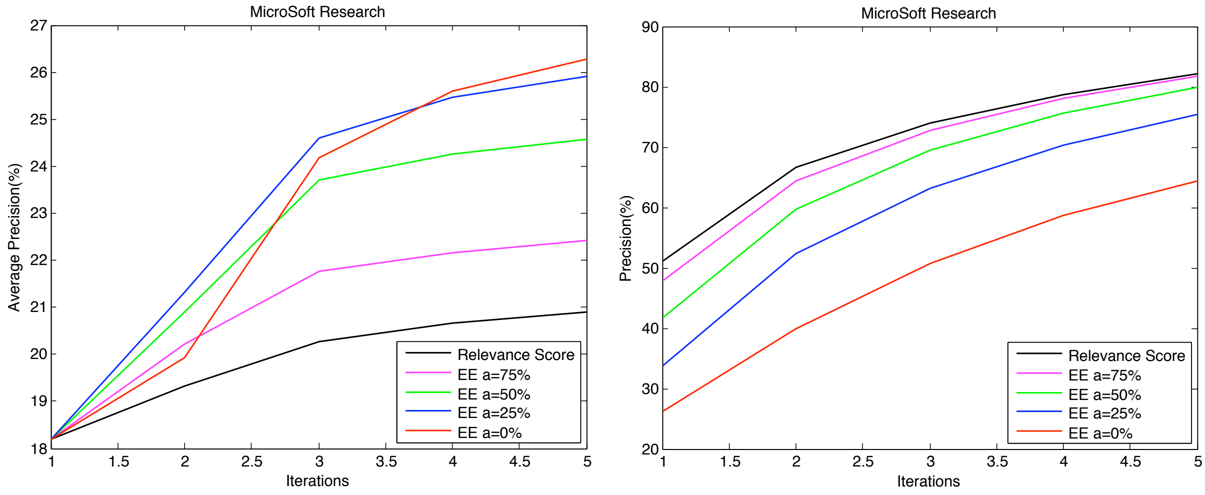
It is possible to see the same behaviour also in Figures 4.16(a) and 4.16(b). At a first sight these results could seem odd but probably the reason is due to the different way in the performance evaluation. The precision in fact is calculated as the ratio between the number of relevant images retrieved and the total number of images retrieved. In these experiments it has been evaluated on twenty images, so one image less it is enough to obtain a loss of the 5%. The average precision, on the contrary is evaluated on all relevant image, so if few of them gain even just one position the performance definitely improves. The value of $\alpha = 0.75$ means that 5 images out of 20 are not chosen between the best scored, so if this choice from a certain point of view improves the performance in terms of the rank of all relevant images, on the other hand can cause that the images presented to the user are more informative but not of the same class of the query.



(a) Average Precision

(b) Precision at top 20 pictures

Figure 4.15: WANG Dataset - Average Precision and Precision for 5 rounds of relevance feedback.



(a) Average Precision

(b) Precision at top 20 pictures

Figure 4.16: MSRC Dataset - Average Precision and Precision for 5 rounds of relevance feedback.

4.4 Dominant set score ranking

The content based image retrieval systems usually assigns a score to each image in the database according to the similarities between the query and the images in the database without taking care the similarities between the images in the database. In order to face this problem it has been proposed several methods that take in account also these “hidden relations” and the widely used methods are those based on the idea of clustering and similarity graphs. Starting from one of these techniques, the *Dominant Sets Clustering* [73], in this section will be proposed a preliminary work that exploit some of the concept of this clustering method in order to develop a innovative relevance feedback method.

4.4.1 Node weighting

Before going deeply into proposed method it is necessary to give before an outline of some of the concepts on which is based the dominant sets theory. It is possible to represent the dataset as an undirected edge-weighted graph with no self-loops $G = (V, E, w)$ [73], where V is the set of vertex, E is the set of edge, and w is the weight function. Vertices in G correspond to images, edges represent neighborhood relationships, and edge-weights reflect similarity between pairs of linked vertices. The weights of the graph can be represented in a matrix A where $a_{ij} = w(i, j)$ and $a_{ii} = 0$. From the complete graph it is possible to extract a subset of nodes such that the overall similarity among nodes of the subset is higher than that between external and internal nodes of the subset. More formally it is possible to define a non-empty subset of nodes $S \subseteq V$ such that the weight of $i \in S$ with regard to S is

$$W_S(i) = \begin{cases} 1, & \text{se } |S| = 1 \\ \sum_{j \in S \setminus \{i\}} \left(a_{ij} - a_{hj}\right) W_{S\setminus\{i\}}(j), & \textit{otherwise.} \end{cases} \tag{4.22}$$

where h is an arbitrary element of $S\setminus\{i\}$. The total weight of S is defined as $W(S) = \sum_{i\in S} W_S(i)$. This allows a sort of indexing of the vertices of the graph, the higher is the value of $W_S(i)$ much more “likely” is that the node i belongs to the cluster. In [73] the authors provide a formal definition of the concept of a cluster in an edge-weighted graph:

Definition 4.1. *A non-empty subset of vertices $S \subseteq V$ such that $W(T) > 0$ for any non-empty $T \subseteq S$, is said to be* dominant *if:*

1. $W_S(i) > 0$, *for all* $i \in S$,
2. $W_{S\cup\{i\}}(i) < 0$, *for all* $i \notin S$

4.4.2 Score evaluation

From this definition and Eq. 4.22, it is possible to find a score related to node weight. The method involves starting from the matrix A whose entries are the distances of each image in the dataset with all the others. Once it has been selected a query, the k nearest are drawn and they are judged by the user as relevant or not. Those that have been deemed relevant are assigned to the cluster S. At this point it is possible to change the matrix A so that the weights “relevant nodes” will be greater than those of the not relevant ones according to [91]:

$$\tilde{A} = A + max(A) \cdot \Lambda \tag{4.23}$$

where A is the weighted adjacency matrix, $max(A)$ represents the maximum entry of A and Λ is a discriminative matrix, where $\Lambda_{i,j}$ is equalt to 1 if the nodes i and j belong to the same cluster, 0 otherwise.

According to the new matrix $\tilde{A}$ and Eq. (4.22) it is possible to evaluate the node weights as $W_S(i)$ $\forall i \in S$ and $W_N(j)$ $\forall j \in N$ where S and N are the clusters of the relevant and non relevant nodes, respectively. The total weight are so computed according to $W(S) = \sum_{i \in S} W_S(i)$ and $W(N) = \sum_{i \in N} W_N(i)$. In order to evaluate the relevance score all images that are not judged by the user are before considered as belonging to the cluster S, then as belonging to the cluster N and the respective weights are computed:

i) $W_{S_k} = W_{S \cup \{k\}}(k)$ and $W_{N_k} = W_{N \cup \{k\}}(k)$ are evaluated;

ii) the improvements with respect to the total weight of S ($W(S_k) = W(S) + W_{S_k}$) and N ($W(N_k) = W(N) + W_{N_k}$) is computed;

iii) the difference between this two weights will be the relevance score for each image;

where k is the index of a generic image.

4.4.3 Experimental Results

Datasets

Experiments have been carried out using a subset of the Corel dataset. It consists of 1290 images that have been manually subdivided into 42 semantic classes with 30 images for each class. Experiments have been performed using *Color Histogram* (see Section 2.2).

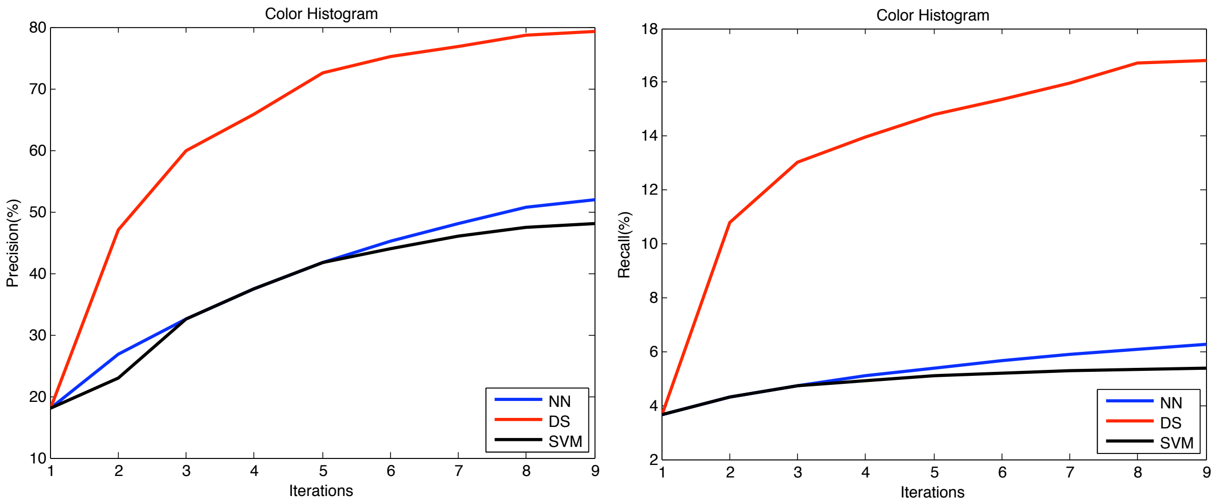
Experimental Setup

In order to test the performances 50 query images have been randomly extracted. Performances are evaluated in terms of precision, recall. The first one is evaluated taking in account the top ten best scored images at each iteration, regardless they have been already labelled by the user. The recall takes into account all the relevant images retrieved so far (see Section 2.4).

For comparison purposes, relevance feedback has been also computed by a SVM classifier with an RBF kernel and with the nearest neighbor relevance score obtained according to Eq. (3.11).

Results

In the Figures 4.17(a) and 4.17(b) precision and recall of the proposed method are reported and it is possible to see how the results obtained by the approach based on Dominant Sets clearly overcome those obtained using SVM and the nearest neighbor approach. These results even if coming from some preliminary experiments and they are obtained in a small dataset seem promising. The choice to use a small dataset has been caused by the computational complexity of the evaluation of the weight if the sets S and N became too large. This computational complexity is due to the recursive form of Eq. (4.22). Anyway the technique seems interesting and it is worth a widening of the investigation. The next step of this research will be find a different way to evaluate the weights and compute the score not considering the non relevant image as a cluster. It is my opinion that this changes could improve the performance and solve the computational complexity problem.



(a) Precision at top 20 pictures

(b) Recall

Figure 4.17: Corel *SMALL* Dataset - Precision & Recall

4.5 Locally Linear Embedding Score

In this section a preliminary work to face, in different way, the problems of high dimensional feature space will be presented. The idea comes from the well known techniques of *Nonlinear Dimensionality Reduction* [81]. The proposed approach is based on the concept of manifold and exploit the property to represent an image as linear combination of its own neighbor. In the next section it will be proposed an overview of the original techniques and in the following will be introduced a modification in order to obtain a relevance score from the image reconstructions.

4.5.1 Reconstruction error evaluation

In [81] the authors compute the low-dimensional embedding by requiring the preservation of local neighboring relations. They suppose that in a d-dimensional space, each data point and its neighbors lies on or near to a locally linear patch of a underlying manifold and characterize the local geometry of these patches by a linear combination that reconstructs each point from its neighbors. Let $\mathbf{x}$ be the generic point and η_j its generic neighbor, the committed reconstruction errors can be evaluated as

$$\epsilon(w) = \sum_i \left| \mathbf{x}_i - \sum_{j=1}^{k} w_{ij}\eta_j \right|^2 \tag{4.24}$$

that adds up the squared distances between all the data points and their reconstructions. The weights that minimize these reconstruction errors can be obtained as [81]

$$w_j = \sum_k C_{jk}^{-1} \left(\mathbf{x} \cdot \eta_k + \lambda\right) \tag{4.25}$$

where C_{jk}^{-1} is the inverse of the neighborhood correlation matrix $C_{jk} = \eta_j \cdot \eta_k$ and λ is the Lagrange multiplier

$$\lambda = \frac{1 - \sum_{jk} C_{jk}^{-1} \left(\mathbf{x} \cdot \eta_k\right)}{\sum_{jk} C_{jk}^{-1}} \tag{4.26}$$

4.5.2 LLE Relevance Score

According to this concept it is possible to generate two reconstructions of each image, one from the relevant images and one from the non relevant ones. Let **r** and **n** two images belonging to the set of relevant (R) and non relevant images (N), respectively. The reconstructions of the image **x** can be formulate as

$$\mathbf{x}_r = \sum_{j=1}^{|R|} w_j^r \cdot \mathbf{r}_j \qquad \mathbf{x}_n = \sum_{j=1}^{|N|} w_j^n \cdot \mathbf{n}_j \tag{4.27}$$

where $|R|$ and $|N|$ are the cardinality of the sets of the relevant and non relevant images, respectively, and w_j^r and w_j^n are the weights that minimize the reconstruction errors using the relevant and non relevant images, respectively. Finally it is possible to evaluate the relevance score as

$$S_{LLE}(\mathbf{x}) = \frac{d(\mathbf{x}, \mathbf{x}_n)}{d(\mathbf{x}, \mathbf{x}_r)} \tag{4.28}$$

At the moment with this technique, they have not been produced results strong enough to be presented in this thesis. However from some preliminary experiments it seems a promising technique, so for this reason it has been proposed as further solution to the high dimensional feature space problem.

Chapter 5

Relevance feedback techniques in multiple feature spaces

This chapter focuses on two problems whose solutions, in a certain sense, permeate each other: *high-dimensional feature spaces* problem and *low discriminative capability of the feature set*. In fact in Section 5.1 it will be presented a way to weight the components of a feature space in order to only select the most signicative so that reducing the space dimension. Another way to do that it could be create a low-dimensional (dis)similarity space from the original one [74]. In Section 5.2 it will be shown a new technique that starting from the idea of (dis)similarity in one space combine multi-feature spaces in order to increase discriminative capability of the system. As the weighted combination of the feature spaces is a well know approach to fuse the different kind of information coming from different sources, the problem of *low discriminative capability of the feature set* is strictly related also on the solutions proposed in Section 5.1.

5.1 Weighted Combination

In this section it will be proposed a weighting mechanism based on the capability of feature space(s) of representing relevant images as nearest neighbors. The approach has a twofold motivation to be investigated, the first is that by identifying which components of the feature space are the most significative it is possible to consider (virtually) only those in the distance evaluations and in this way to overcome the problem of using high-dimensional spaces. The second is that selecting the most appropriate components the relevant images can be represented as neighbors of each other, and the non relevant images, on the contrary, far away from the relevant ones. In the following section they will be outlined the different ways in which weighted metrics can be formulated and they will be briefly described two widely used weighting evaluation techniques. After this necessary overview the proposed technique with its related result will be presented.

5.1.1 Weighted Metrics

Prior to presenting the techniques for estimating the feature weights, let me describe how weighted metrics can be formulated. This discussion is mainly based on the formulation of weighted metrics in [84]. Let me point out, however, that the formulation in [84] is related to the computation of similarities from the query vector, while the following discussion is related to the computation of similarities from relevant and non-relevant images. An image $\mathbf{x}$ can be represented as $\mathbf{x} = \mathbf{x}(F)$, where F is a set of low level features associated with the image, as color and texture. Each feature f_i can be modelled by several representations, e.g. *Color Histogram* and *Scalable Color* are representations of the color feature. Each representation f_{ij} is itself a vector with multiple components

$$f_{ij} = \left[f_{ij1}, \ldots, f_{ijh}, \ldots, f_{ijk}, \ldots, f_{ijN}\right], \tag{5.1}$$

where N is the vector length. For each level f_i, f_{ij} and f_{ijk} is possible to associate respectively a weight w_i, w_{ij} and w_{ijk}. Let me consider a similarity measure M. According to it, is possible to evaluate the similarity between two generic images in terms of f_{ij} and its weights w_{ijk} as

$$S\left(f_{ij}\right) = M\left(f_{ij}, w_{ijk}\right) \text{ with } k = 1 \ldots N, \tag{5.2}$$

For example, to compare two images represented by a vector it is possible to use the Minkowski metric,

$$d_p^{f_{ij}}\left(\mathbf{x}, \mathbf{y}\right) = \left(\sum_{k=1}^{N} \left|\mathbf{x}(f_{ijk}) - \mathbf{y}(f_{ijk})\right|^p\right)^{1/p}, \tag{5.3}$$

with $p \geqslant 1$. Eq. (5.2) can be rewritten as

$$S\left(f_{ij}\right) = \left(\sum_{k=1}^{N} w_{ijk} \left|\mathbf{x}(f_{ijk}) - \mathbf{y}(f_{ijk})\right|^p\right)^{1/p}. \tag{5.4}$$

Moreover it is possible to define another level between the representation one and the vector components one to join them together in accordance with the characteristics of the representations sets. For example an image in the *Color Histogram Layout* representation is divided into G sub-images ($G/4$ horizontal splits and $G/4$ vertical splits) and each sub-images is computed in the Color Histogram representation. Therefore the Eq. (5.1) becomes

$$f_{ij} = \left[g_{ij1}, \ldots, g_{ijk}, \ldots, g_{ijG}\right], \tag{5.5}$$

and

$$\begin{aligned} g_{ij1} &= \left[f_{ij1}, \ldots, f_{ijh}\right] \\ &\vdots \\ g_{ijG} &= \left[f_{ijk}, \ldots, f_{ijN}\right] \end{aligned} \tag{5.6}$$

In the same way as seen before, it is possible to assign a weight w_g to the sets of components, with $g = 1 \ldots G$ and Eq. (5.2) can be extended, for example, to the case in which weights are assigned to the G subsets of the *Color Histogram Layout* representation. In this case Eq. (5.3) becomes

$$S\left(f_{ij}\right) = \sum_{g=1}^{G} w_g \cdot d_p^g\left(\mathbf{x}, \mathbf{y}\right), \tag{5.7}$$

where $d_p^g(\mathbf{x},\mathbf{y})$ is the distance between $\mathbf{x}$ and $\mathbf{y}$ in the g sub-space. Finally, weights can also be estimated for the similarity measure, at the level of the feature f_i

$$S(f_i) = \sum_j w_{ij} S(f_{ij}), \tag{5.8}$$

and furthermore at the highest level

$$S = \sum_i w_i S(f_i). \tag{5.9}$$

Instead of a similarity measure, the weights can be used with a score of relevance that ranks the images as seen in the Eq. (3.11). In particular, weights can be estimated either for combining relevance scores assigned to images for each feature, or for combining scores originated from feature subsets (Eq. (5.6)). In these case Eq. (5.4) and Eq. (5.7) can be rewritten as

$$rel\big(\mathbf{x}(f_{ij})\big) = \sum_{k=1}^{N} w_{ijk} \cdot rel\big(\mathbf{x}(f_{ijk})\big), \tag{5.10}$$

$$rel\big(\mathbf{x}(f_{ij})\big) = \sum_{g=1}^{G} w_g \cdot rel\big(\mathbf{x}(g_{ijg})\big), \tag{5.11}$$

5.1.2 Estimation of feature relevance

Our research has been mainly focused on the problem of weights assignment to the components (or sets of components) of the representation vector and to the relevance scores. The measures of similarity and relevance has be done considering only one representation at a time. For these reasons in general the use of word *features* (or *sets of features*), it will be meant the components (or sets of components) of the representation vector, whereas *feature space* it will be meant a generic representation. The aim of weighting the features in a similarity metric is to assign more importance to those features allowing to retrieve a larger number of relevant images. Typically, the weights assigned to different features are computed by comparing the features values of the query vector with the corresponding features of relevant images.

Inverse of standard deviation

The simplest technique is based on the computation of the *standard deviation* for each feature [83], by taking into account relevant images only. Let me consider a $R \times F$ feature matrix I, where R is the number of relevant images and F is the number of features. Each column of I is a vector composed of the values of the same feature component for all the R relevant images. If the values of the j-th column vector are similar to each other, it means that the relevant images have similar values for the feature j, and more over that the feature is closely related to the query. The larger the similarity between the values, the better the relevance of that feature, and the larger the weights that can be assigned. Therefore the inverse of the standard deviation of each value of vector j is a good weight for the feature j

$$w_j = \frac{1}{\sigma_j}, \tag{5.12}$$

where j is the j-th feature and σ_j is its standard deviation.

Probabilistic learning

Another technique to weight the features is based on the error made by predicting the probability that the value $\mathbf{x}(f_i)$ of the i-th feature component of image $\mathbf{x}$ is equal to the value $\mathbf{z}(f_i)$ of the i-th feature component of the query $\mathbf{z}$ [75]. Let me assume that $y \in \{0,1\}$ is the class label, and $\mathbf{x}_j$ is the j-th retrieved image, with $j = 1 \dots K$. The measure of relevance of the i-th feature component for the query $\mathbf{z}$ is [75]

$$r_i(\mathbf{z}) = \frac{\sum_{j=1}^{K} y_j 1\left(\left|\mathbf{x}_j(f_i) - \mathbf{z}(f_i)\right| \leqslant \Omega\right)}{\sum_{j=1}^{K} 1\left(\left|\mathbf{x}_j(f_i) - \mathbf{z}(f_i)\right| \leqslant \Omega\right)}, \tag{5.13}$$

where y_j can assume value 1 (relevant) or 0 (not relevant) depending on the label assigned by the user, $1(\cdot)$ is a function that returns 1 if its argument is true or 0 otherwise. Ω is chosen such that

$$\sum_{j=1}^{K} 1\left(\left|\mathbf{x}_j(f_i) - \mathbf{z}(f_i)\right| \leqslant \Omega\right) = C, \tag{5.14}$$

where $C \leqslant K$ is a constant. The weight for the i-th feature component is then given by

$$w_i(\mathbf{z}) = \frac{e^{(T \cdot r_i(\mathbf{z}))}}{\sum_{l=1}^{F} e^{(T \cdot r_l(\mathbf{z}))}}, \tag{5.15}$$

where F is the number of the features and T is a parameter which can be chosen in order to increase the influence of r_i with respect to w_i. In fact if $T = 0$, $w_i = 1/F$ for all the features, on the other hand if T is big, the influence of r_i is larger.
The above mentioned weights reflect the relevance that certain features have on the query. If the weight is small, the feature is not helpful to predict the query, whereas the larger the weight, the larger its contribution to predict the query. It is worth noting that the weights estimated by probabilistic learning are closely related with the position of the query regardless of the relative position of the relevant and non-relevant images in the feature space.

5.1.3 Neighborhood-Based feature weighting

While in the previous subsections it has been presented some feature weighting techniques that could be used with nearest-neighbor relevance feedback, in this section a technique specifically tailored to the nearest neighbor technique will be proposed. The aim is to modify the distance metric through appropriate weights so that relevant images will be nearer to each other than non-relevant images to relevant images [76]. This technique is based on the same rationale behind nearest neighbor relevance computation [31]. Thus, first the “relevance” of different feature space is estimated in terms of their capability of representing relevant images as nearest neighbors, then the relevance of an image is estimated according to the relevant and non relevant images in its nearest neighborhood. The degree of relevance of an image can be computed as follows:

$$rel_{NN}(\mathbf{x}) = P(relevant\,|\,\mathbf{x}) = \frac{p_{NN}^{r}(\mathbf{x})}{p_{NN}^{r}(\mathbf{x}) + p_{NN}^{nr}(\mathbf{x})} = \frac{\|\mathbf{x} - NN^{nr}(\mathbf{x})\|}{\|\mathbf{x} - NN^{r}(\mathbf{x})\| + \|\mathbf{x} - NN^{nr}(\mathbf{x})\|} \tag{5.16}$$

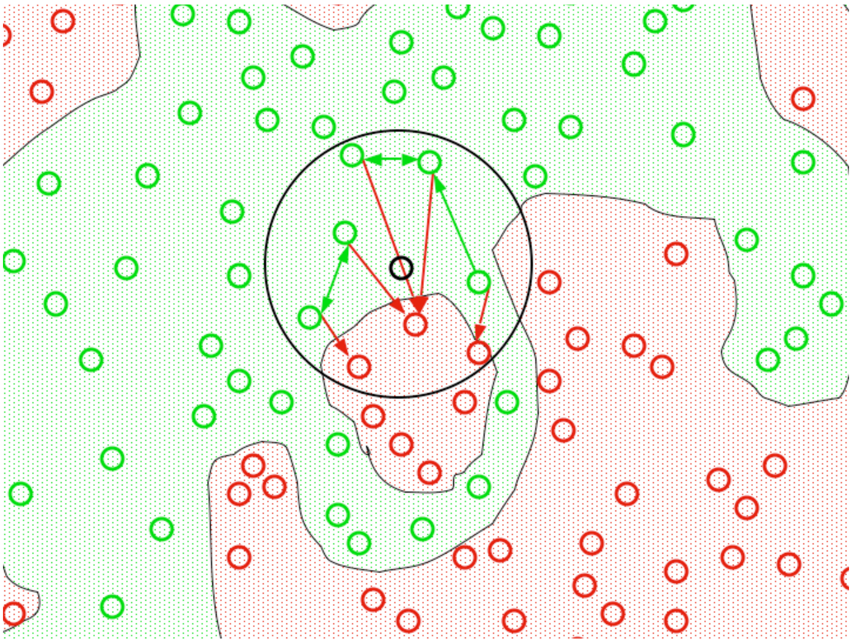


Figure 5.1: The weights consider the positions that relevant and not relevant images assume each other

where p_{NN} is defined as follows

$$p_{NN}(\mathbf{x}) = \frac{1/t}{V(\|\mathbf{x} - NN(\mathbf{x})\|)} \tag{5.17}$$

where t is the number of images and $\mathbf{x}$ is the image considered. $NN(\mathbf{x})$, $NN^r(\mathbf{x})$, $NN^{nr}(\mathbf{x})$ denote the nearest neighbor, the relevant nearest neighbor, and the non relevant nearest neighbor of $\mathbf{x}$, respectively, and V is the volume of the minimal hypersphere centered in $\mathbf{x}$, that contains $NN(\mathbf{x})$. The volume $V(r)$ of hypersphere in a d-dimensional space is expressed as $V(r) = V_d \cdot r^d$ where r is the radius of the hypersphere and V_d is a constant equal to $\frac{\pi^{d/2}}{\Gamma\left(\frac{d}{2}+1\right)}$ and $\Gamma(\cdot)$ is the *Gamma function*.

According to Eq. (5.16) let me estimate the relevance of feature as

$$rel_{NN}(f_x) = \frac{p^r_{NN}(f_x)}{p^r_{NN}(f_x) + p^{nr}_{NN}(f_x)} \tag{5.18}$$

where $p^r_{NN}(f_x)$ and $p^{nr}_{NN}(f_x)$ it is estimated as follows

$$p^r_{NN}(f_x) = \frac{1/t}{V^r_{NN}(f_x)} \qquad p^{nr}_{NN}(f_x) = \frac{1/t}{V^{nr}_{NN}(f_x)} \tag{5.19}$$

where t is the number of images. $V^r_{NN}(f_x)$ it is estimated as the average volume around relevant images which contains its nearest relevant image, and $V^{nr}_{NN}(f_x)$ as the average volume around relevant images which contains its nearest non-relevant image

$$V^r_{NN}(f_x) = \frac{1}{|R|}\sum_{i\in R} d^{f_x}_{min}(\mathbf{x}_i, R) \tag{5.20}$$

$$V_{NN}^{nr}\left(f_x\right)=\frac{1}{|R|}\sum_{i\in R} d_{min}^{f_x}\left(\mathbf{x}_i,N\right) \tag{5.21}$$

where R and N are respectively the set of the relevant and non-relevant images. $\mathbf{x}$ is a generic image, $d_{min}^{f_x}(\cdot)$ is a function that returns the minimal distance between an image and a set of images, and $|\cdot|$ is a function that returns the cardinality of the set. This distance is measured along a component f_x of the features space and is computed as

$$d_{min}^{f_x}\left(\mathbf{x}_i,M\right)=\min\left[d_p^{f_x}\left(\mathbf{x}_i,\mathbf{x}_k\right)\right]\ \forall\mathbf{x}_k\in M, \tag{5.22}$$

where M is a generic image's set. Summing up, the weights associated to each feature f_x can be computed as follows, according to the above equations:

$$w_{f_x}=rel_{NN}(f_x)=\frac{\sum\limits_{i\in R} d_{min}^{f_x}\left(\mathbf{x}_i,R\right)}{\sum\limits_{i\in R} d_{min}^{f_x}\left(\mathbf{x}_i,R\right)+\sum\limits_{i\in R} d_{min}^{f_x}\left(\mathbf{x}_i,N\right)} \tag{5.23}$$

The above reasoning can be also used when the relevance of a set of features has to be estimated, according to Section 5.1.1. Features' relevance has been so far only considered along one dimension at a time, but the relevance can also be captured by examining several features simultaneously. That is, it has been considered that the features were not completely independent. In accordance with the features sets characteristics, each set has been divided in three or four sub sets and a weight to each set has been assigned rather than to the singular features as seen in Section 5.1.1. The Eq. (5.23) can thus be formulated as

$$w_g=rel_{NN}(g)=\frac{\sum\limits_{i\in R} d_{min}^{g}\left(\mathbf{x}_i,R\right)}{\sum\limits_{i\in R} d_{min}^{g}\left(\mathbf{x}_i,R\right)+\sum\limits_{i\in R} d_{min}^{g}\left(\mathbf{x}_i,N\right)} \tag{5.24}$$

where, $d_{min}^{g}(\cdot)$ is a function that returns the minimal distance between two images measured in the $g\text{-}th$ feature set of a certain feature space.

In such a way the more each relevant image is far from its closest non-relevant image, and close to the nearest relevant image, the larger the weight assigned to the feature (or features set) used to evaluate the distance. In fact the smaller is the distance between two relevant images (and the larger the distance from non-relevant images) along certain components (or sets of components) of the feature space, the larger is the probability of finding other relevant images in its neighborhood along those components (or sets of components). Therefore a large weight will be assigned to those features (or features sets). It can be easily see that when the distance between relevant images is large and when the distance between relevant and not relevant images is small, $w_{f_x}\approx 0$. On the other hand, when the relevant images are clustered in a region of the feature space and non-relevant images lie outside the relevant region, then $w_{f_x}\approx 1$.

5.1.4 Experimental Results

Datasets

Experiments have been carried out using a subset of the Corel dataset obtained from the UCI KDD repository (Corel *SMALL*) (see Section 2.5). Experiments have been performed using *Color Histogram* (see Section 2.2). It measures the density of colors in the entire image using the HSV color space (8 ranges for H and 4 ranges for S) so, according to Section 5.1.1, it has been also split into 4 subsets, each subset made up of 8 components.

Experimental Setup

In order to test the performances 500 query images have been randomly extracted. Performances are evaluated in terms of precision, recall and the F measure. The first one is evaluated taking in account the top twenty best scored images at each iteration, regardless they have been already labelled by the user. The recall takes into account all the relevant images retrieved so far (see Section 2.4).

The proposed *Feature-Relevance Nearest Neighbor* (FR-NN) weighting technique has been tested in a nearest neighbor technique based on the computation of a relevance score for each image according to its distance from the nearest relevant image, and the distance from the nearest non relevant image [31]. This approach has been used both with the weighted distance measure in Eq. (5.4), where weights are assigned to each feature component, and with the weighted distance in Eq. (5.7) where weights are assigned to each feature subset (FR-NN SubFeat). Finally, the performances of FR-NN when used to weight relevance scores computed in different feature subspaces (Eq. (5.11)), are also shown.

Reported results clearly show that the weighting scheme is effective when used to compute weighted similarities, rather than to combine relevance scores. It is worth noting that reported experiments are related to the combination of relevance scores computed over each feature component. Thus, the poor performances simply reflect the fact that individual feature components are not effective for computing relevance scores. Retrieval performances have also been compared with the two weighting schemes showed in Section 5.1.2: the *Probabilistic Feature Relevance Learning* (PFRL) (see Eq. (5.15)), and the method of the inverse of the standard deviation (DevSt) (see Eq. (5.12)).

For comparison purposes, relevance feedback has been also computed by a SVM classifier with an RBF kernel and with the nearest neighbor approach without weight.

Results

Figures 5.2, 5.3, and 5.4 clearly show that the use of a weighted distance measure in the framework of the nearest-neighbor relevance feedback improve the performances of the "pure" nearest neighbor technique. In particular, the precision (Figure 5.2) of the proposed FR-NN weighting technique depends on the weighting scheme adopted. When weights are assigned to individual features, the performances decrease with respect to the use of unweighted distance measures. On the other hand, the computation of a distance measure as a weighted combination of distances in different feature subsets, allows attaining the best performances till the eighth iteration. Thus it can be concluded that the proposed weighted distance metric allows improving the performances of nearest neighbor relevance feedback technique, when feature components are grouped according to their meaning.

The results in terms of the Recall measure (Figure 5.3) confirm the effectiveness of the proposed weighted scheme following the trend of the precision results with the only difference in the use of FR-NN to weight relevance scores computed in different feature subspaces. In the precision graph this trend is very low whereas in recall overcome all the other methods. Finally, the graphs reporting the F-measure (Figure 5.4), which takes into in account both the recall and the precision, clearly show that: i) performances of the nearest-neighbor relevance feedback can be improved by computing the relevance of different feature subspaces, and adopting a weighted distance measure accordingly; ii) the proposed technique for estimating the weights of the distance measure can provide better results with respect to other weighting schemes.

It is worth to note how in Figures 5.5 and 5.6 the PRFL precision and recall have a very different behaviour according to the variation of the parameter T (see Eq. (5.15)). It is possible to see, for example, that the best value of T for the recall does not correspond to the best precision. The tuning of this parameter is a long and annoying task and from preliminary experiments has not been possible find a rule to find the best value. On the contrary, the *Neighborhood-Based feature weighting*, that obtained comparable when not better results, does not need some tuning phase.

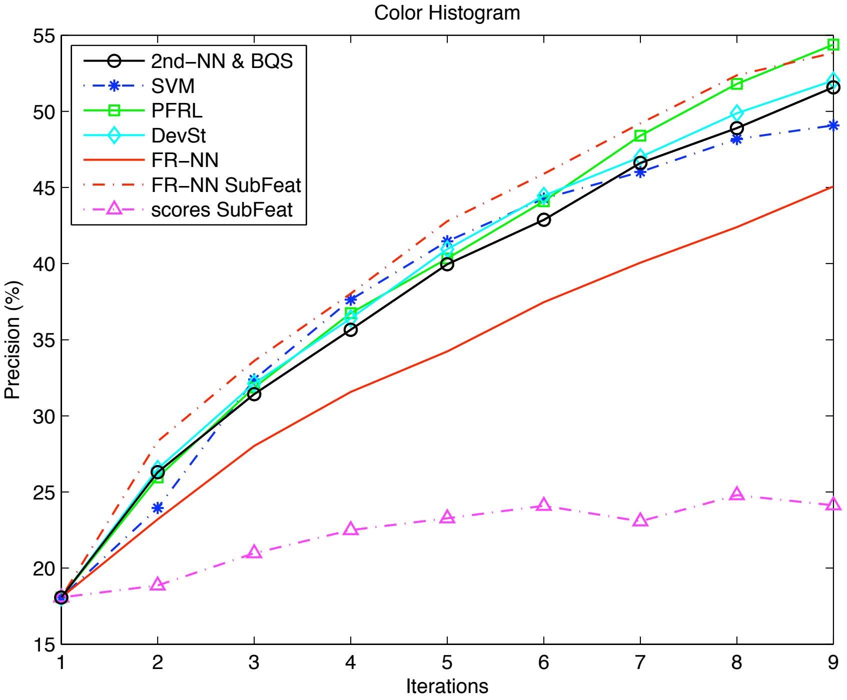


Figure 5.2: Corel *SMALL* Dataset - Precision for 9 rounds of relevance feedback.

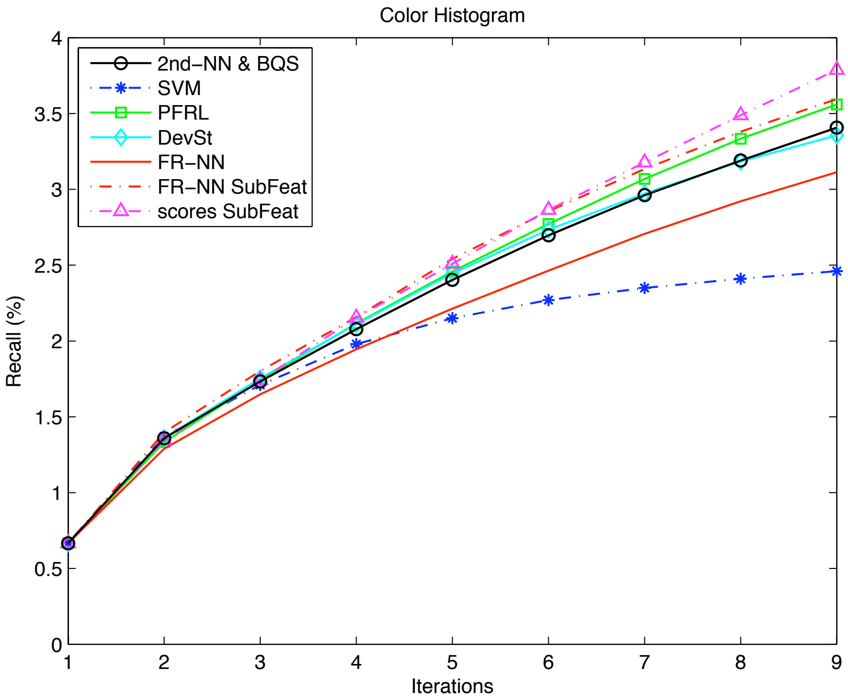


Figure 5.3: Corel *SMALL* Dataset - Recall for 9 rounds of relevance feedback.

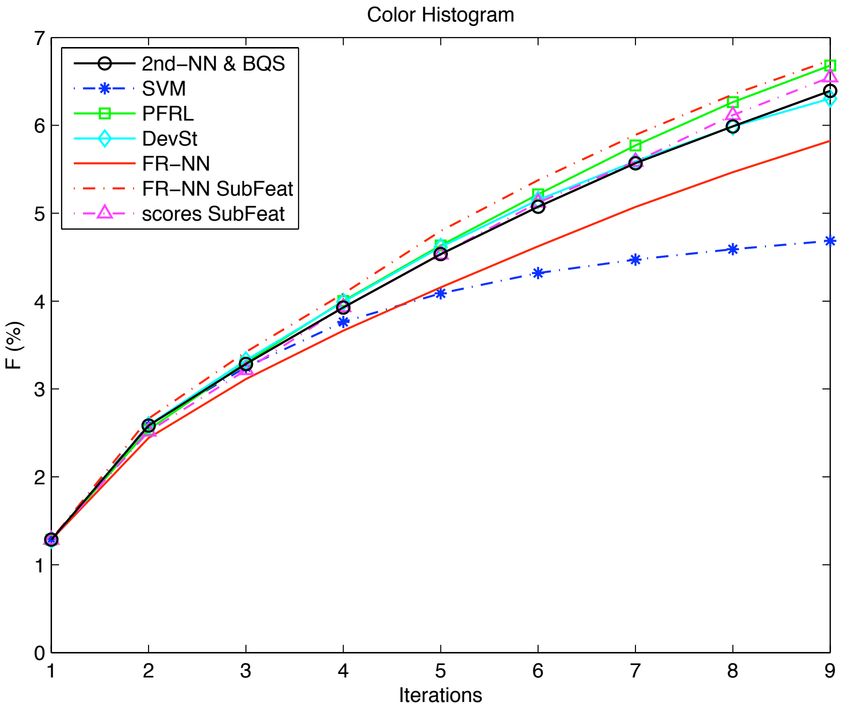


Figure 5.4: Corel *SMALL* Dataset - F measure for 9 rounds of relevance feedback.

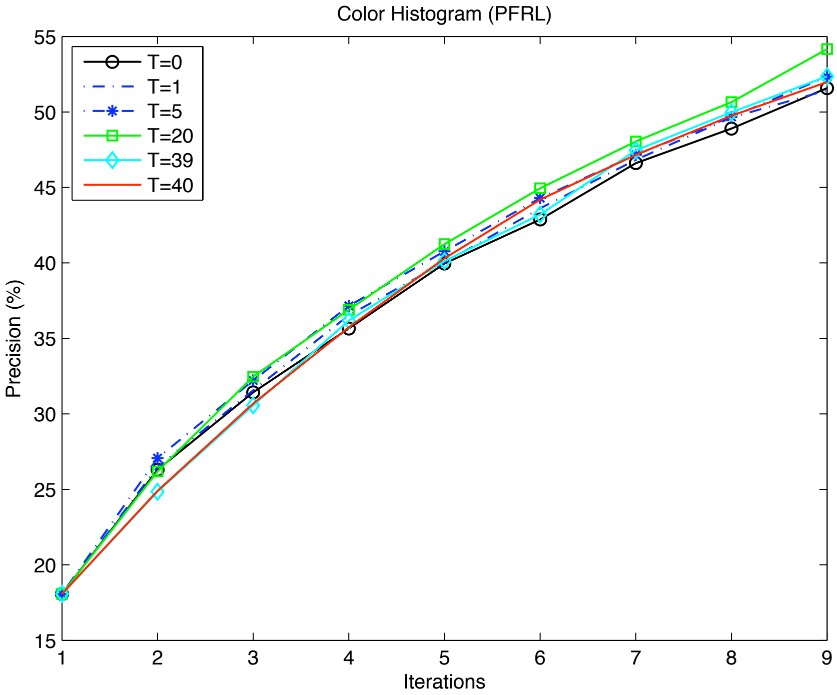


Figure 5.5: Corel *SMALL* Dataset - PFRL Precision for 9 rounds of relevance feedback under T parameter variation.

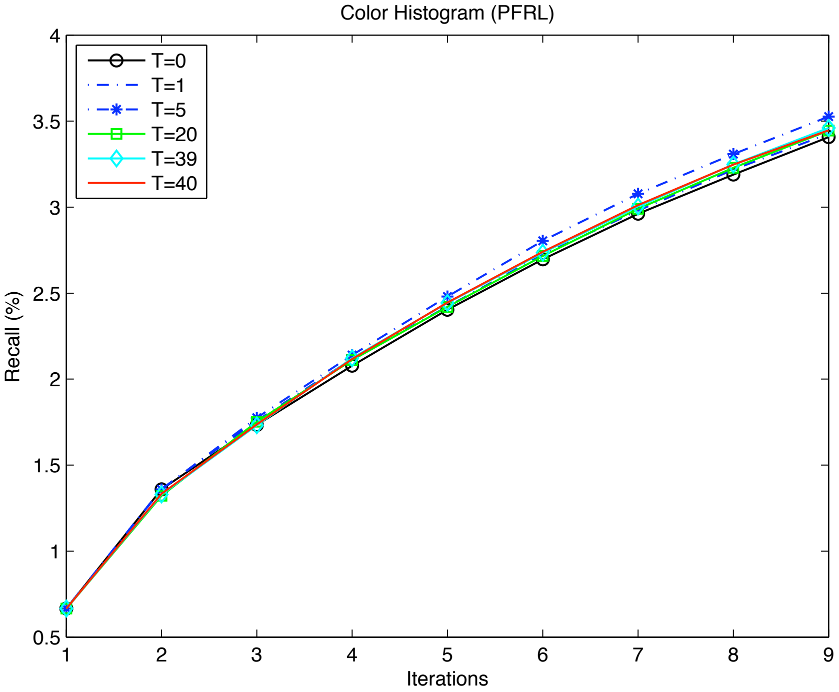


Figure 5.6: Corel *SMALL* Dataset - PFRL Recall for 9 rounds of relevance feedback under T parameter variation.

5.2 Space combination

In this section it will be described two different technique to combine several feature spaces. The first is based on a weighted metric to minimize the nearest neighbor classification error, the second is a new approach that exploit the concept of (dis)similarity space.

5.2.1 Error minimization

It is well known that nearest neighbor paradigm is based on the assumption that an image is as much as relevant as much as its dissimilarity from the nearest relevant image is small. Analogously, an image is as much as non-relevant as much as its dissimilarity from the nearest non-relevant image is small. In the pattern recognition field it is a well known as well the combination of several information from different sources technique can improve the retrieval or classification performance. With this in mind it has been decided to evaluate a unified distance measure that takes in account different metrics in different feature sets and that combines them as a weighted sum. The weights are evaluate in order to advantage those feature space that "draws up" the relevant images each other and that "move" the non relevant ones away. A way to evaluate some weights for nearest neighbor searches (using L2-norm) was investigated in [72] where a weight for each component of the feature vectors of one feature space is evaluated from labelled training-data. In this section will be shown a modification of this approach where more than one feature space is used and a specific metric for each feature space is computed and normalized in order to be compared.

Weight evaluation

In Section 5.1.1 it has been shown how it is possible to express the similarity between two image at different levels: component, feature, and representation level. A "composed" distance can be formulated as

$$\mathbf{d}\left(\mathbf{x},\mathbf{y}\right) = w_1 d_1\left(\mathbf{x},\mathbf{y}\right) + w_2 d_2\left(\mathbf{x},\mathbf{y}\right) + \cdots + w_n d_n\left(\mathbf{x},\mathbf{y}\right) = \sum_{i=1}^{n} w_i d_i\left(\mathbf{x},\mathbf{y}\right), \tag{5.25}$$

where $d_i\left(\mathbf{x},\mathbf{y}\right)$ could be any metric of those described in Section 2.3, n is the number of considered feature space, and w_i is the weight to optimize. According to the nearest neighbor paradigm the (see Eq. (3.12)) optimal weights should be those that minimize the distance among the relevant images and maximize the one between relevant and non relevant images. Proceeding analogously to the procedure proposed in [72] it is possible to couch the function to optimize as

$$J(w) = \frac{1}{Z}\sum_{\mathbf{x}\in R}\sum_{\mathbf{x}'\in R-\{\mathbf{x}\}}\sum_{\mathbf{y}\in N} step\left(\frac{\mathbf{d}\left(\mathbf{x},\mathbf{x}'\right)}{\mathbf{d}\left(\mathbf{x},\mathbf{y}\right)}\right) = \frac{1}{Z}\sum_{\mathbf{x}\in R}\sum_{\mathbf{x}'\in R-\{\mathbf{x}\}}\sum_{\mathbf{y}\in N} step\left(\frac{\sum_{i=1}^{n} w_i d_i\left(\mathbf{x},\mathbf{x}'\right)}{\sum_{i=1}^{n} w_i d_i\left(\mathbf{x},\mathbf{y}\right)}\right) \tag{5.26}$$

where R iand N are the sets of relevant and non relevant images, respectively, $step(\cdot)$ is the *step function*, $Z = |R|\,|R-1|\,|N|$ and $|\cdot|$ is a function that returns the cardinality of the set. In order to optimize this function according to a gradient descent approach it is necessary that it is derivable. To this end the *step function* is substituted by the sigmoid *function*

$$sigm_{\beta}\left(s\right) = \frac{1}{1+e^{-\beta s}} \tag{5.27}$$

and its first derivative is

$$sigm'_{\beta}(s) = \frac{\beta e^{\beta(1-s)}}{\left(1+e^{\beta(1-s)}\right)^2} \tag{5.28}$$

$sigm'_{\beta}$ is a function whose maximum is in $s = 1$ and in $|s-1| \gg 0$ is almost 0. The parameter β controls the trend of $sigm'_{\beta}$: when it is small the function is almost a constant for a wide range of values of z, if it is large $sigm'_{\beta}$ approaches the Dirac delta function (see Figure 5.7). According to Eq. (5.26) the function gradient is

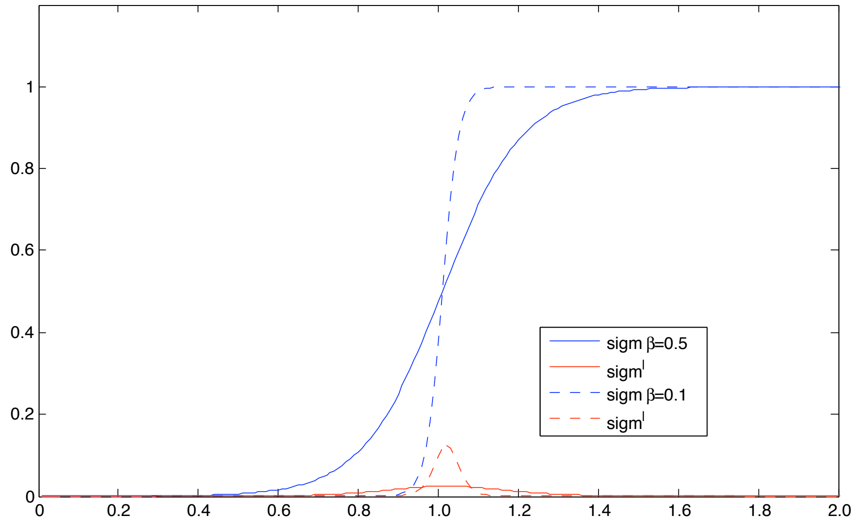


Figure 5.7: Sigmoid function

$$\frac{\partial J(w)}{\partial w_i} = \frac{1}{Z}\sum_{\mathbf{x}\in R}\sum_{\mathbf{x}'\in R-\{\mathbf{x}\}}\sum_{\mathbf{y}\in N}\left[step'\left(\frac{\mathbf{d}(\mathbf{x},\mathbf{x}')}{\mathbf{d}(\mathbf{x},\mathbf{y})}\right)\cdot\frac{d_i(\mathbf{x},\mathbf{x}')\,\mathbf{d}(\mathbf{x},\mathbf{y})-\mathbf{d}(\mathbf{x},\mathbf{x}')\,d_i(\mathbf{x},\mathbf{y})}{\left(\mathbf{d}(\mathbf{x},\mathbf{y})\right)^2}\right]. \tag{5.29}$$

For each query the weights w_i are initialized to one and the distances $\mathbf{d}(\cdot)$ between the query and all images in the database are evaluated. The twenty nearest images are selected for the user feedback and according to Eq. (5.29) the function gradient is evaluated. Finally, it is possible to formulate an iterative algorithm to find the optimum weights. For each feedback iteration it is fixed a number of rounds of weight updates and after each update Eq. (5.29) is evaluated again. The weight update is according to

$$w_i^{t+1} = w_i^t - \mu\left(\frac{\partial J(w)}{\partial w_i}\right) \tag{5.30}$$

where t is the round of updating and μ is the learning rate. After that the weight update is terminated, the distances $\mathbf{d}(\cdot)$ between the query and all images in the database are evaluated again and the twenty nearest images are presented to the user for the feedback phase. It is worth to note that the contribute of the function gradient is as much higher as the ratio, between the distance among the relevant images and the one between relevant and non relevant images, is equal to one. In other words if a feature space is not able to properly separate the relevant images from the non relevant ones, that feature space receive a lower weight.

Experimental Results

Datasets

Experiments have been carried out using three datasets, namely the Caltech-256 dataset, the Microsoft Research Cambridge Object Recognition Image Database (MSRC), and a subset images of the WANG dataset consisting of 700 images (see Section 2.5). From Caltech-256 five different kind of features have been extracted, namely the *Color and Edge Directivity Descriptor*, *ScalableColor*, *ColorLayout*, *Edge Histogram*, and *Tamura* descriptors. The open source library LIRE (Lucene Image REtrieval) has been used for feature extraction [66]. The images from WANG amd MSRC are represented by a 512-dimensional *colour istogram* and a 512-dimensional *Tamura* texture feature histogram and by a vector of 4096 components of SIFT descriptors extracted at Harris interest points (see Section 2.2).

Experimental Setup

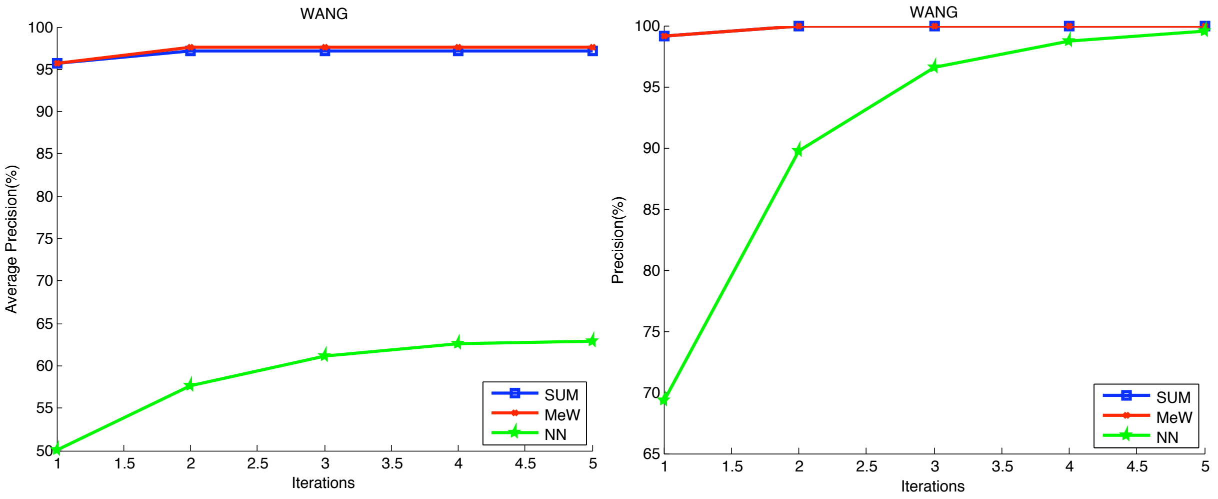
In order to test the performances 500 query images from Caltech-256 dataset have been randomly extracted , for WANG and MSRC datasets each image is used as query. The top twenty best scored images for each query are returned to the user. Relevance feedback is performed by marking images belonging to the same class of the query as relevant, and all other images in the top twenty as non-relevant. Performances are evaluated in terms of precision and mean average precision. The first one is evaluated on twenty images taking in account the top best scored images at each iteration and the relevant images retrieved in the previous iterations, in the average precision evaluation all relevant images have been considered (see Section 2.4).

In order to choose the most suitable values of β and μ, a number of preliminary experiments have been performed. Accordingly, the number of updating round t it has been fixed at 100, the value of μ at 0.001 for all three databases, and the value of β at 10 for WANG and MSRC and equal to 1 for Caltech-256

For comparison purposes a nearest neighbor technique based on the computation of a relevance score [31] has been evaluated using both the sum of all distances computed in different feature spaces above mentioned and using each feature space separately. More in detail WANG has been compared separately using a representation of a 512-dimensional *colour histogram* and a 512-dimensional *Tamura* texture histogram concatenated in a unique vector, MSRC using a vector of 4096 components of SIFT descriptors and Caltech-256 with the *Color and Edge Directivity Descriptor* and the *Edge Histogram* descriptor. Experiments have been also performed using the other features sets, but they have not been reported as they obtained worse results.

Results

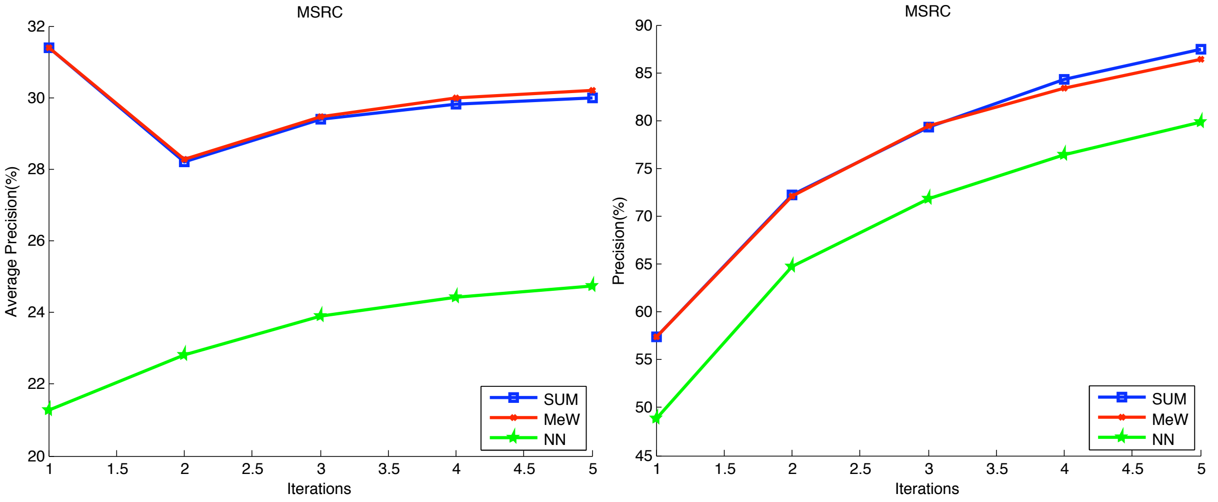
In Figures 5.8(a) and (b) it is possible to see how much is the improvement obtained combining several feature space, both in precision and in average precision especially in the first iteration the result of the combined distances overcome considerably the method that uses only one kind of distance. On the other hand, between the composed distance obtained through the simply sum of all single distances and that obtained with the *Minimization-Error* algorithm there are not a significative differences. A similar behaviour can be also noticed in Figures 5.9(a) in the MSRC dataset and Figures 5.8(b) it is possible even to see a better result for the sum. This behaviour is at clear odds with the results obtained with Caltech-256 (Figures 5.10(a) and (b)), in fact in the figures it is possible to see not only the improvement of the combined method with respect to the performance obtained with *CEDD* or *the Edge Histogram* descriptor, but it is possible to see the better trend of the *Minimization-Error* algorithm with respect to the sum. These difference can be explained taking in account that the high dimensional vectors used with WANG and MSRC obtain a very high results even when used separately and probably do not “catch” different characteristics of the images. On the contrary, the combined distance used in Caltech-256 is composed by very different kind of feature descriptors that are better fused with the *Minimization-Error* algorithm.



(a) Average Precision

(b) Precision at top 20 pictures

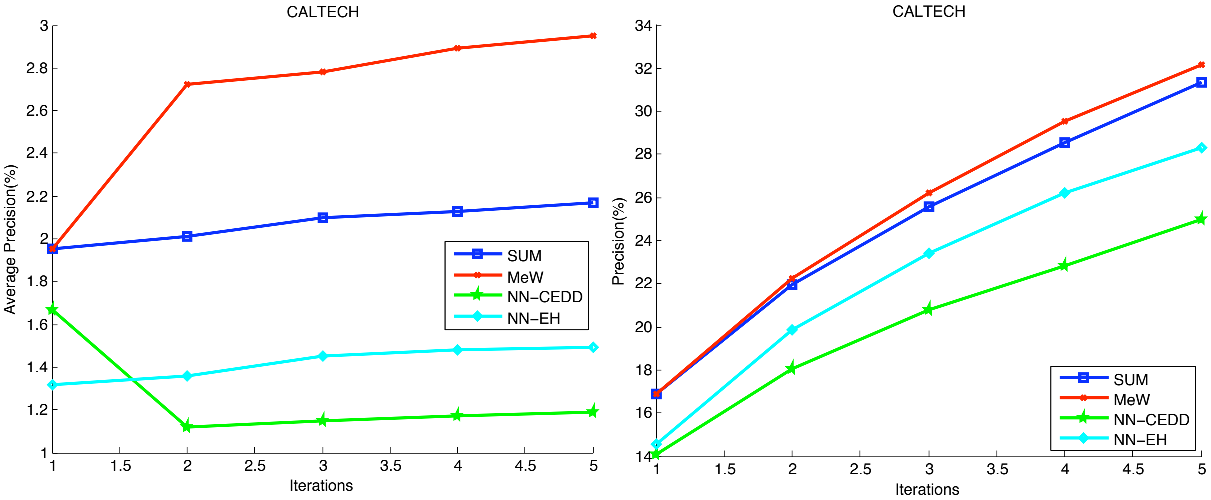
Figure 5.8: WANG Dataset - Average Precision and Precision for 5 rounds of relevance feedback.



(a) Average Precision

(b) Precision at top 20 pictures

Figure 5.9: MicroSoft Research Dataset - Average Precision and Precision for 5 rounds of relevance feedback.



(a) Average Precision

(b) Precision at top 20 pictures

Figure 5.10: Caltech-256 Dataset - Average Precision and Precision for 5 rounds of relevance feedback.

5.2.2 Dissimilarity representation from multi-feature spaces

In this section will be proposed a novel approach to exploit the information provided by different feature spaces for image retrieval tasks. The goal of the proposed approach is to provide an effective way of combining information form multiple high-dimensional spaces, and reduce the cost of computing similarities. This goal it has been achieved by representing images in a "dissimilarity space" [74] where each component is associated to one of the feature spaces, and it is computed as the (dis)similarity of the image with a reference image. This new representation allows the distances between images belonging to the same class being smaller than in the original feature spaces. In addition, it allows computing similarities between images by taking into account multiple characteristics of the images and thus obtaining more accurate retrieval results.

Before entering into the details of the proposed technique, let me recall that the goal is to produce an effective way of combining different feature representations of images in the context of a nearest neighbor relevance feedback approach for content-based image retrieval. Relevance feedback provides the systems a number of images that are relevant to the user's needs at each iteration. It is quite easy to see that if different image representations is considered, usually different sets of images are found in the nearest-neighborhood of relevant images. Which strategy can be employed to assess which of the images can be considered as relevant? One solution can be the use of combination mechanisms based on the weighted fusion of similarity measures computed in different feature spaces. As an alternative, strategies based on the computation of the max, or the min similarity measure can be employed. Finally, the computation of similarity can be carried out using a vector where the components from different representations are stacked. The fusion of similarities requires some heuristics to compute the weights of the combination, while the max and min rules can be more sensitive to "semantic" errors in the evaluation of similarity . Finally, the use of stacked vectors can be computationally expensive, and can suffer from the so-called curse of "dimensionality", as the dimension of the resulting space may be too large compared to the number of available samples of relevant images.

In order to provide a solution to the computation of the relevance of an image with respect to the user's goal by exploiting information from different image representations, it has been proposed to construct a dissimilarity space by computing the dissimilarities from a single prototype using multiple feature representations. Reported results show that the proposed technique outperforms other combination techniques not only in terms of precision and recall, but also in terms of the execution time. It is also quite effective when relevance feedback is exploited. In the following sections it will be before introduced the concept of dissimilarity space then will be presented a brief overview of some techniques to combine different feature spaces. Finally, a description of the proposed method and its experimental results will be provided.

From multi-spaces to dissimilarity spaces

Classic dissimilarity spaces

For a given classification task, P prototypes are chosen, and the distances $d(\cdot)$ between each pattern and the prototypes are computed. These distances can be computed in a low-level feature space. Each pattern is then represented in terms of a P-dimensional vector, where each component is the distance between the pattern itself and one of the P prototypes. Let me define

$$P = \{\mathbf{p}_1, \ldots, \mathbf{p}_P\} \tag{5.31}$$

the set of prototypes $\mathbf{p}$ and let $d\left(\mathbf{x}_i, \mathbf{p}_j\right)$ be the distance between pattern $\mathbf{x}_i$ and the prototype $\mathbf{p}_j$. Pattern $\mathbf{x}_i$ in the dissimilarity space will be thus represented as follows:

$$\mathbf{x}_i^P = \left[d\left(\mathbf{x}_i, \mathbf{p}_1\right), \ldots, d\left(\mathbf{x}_i, \mathbf{p}_P\right)\right]. \tag{5.32}$$

It should be quite clear that the performances depend on the choice of the prototypes, especially when this technique is used to transform a high-dimensional feature space into a lower dimensional feature space. The literature clearly shows that the choice of the most suitable prototypes is not a trivial task [74].

Techniques for Combining Different Feature Spaces

In the introduction of the section a number of techniques that can be used to combine different image representations have mentioned. In the following a briefly review of some of them it will be presented, because they will be used in the experimental section for comparison purposes. In particular, seven different combination methods will be shown. Four combination methods aims to combine the relevance scores computed after relevance feedback, while the other three methods aim at combining distances.

The four techniques used to combine the relevance scores computed separately in each of the available feature spaces f are the following:

$$S_{\text{MAX}}\left(\mathbf{x}_i\right) = \max_{f \in F}\left(S_f\left(\mathbf{x}_i\right)\right) \tag{5.33}$$

$$S_{\text{min}}\left(\mathbf{x}_i\right) = \min_{f \in F}\left(S_f\left(\mathbf{x}_i\right)\right) \tag{5.34}$$

$$S_{\text{Mean}}\left(\mathbf{x}_i\right) = \frac{\sum\limits_{f \in F} S_f\left(\mathbf{x}_i\right)}{|F|} \tag{5.35}$$

where F is the set of the feature spaces and S is the score evaluated according to Eq. (3.11). RR weight is the weighted sum of the relevance scores, where the Relevance Rank Weights are obtained as in the following equation [34]

$$w_{\text{RR}_f} = \frac{\sum\limits_{j \in R} \dfrac{1}{\text{rank}_f(\mathbf{x}_j)}}{\sum\limits_{f' \in F} \sum\limits_{j \in R} \dfrac{1}{\text{rank}_{f'}(\mathbf{x}_j)}} \tag{5.36}$$

and

$$S_{\text{RR}}\left(\mathbf{x}_i\right) = \sum_{f \in F} w_{\text{RR}_f} \cdot S_f\left(\mathbf{x}_i\right) \tag{5.37}$$

where $f \in F$, and R is the set of the relevant images and $rank_f(\mathbf{x})$ is the function that returns the position of the image $\mathbf{x}$ after that all images in the database have been sorted according to the relevance score evaluated in the feature space f.

The other three combination methods are used to combine the distances computed in different feature spaces. One method computes the sum of the normalized distances (SUM), the second method computes the "Nearest-Based" weighted sum (NBW) where the weights are computed in a similar way as in Eq. (5.23):

$$w_f = \frac{\sum\limits_{i\in R}\sum\limits_{j\in R} d_f\left(\mathbf{x}_i,\mathbf{x}_j\right)}{\sum\limits_{i\in R}\sum\limits_{j\in R} d_f\left(\mathbf{x}_i,\mathbf{x}_j\right)+\sum\limits_{i\in R}\sum\limits_{h\in N} d_f\left(\mathbf{x}_i,\mathbf{x}_h\right)} \tag{5.38}$$

where $f\in F$, $d_f(\cdot)$ is a function that returns the distance between two images measured in the feature space f, and R, and N are respectively the set of the relevant and non-relevant images.

The third method is that presented in Section 5.2.1 and the weights are obtained according to Eq. (5.30)

Dissimilarity representation from multi-feature spaces

In order to formalize the proposed technique, let

$$F=\{f_1,\ldots,f_M\} \tag{5.39}$$

be the set of low-level feature spaces extracted from the images, and let

$$d_{f_m}\left(\mathbf{x}_i^{f_m},\mathbf{x}_j^{f_m}\right) \tag{5.40}$$

be the distance between the images $\mathbf{x}_i$ and $\mathbf{x}_j$ evaluated in the feature space f_m. Given a reference image $\mathbf{q}$, the new representation of a generic image $\mathbf{x}_i$ in the *dissimilarity multi-space* is

$$\mathbf{x}_i' = \left[d_{f_1}\left(\mathbf{q}^{f_1},\mathbf{x}_i^{f_1}\right),\ldots,d_{f_M}\left(\mathbf{q}^{f_M},\mathbf{x}_i^{f_M}\right)\right]. \tag{5.41}$$

Summing up, while dissimilarity space are usually constructed by stacking dissimilarities from multiple prototypes, stacking multiple dissimilarities originated by multiple feature representations of the same image has been proposed.

Let me have a close look on the choice of the reference image to be used in Eq. (5.41). When the first round of retrieval is performed, i.e., no feedback is available, the query image as the reference point is used. At each round of relevance feedback, the reference point is computed according to a "query shifting mechanisms", i.e., a mechanism designed to exploit relevance feedback by computing a new query vector in the feature space such that its neighborhood contains relevant images with high probability [82]. In particular, it has been used the Bayes Query Shifting (BQS) approach (see Section 3.1.1).

The choice of the query, and the BQS as the reference prototypes is twofold. First of all, as they have been took into account retrieval tasks in which the user performs a "query by example" search, the query image, and the BQS are aimed to represent the concept that the user is searching for by definition. On the other hand, the use of multiple images as prototypes can introduce some kind of "noise" because not all the images may exhibit the same "degree" of relevance to the user's needs. The second reason is that the use of a single prototype makes the search independent from the number of images in the database that are relevant to the user's query.

In the literature of content-based image retrieval, few works addressed the use of dissimilarity spaces to provide for a more effective representation. Some of the approaches proposed so far employed the original definition of dissimilarity space, where dissimilarities are computed by taking into account multiple prototypes of relevant images [70, 32]. Other authors have proposed to use the “dissimilarity space” technique for combining different feature space representations [12]. However, their approach is based on the computation of dissimilarity relationships between all the patterns in the dataset. Then, a number of prototypes are selected in each feature space, and the resulting dissimilarity spaces are then combined to attain a new multi-modal dissimilarity space. Thus the components of the resulting space are not related to the number of the original feature spaces, but they are related to the number of patterns used as prototypes to create the different dissimilarity spaces.

Nearest-Neighbor Relevance Feedback in the Dissimilarity Multi-Space

The generation of the dissimilarity space is strictly related to the use of a nearest neighbor approach to exploit relevance feedback in multiple feature spaces. In fact, the new space provides for a compact representation of patterns that ease the computation of nearest-neighbor relationships in multiple low-level feature representations. The dissimilarity representation basically assumes that patterns belonging to the same category are typically represented as close points in the space made up of dissimilarities computed with respect to a set of prototypes. Analogously, it is expected that relevant images are represented as close points in the space made up of dissimilarities computed with respect to the query image in multiple low-level feature spaces. The nearest neighbor technique employed is the same used in the other chapters and it is based on the computation of a relevance score for each image. This score is further combined to a score related to the distance of the image from the point computed according to the BQS (Eq. (3.6)), that is the likelihood that the image is relevant according to the users’ feedback. The combined relevance score is computed according to Eq. (3.11).

The dissimilarity multi-space is included in a content-based retrieval system with nearest neighbor relevance-feedback according to the following algorithm :

i) the user submits a query image $\mathbf{q}$. The distances $d_{f_m}\left(\mathbf{q}^{f_m}, \mathbf{x}_i^{f_m}\right)$, $m = 1, \ldots, M$, and $i = 1, \ldots N$ are computed, where M is the number of low-level features used to represent the images, and N is the number of the images in the database;

ii) these distances are used to create, for each image $\mathbf{x}_i$, the new dissimilarity representation $\mathbf{x}'_i$, $i = 1, \ldots N$, according to Eq. (5.41);

iii) the Euclidean distances $d'\left(\mathbf{q}', \mathbf{x}'_i\right)$ between the dissimilarity representation of the query, and the dissimilarity representation of all the images are computed, and then sorted from the smallest to the largest;

iv) the first k images are labelled by the user as being relevant or not;

v) after the relevance feedback, the new query point $\mathbf{q}_{BQS}$ is calculated according to Eq. (3.6) in each feature space;

vi) the distances $d_{f_m}\left(\mathbf{q}_{BQS}^{f_m}, \mathbf{x}_i^{f_m}\right)$ are computed in the low-level feature spaces analogously to steps **i)** and **ii)**, where the query $\mathbf{q}$ is substituted with the new query point $\mathbf{q}_{BQS}$. These distances are used to create a new dissimilarity representation;

vii) in this new space, a score for all the images in the dataset is evaluated according to Eq. (3.11), where all the distances are computed according to the dissimilarity representation;

viii) all the images are sorted according to the value of the relevance score, and the first k images are labelled by the user as in step **iv)**;

ix) the algorithm starts again from step **iv)** until the user is satisfied.

Experimental Results

Datasets

Experiments have been carried out using the Caltech-256 dataset whence five different features have been extracted, namely the *Tamura* features (18 components), the *Scalable Color* (64 components), *Edge Histogram* (80 components), *Color Layout* descriptors (12 components), and the *Color and Edge Directivity Descriptor* (*CEDD*, 144 components). The open source library LIRE (Lucene Image REtrieval) has been used for feature extraction (see Sections 2.5 and 2.2).

Experimental Set-up

In order to test the performances, 500 query images have been randomly extracted from the dataset, covering all the semantic classes. The top twenty best scored images for each query are returned to the user. Relevance feedback is performed by marking images belonging to the same class of the query as relevant, and all other images in the top twenty as non-relevant. Performances are evaluated in terms of retrieval precision, and recall. The first one is evaluated taking in account the top twenty best scored images at each iteration, regardless they have been already labelled by the user. The recall takes into account all the relevant images retrieved so far(see Section 2.4).

In order to evaluate the improvement attained by the proposed method in the next subsection, the attained results will be shown separately in each feature space, and the performance related to the six combination techniques described in Section 5.2.2.

Results

Figures 5.11 and 5.12 show the performance of the proposed dissimilarity space representation compared to the seven combination techniques described in Section 5.2.2, and the performance attained separately in each feature space.

By inspecting the behavior of the precision reported in Figure 5.11, it can be easily seen that the highest performances are provided by the proposed dissimilarity (DS) based technique, and by the MEAN of the relevance scores. The combination of the relevance scores by the MAX and RR Weight rules allows attaining higher precision results than those attained by four out of the five features considered. This result is quite reasonable as typically the goal of the combination is to avoid choosing the worst problem formulation. In addition, it can be seen that the weighted combination (RR Weight) provides a lower result compared to the arithmetic MEAN, thus confirming the difficulty in providing an effective estimation of the weights. Finally, the worst result is attained by the MIN rule, that represents the logical AND function. Thus, it is possible to conclude that, at least for the considered data set, the fusion of information from multiple feature spaces is more effective than the selection of one feature space. In addition, the results attained by the proposed DS space and the MEAN rule confirm that an unweighted combination can be more effective than weighted combination or selection. If the three techniques based on the combination of the distances are considered, namely the SUM rule, the NBW rule, and the *Minimization-Error* rule (MeW), it is possible to see that their performances are lower than those of the techniques based on the combination of scores. In addition, the computational cost of the first two techniques is quite expensive (Fig. 5.13). This behavior can be explained by the objective difficulties in combining distances computed in different feature spaces. On the other hand, the fusion of the relevance scores, or the use of the dissimilarity space approach allows to effectively combine similarity measures computed in different spaces. It is worth to note that the trend shown in Figure 5.11 it is much lower than that presented in Figure 5.10(b), the reason is that in this case the top twenty best scored images at each iteration are evaluated in Section 5.2.1 the precision is evaluated on twenty images taking in account the top best scored images at each iteration and the relevant images retrieved in the previous iterations so it is easier to reach higher performance.

If the time complexity of the considered techniques it is took into account, it can be seen that the speed of the proposed method is also higher than that of individual feature spaces. This effect can be explained by considering that all the distances from each image to the query are computed only once, during the first retrieval iteration. All the following iterations can exploit this result, and all the computation are made in the low-dimensional dissimilarity space.

If the recall it is considered (Figure 5.12), it is possible to see that MEAN rule, and the DS approaches are still the best technique. It is worth noting that all the combination techniques, except for the MIN, provide an improvement in recall with respect to the performance attained in the individual feature spaces. In particular the fusion techniques working at the distance level provided good results, quite close to those attained by the RR Weight rule. Thus the proposed DS technique is effective in combining different feature spaces, both in terms of precision and recall, and in terms of the computational time.

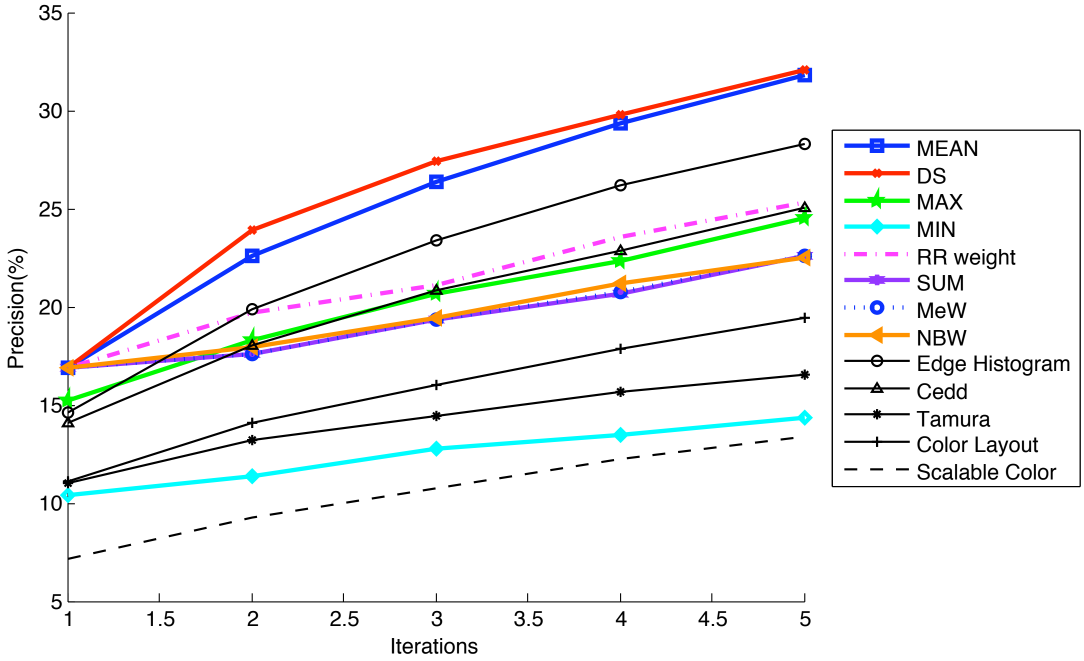


Figure 5.11: Caltech-256 Dataset - Precision for 5 rounds of relevance feedback.

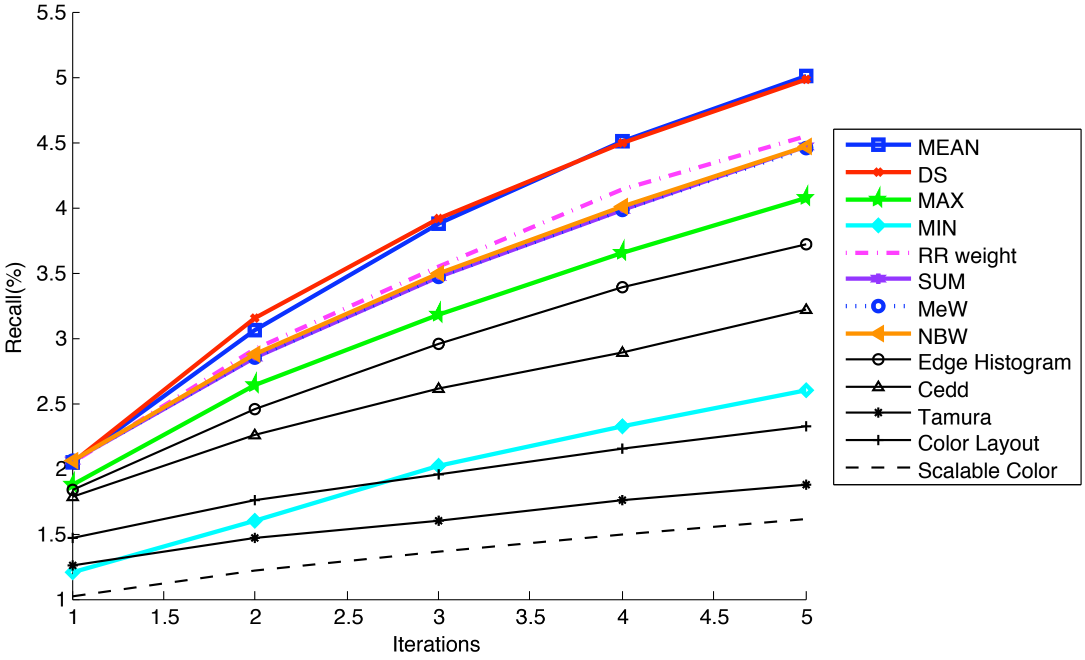


Figure 5.12: Caltech-256 Dataset - Recall for 5 rounds of relevance feedback.

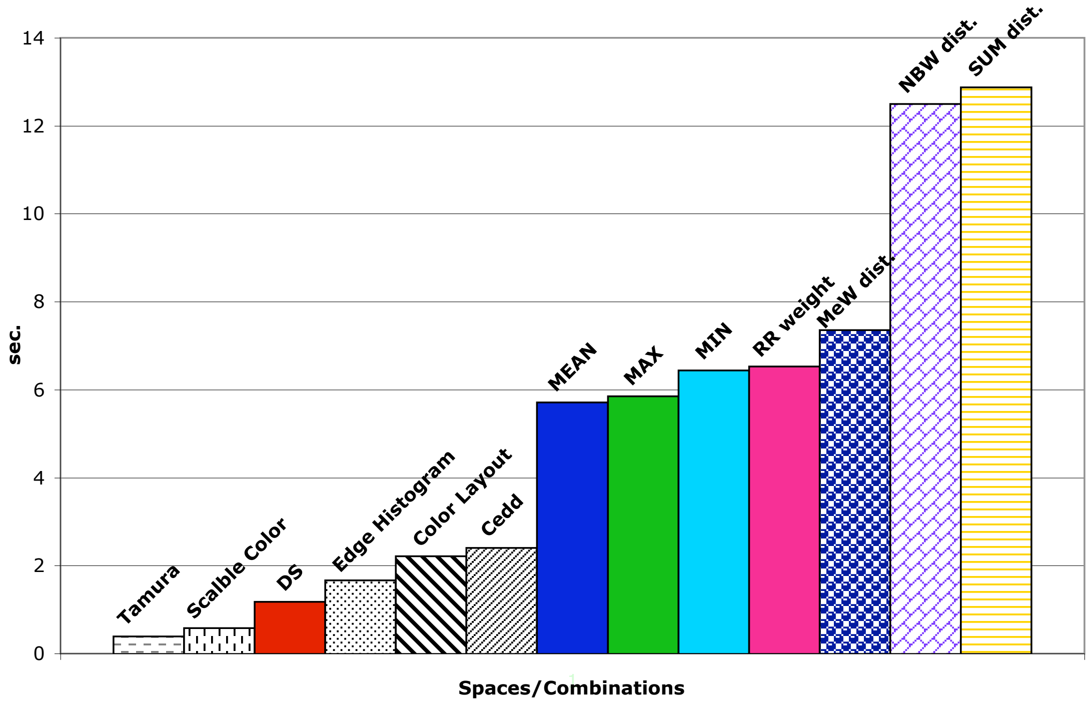


Figure 5.13: Caltech-256 Dataset - Mean execution time for 1 round of relevance feedback.

Chapter 6

Combination of multiple relevance feedback techniques

One of the peculiarity of content based image retrieval is its applicability in a huge number of application, in fact it is possible to use it in photography as in the field of the fashion, to retrieve paintings from a booklet of a picture gallery or a certain kind of images from the net and the number in growing every day. This peculiarity however is also its weak point in fact is not possible find a satisfactory and definite answer to the question:"Given an example, how find the largest number of similar image?"

In a limited domain it is possible to optimize some techniques taking in account the dataset, the type of the images, the distribution of the patterns and the extracted features but is almost impossible find a technique that fits itself to every kind of applications. The aim of this work is combine two very different techniques in order to obtain good results in different datasets and representations rather than high performance in a specific dataset. The two used techniques are the Support Vector Machines [118] and an algorithm Nearest Neighbor based [31]. The reason for the choice of this two techniques is that they work well in different feature space and if used in the same representation space retrieve different images. In order to point out these differences a different kind of measure will be proposed: a *diversity measure*. The introduction of this measure has been necessary in order to understand how the two different systems work and how exploit these differences to build a stronger system.

6.1 Nearest neighbor and SVM scores fusion

In this section a comparison between an SVM classifier with an RBF kernel and a nearest neighbor approach will be proposed. The two methods have been investigated during a content based image retrieval task in which the relevance feedback is applied. In both approaches it has been considered a relevance score obtained by them: the nearest neighbor relevance score has been obtained according to Eq. (3.11), as SVM score has been used the evaluated margin provided by the classifier. In the comparison these scores have been compared with those obtained through different kind of score fusion. In the next section a brief overview of the used fusion method will be given.

6.1.1 Score fusion

In the fusion of the investigated approaches it has been applied the widely used combination rules: *max*, *min*, and *mean*. In addition as further combination technique has been also used a weighted sum where the weights have been evaluated as in [34] (see also Eq. (5.36)). The main difference between the three classic rules and the weighted sum is that in one case the selection is based on how a certain method has classified the relevant images in the previous iteration. In other words taking into account, in a certain sense, its history it aim at predicting the system more effective. The other rules are based on the current relevance score value. The combination rules can be evaluated according to the following equations:

$$S_{\mathrm{MAX}}(\mathbf{x}) = \max\left(S_{\mathrm{NN}}(\mathbf{x}), S_{\mathrm{SVM}}(\mathbf{x})\right); \tag{6.1}$$

$$S_{\min}(\mathbf{x}) = \min\left(S_{\mathrm{NN}}(\mathbf{x}), S_{\mathrm{SVM}}(\mathbf{x})\right); \tag{6.2}$$

$$S_{\mathrm{Mean}}(\mathbf{x}) = \frac{S_{\mathrm{NN}}(\mathbf{x}) + S_{\mathrm{SVM}}(\mathbf{x})}{2}. \tag{6.3}$$

The Relevance Rank score can be formulated as:

$$S_{\mathrm{RR}}(\mathbf{x}) = w_{\mathrm{RR_{NN}}} \cdot S_{\mathrm{NN}}(\mathbf{x}) + w_{\mathrm{RR_{SVM}}} \cdot S_{\mathrm{SVM}}(\mathbf{x}); \tag{6.4}$$

and the weight w_{RR} for a generic approach M as

$$w_{\mathrm{RR_M}} = \frac{\displaystyle\sum_{i=1}^{|R|} \frac{1}{\mathrm{rank}_M(i)}}{\displaystyle\sum_{i=1}^{|R|} \frac{1}{\mathrm{rank}_M(i)} + \sum_{i=1}^{|R|} \frac{1}{\mathrm{rank}_{X'}(i)}} \tag{6.5}$$

with $M \neq M'$ and where R and $|R|$ are the set of the relevant images and its cardinality, respectively.

6.1.2 Diversity measure

In this section will be presented also a measure of diversity that evaluates the percentage of relevant images retrieved by just one of the two methods with respect to the total number of relevant images retrieved by both methods, i.e. it measures the ability of each technique to retrieve different images with respect to those retrieved by the other one. Taking in account this measure is possible to understand if the set of relevant images retrieved by one method overlap the same set retrieved by the other and if it could be more or less convenient combine the two methods. If the diversity is high the combination can work very well because the two algorithms find completely different images, on the contrary if diversity is low the two techniques find the same images and in this case the combination in not useful.

$$\mathscr{D} = \frac{|R_{\mathrm{SVM}} \cup R_{\mathrm{NN}} - R_{\mathrm{SVM}} \cap R_{\mathrm{NN}}|}{|R_{\mathrm{SVM}} \cup R_{\mathrm{NN}}|} \tag{6.6}$$

where R_{SVM} (R_{NN}) is the set of the relevant images retrieved using the SVM (NN) method.

6.1.3 Experimental Results

Datasets

Experiments have been carried out using three datasets, namely Caltech-256 dataset, WANG dataset, and Microsoft Research Cambridge Object Recognition Image Database (MSRC) (see Section 2.5). From Caltech-256 two different kind of features have been extracted, namely the *Edge Histogram* descriptor (80 components), and the *Color and Edge Directivity Descriptor* ($Cedd$, 144 components). The open source library LIRE (Lucene Image REtrieval) has been used for feature extraction. The images from WANG are represented by a 512-dimensional *colour histogram* and a 512-dimensional *Tamura* texture feature histogram concatenated in a unique vector. The images of MSRC instead are represented by a vector of 4096 components of SIFT descriptors extracted at Harris interest points and the *Color and Edge Directivity Descriptor* (see Section 2.2).

Experimental Setup

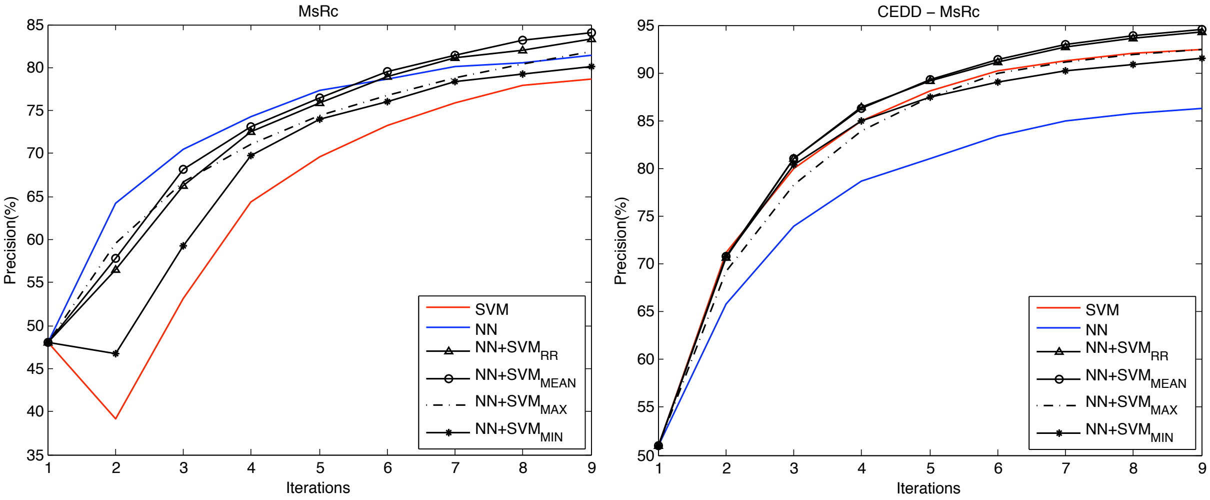
In order to test the performances 500 query images from Caltech-256 dataset have been randomly extracted, for WANG and MSRC datasets each image is used as query. The top twenty best scored images for each query are returned to the user. Performances are evaluated in terms of precision, recall, relative precision and recall, and diversity. The first one is evaluated taking in account the top twenty best scored images at each iteration, regardless they have been already labelled by the user. The recall takes into account all the relevant images retrieved so far (see Section 2.4). The relative precision is the counting of the number of relevant images with respect to the total number of the retrieved images that a method would find if he had worked alone. The relative recall it is analogously evaluate.

It is worth to note that the relative precision and recall have nothing to do with the learning process or with the global performance, they are only a measure of how the two system work in different way and which works better starting from the same "training" set. In fact in each iteration the two system use the same relevant and non relevant images to be learnt (those retrieved by the combination) and at the end of each one, before to combine the score, it is possible to evaluate which images are relevant for each methods. The aim of these measures is to investigate this aspect.

Experiments have been carried out running a fist iteration in which the images are sorted from the nearest to the furthest from the query; after the user feedback in the other iterations the proposed approaches have been used.

Results

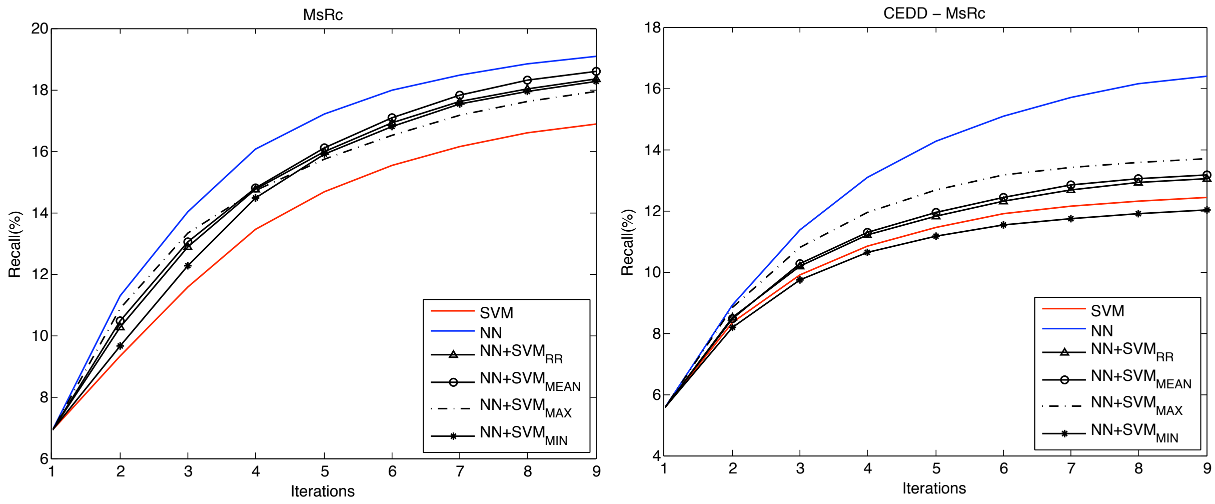
In Figures 6.1 and 6.2 are reported the precision and the recall evaluated on MSRC dataset. It is easy to see from Figures 6.1 that the SVM and the nearest neighbor approach (NN), have a very different behaviour. The SVM works clearly better with CEDD, whereas the NN with the SIFT descriptor. It is possible also to see that the combination in both cases works better than the two methods. If the recall is considered (Figures 6.2) the best performance are reached by the NN but also the combination obtain no so bad result. As regards the relative precision and recall for SIFT descriptor, reported in Figures 6.3, as for precision and recall separately measured for both retrieval techniques, the trend of the NN overcame that of SVM. On the contrary in CEDD feature set, the results of relative recall is a little unexpected (Figures 6.4(b)), in fact contrary to the recall of the methods separately considered, in this case the SVM obtains the best result. In Figure 6.5 it is reported the diversity measure evaluated according to Eq. (6.6) considering both the different combinations (NN + SVM) and the method working separately (NN&SVM). In the fist iteration the value is equal to 0, in fact all the retrieved images are the same for SVM and NN, from the second to, at least, the fourth, instead it is possible to see how the number of images retrieved by just one method is higher when the methods are combined. It is worth to note that this measure does not evaluate if one method works better or worse than another but counts simply how many relevant images are retrieved by just one of the two methods without taking care which technique retrieve more of them. The results obtained con Caltech confirm that the combination is again the best solution if it is considered the precision. In Figures 6.6(a) and 6.6(b) it is also possible to see how the two descriptors obtain different performances according to the different type of approach. CEDD again proves to be more suitable for the SVM, Edge Histogram instead permits a better nearest neighbor approach. As regards the recall (Figures 6.7) both the different kind of combinations and retrieval techniques separately considered work in the same way except for the SVM in EH feature set. This result proves that the combination is robust even if one of the combined techniques obtain very bad results. In Figures 6.8(a) and (b) it is possible to observe the differences between the NN and SVM techniques apart considered the same techniques when they are combined. In Figure 6.6(a) works better the NN approach, in 6.8(a) works better the SVM. This behaviour can be explained taking into account that the SVM need more training patterns than NN, so when the SVM is used in combination with NN, after few iterations it can exploit the higher number of relevant images obtained by the combination. It is worth to note that in the relative precision and recall is not important the absolute value of the performance considered but the difference between two iterations in succession. The diversity is reported in Figure 6.10 and it shows how in Caltech the two methods, separately considered, are able to retrieve a larger number of different images contrary to that happens in MSRC where is the combination that



(a) Precision at 20 picture - SIFT Descriptor

(b) Precision at 20 picture - CED Descriptor

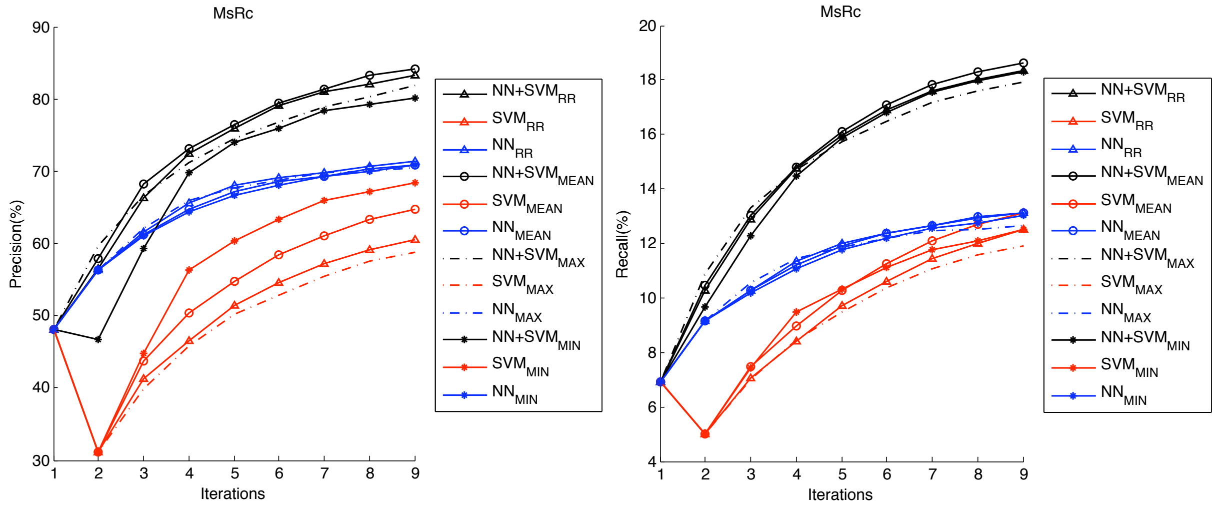
Figure 6.1: MicroSoft Research Dataset - Precision using SIFT and CED Descriptor for 9 rounds of relevance feedback.



(a) Recall - SIFT Descriptor

(b) Recall - CED Descriptor

Figure 6.2: MicroSoft Research Dataset - Recall using SIFT and CED Descriptor for 9 rounds of relevance feedback.

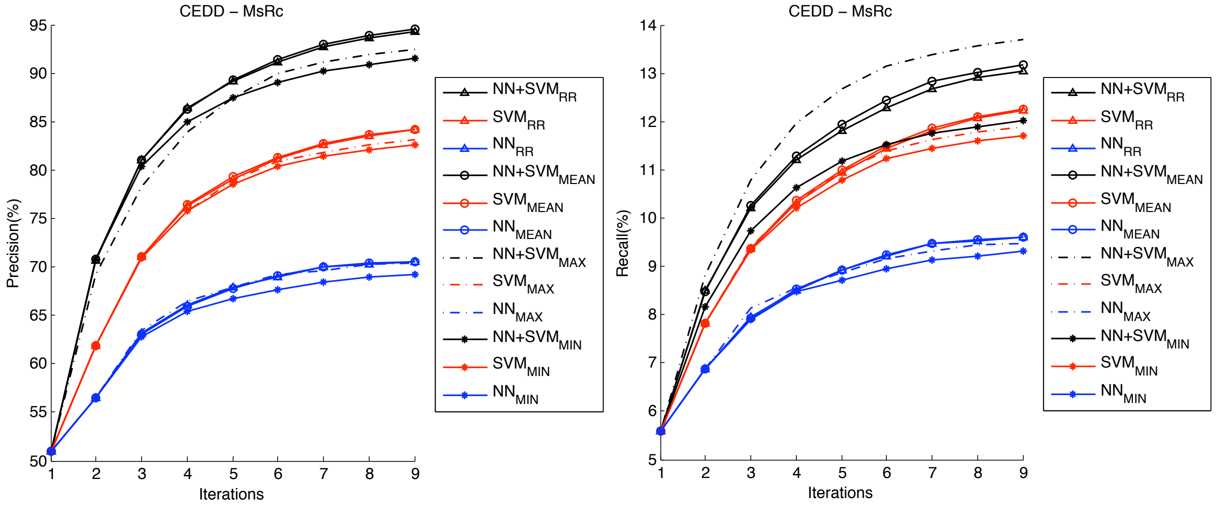


(a) Relative Precision

(b) Relative Recall

Figure 6.3: MicroSoft Research Dataset - Relative Precision and Relative Recall for 9 rounds of relevance feedback using SIFT Descriptor.

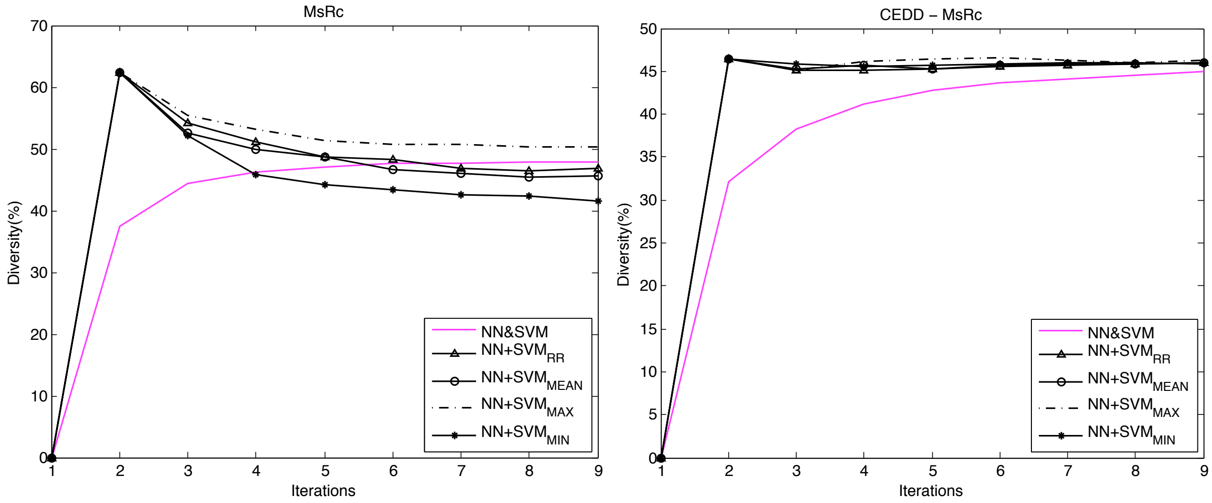
has a higher trend. This result is a little unexpected, but could be explained with the large difference of performance between the SVM and NN when are separately considered. The use of WANG dataset confirm that the use of the combination can be useful also when two kinds of performance, in this case precision and recall (Figure 6.11), obtain different results according to different approaches. From the other graphs it is possible to observe how, again, the SVM in the relative precision and recall after the first iterations increase more quickly than the NN (Figure 6.12) and how the combination retrieves diverse images (Figure 6.13). The proposed results are a selection of a large number of experiments. Besides those that has been shown in this thesis, experiments with *Tamura* features, the *Scalable Color*, *Edge*



(a) Relative Precision

(b) Relative Recall

Figure 6.4: MicroSoft Research Dataset - Relative Precision and Relative Recall for 9 rounds of relevance feedback using CED Descriptor.

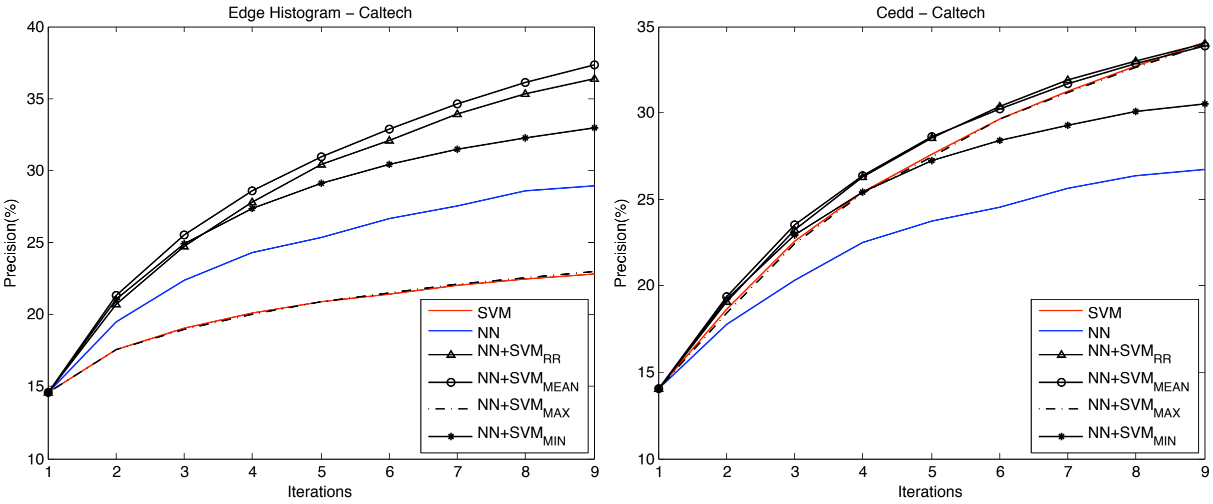


(a) Diversity on SIFT feature set

(b) Diversity on CEDD feature set

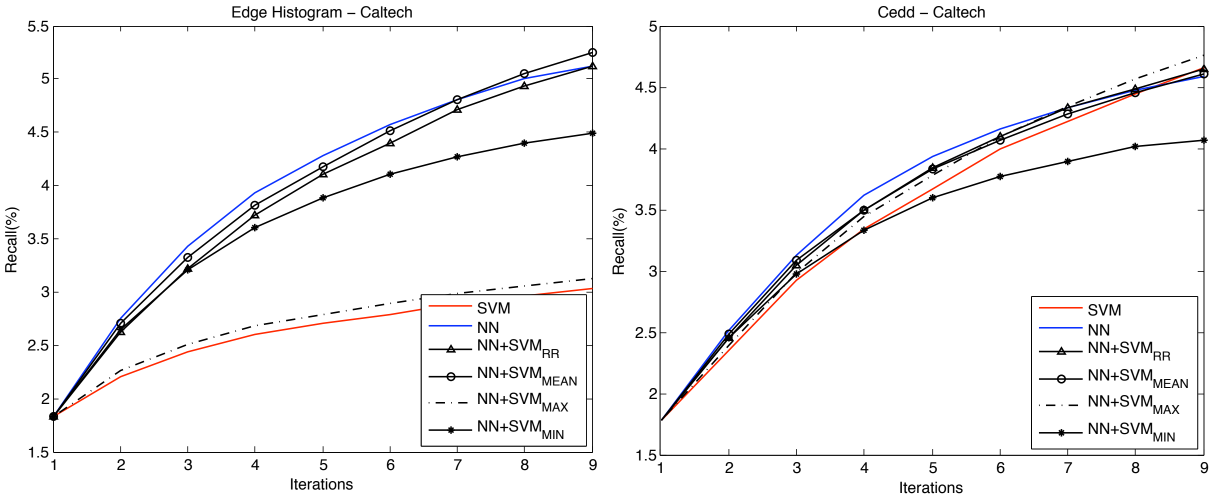
Figure 6.5: MicroSoft Research Dataset - Diversity measure using SIFT and CED Descriptor for 9 rounds of relevance feedback.

Histogram, *Color Layout* descriptors, and the *Color and Edge Directivity Descriptor* has been performed using the MSRC and Caltch dataset but they have not been reported in this dissertation, as they do not provide additional information.



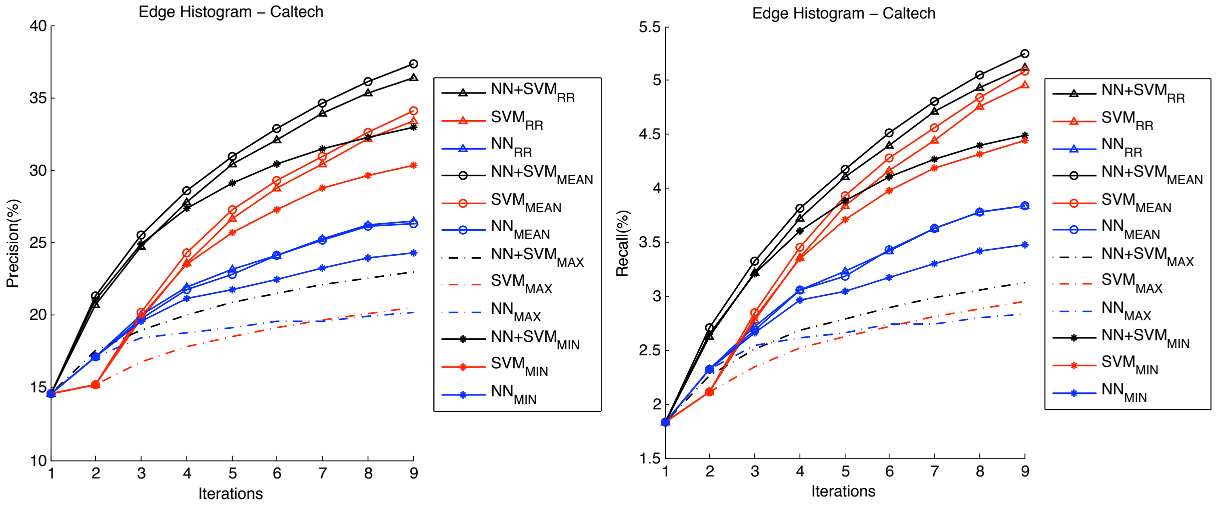
(a) Precision at 20 picture - EH Descriptor (b) Precision at 20 picture - CED Descriptor

Figure 6.6: Caltech-256 Dataset - Precision using Edge Histogram and CED Descriptor for 9 rounds of relevance feedback.



(a) Recall - EH Descriptor (b) Recall - CED Descriptor

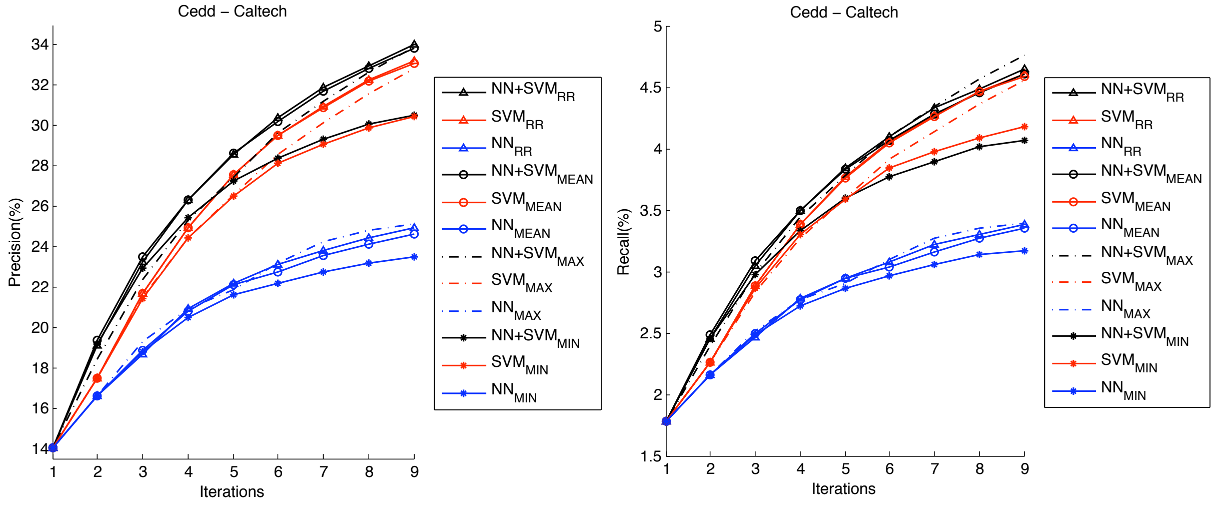
Figure 6.7: Caltech-256 Dataset - Recall using EH and CED Descriptor for 9 rounds of relevance feedback.



(a) Relative Precision

(b) Relative Recall

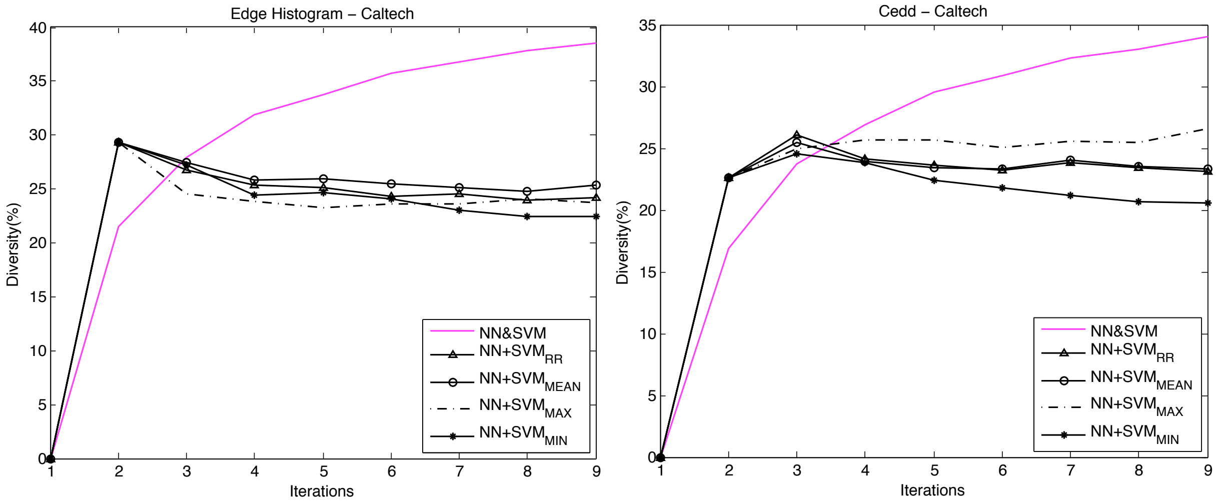
Figure 6.8: Caltech-256 Dataset - Relative Precision and Relative Recall for 9 rounds of relevance feedback using EH Descriptor.



(a) Relative Precision

(b) Relative Recall

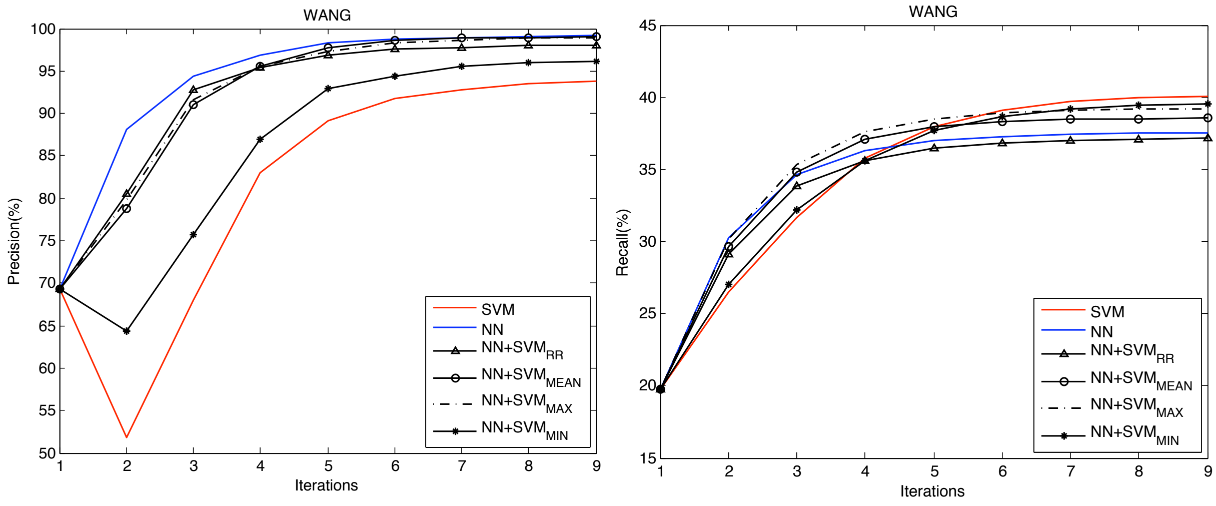
Figure 6.9: Caltech-256 Dataset - Relative Precision and Relative Recall for 9 rounds of relevance feedback using CED Descriptor.



(a) Diversity on EH feature set

(b) Diversity on CEDD feature set

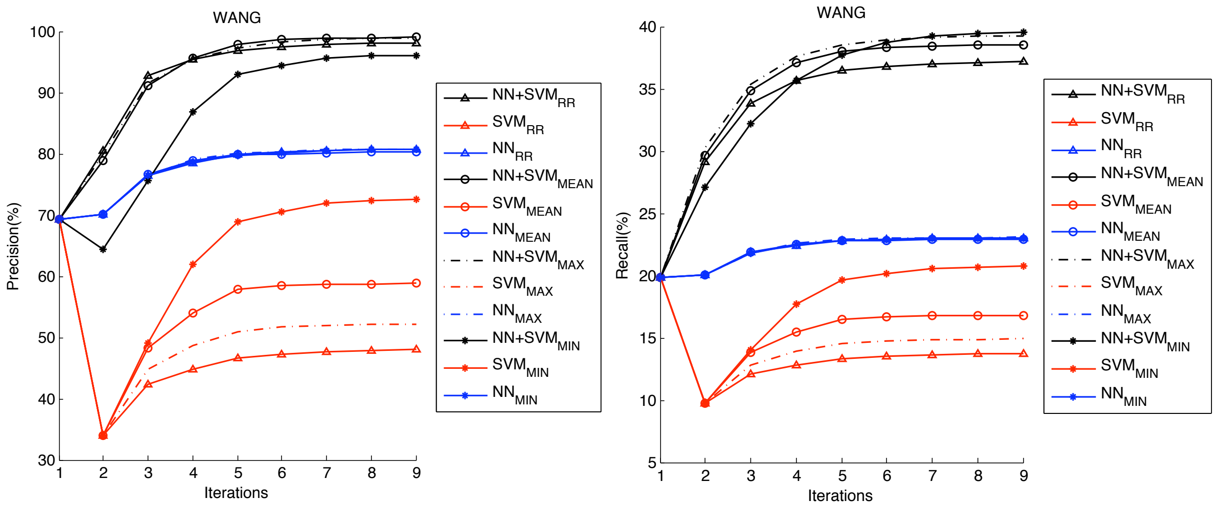
Figure 6.10: Caltech-256 Dataset - Diversity measure using EH and CED Descriptor for 9 rounds of relevance feedback.



(a) Precision at 20 picture

(b) Recall

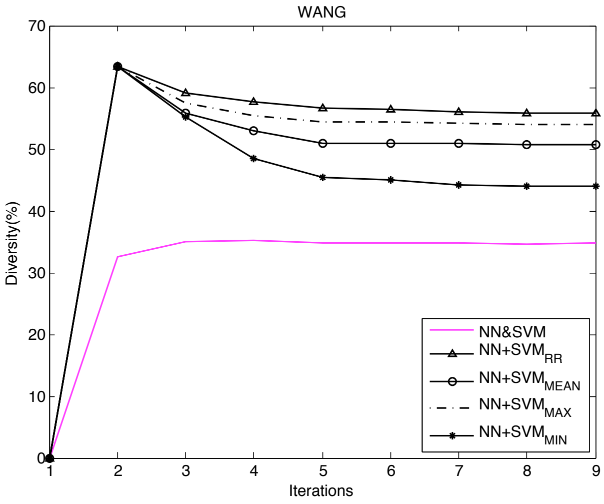
Figure 6.11: WANG Dataset - Precision and Recall for 9 rounds of relevance feedback.



(a) Relative Precision

(b) Relative Recall

Figure 6.12: WANG Dataset - Relative Precision and Relative Recall for 9 rounds of relevance feedback.



(a) Diversity measure

Figure 6.13: WANG Dataset - Diversity measure for 9 rounds of relevance feedback.

Chapter 7

Conclusions and Future Works

Nowadays very large archives of digital images are easily produced thanks to the wide availability of digital cameras, that are often embedded into a number of portable devices. Each personal computer, as well as photo-sharing and social-network web sites, are rapidly becoming the repository for thousands, or even billions of images. As a consequence there is an increasing need for tools enabling the semantic search, classification, and retrieval of images. The use of meta-data associated to the images solves the problems only partly, as the process of assigning meta data to images is not trivial, is slow, and closely related to the persons who performed the task. A solution that is imposing itself is the content based search, i.e. based on intrinsic characteristics of images, such as color distribution, detection of edges and objects, etc.. Through these characteristics, defined as low-level feature, the content based retrieval systems evaluates the similarities between the images, submitted by the user as query, and all those in the database, finding the most alike ones. The search is then, refined by the system by means of feedbacks of the user that judges as relevant or non relevant the images that the system retrieves iteration by iteration. Even if the retrieval systems have been investigated since almost twenty years, they still suffer some not fully resolved problems as *unbalanced learning, high dimensional feature space, low informative training set* and *low discriminative capability of the feature set* problems.

7.1 Where is the content based image retrieval now?

In this thesis some of the problems that mainly plague the content based retrieval field have been faced, the relevance feedback demonstrated that it can help to solve some of them but it is not enough. First of all, it is necessary to ask ourself a question: *What am I looking for? What do I expect to find?* As it has been mentioned in Chapter 2 it is possible to distinguish different levels of research and as it is possible to imagine that not everyone can be performed in the same way. Relevance feedback techniques certainly allows to perform searches at Level 1, where the comparisons between the images can be made considering the intrinsic features of an image. It is possible to find objects, to distinguish different landscapes and even, through face recognition techniques, it is possible also to distinguish different people in an image.

On the other hand, what is still very difficult to do is perform searches at the Level 2 and 3, where concepts that go beyond what is physically represented are required. Two clear examples are the Figure 2.4 and some classes of Caltech dataset. In the first case even though the two figures are on the same background and the color and shapes characteristics are very similar, nevertheless they represent completely different concepts. In the case of Caltech, on the contrary, Figure 2.13 shows examples of images that are very different from each other but that represent the same concept. It goes without saying that such research can not be based solely on the content. At the present situation, the relevance feedback and the CBIR in general, are certainly helpful in searches carried out in a restricted domain, for example searches of monuments within a regional or national digital library or searches of clothing in a fashion mail-order catalogue. In these fields one finds oneself dealing with large datasets that although large they are predetermined. In this scenario, the techniques of *Directed Pattern Injection*, *Dominant Set* and *Locally Linear Embedding*, as well as *Exploitation-Exploration* certainly offer a great help in improving search and in relating to each other the similar images.

In higher semantic levels searches or in those performed on the web the information based only on low-level features is definitely not enough. It is necessary also to derive more knowledge from the context from whom the image is extrapolated. This type of investigation, inevitably, involves the combination of information that may be the most disparate: users' tag, text descriptions, web site of origin of the image, or also image meta-data as the time when it was taken or even GPS position. The combination of such diverse data is a task that must be handled with care. Methods based on *weighted combination* or based on *(dis)similarity* as well as the *combination of multiple classifiers* can certainly play a key role in this context. Given the great variability of the images on the web, surely also the different approaches have their own influence, very adaptable methods such as those based on on-line learning, certainly can definitely give a good contribution.

The fact that "pure" relevance feedback techniques are more suitable for searches on a bounded domain it does not necessarily imply a limited number of applications where this type of searches can be performed. In addition to purely private uses such as, for example, the holiday photos cataloguing (see Chapter 2), even in professional field the possible uses of these techniques are very numerous. The early recognition and the blockage of pornographic images is a problem now increasingly felt, especially by the social networks that, for obvious reasons, can not manually verify all published material. Even in the legal / judicial field, the analysis of videos and digital photos is a particularly challenging task, a pre-selection of desired information (objects, faces, people, ...) can be a great help to the police inspectors. At the end of this study it is my opinion that although the relevance feedback does not enjoy so much success as several years ago it still has potentials to be expressed. Clearly it has to be used in appropriate contexts, taking account of its inherent limitations.

7.2 Where are we going?

The reported results show how facing the problems discussed in this thesis, even though one at a time, can improve the performance of the retrieval systems, in addition the different kind of combination illustrated demonstrate also that the combination of different approach, different feature set, and different classifiers can produce more improvements. With this in mind the next step in the research activity will be the combination of different approaches in order to solve more problems at the same time.

In addition to dealing with solutions to improve well known techniques in the field of retrieval systems, certainly the research should also take care of adapting the traditional systems to innovations that are appearing on the way. In recent years the Internet communities have got a footing and with them a world of information sharing systems. In the field of information retrieval this exchange of data can be a valuable instrument if used properly. For some time the recommendation systems have become popular in the way of advertising and sales on the Internet. The simple concept that: if you and your *"friends"* share the same interests then it is probably that you are interested in the same things, is the principle upon which is based the modern form of long-term learning. The use of social networks, from this point of view, is certainly a field to be explored. The sharing of photos, videos and links in general, for example, is an implicit form of expressing own opinion of relevance, as well as the famous button "Like", in the social network facebook, is an explicit assessment of the considered resource. The interactivity of these platforms from the perspective the information retrieval researchers, is surely their main strong point. In these communities occurs spontaneously exactly what a user is not normally inclined to do: tagging, labelling and commenting. Certainly to be able to intercept these opinions from the users, can find new outlets for relevance feedback that is just born to to meet the needs of users.

Although the content based image retrieval may seem a difficult challenge, today facebook receives some 415,000 video uploads per day, with 155,000 of them directly from webcams. In the past the researchers tried to borrow the techniques used the image retrieval field for the video but it soon became clear that it was not enough. The new frontier for multimedia retrieval is the search by the concept. Covering semantic space from different perspectives it appears, at the moment, to be the only method by handle the huge amount of data.

Bibliography

[1] Information technology - Multimedia content description interface - Part 3: Visual, ISO/IEC Std. 15938-3:2003, 2003. [cited at p. 11, 12, 15]

[2] Gaurav Aggarwal, T. V. Ashwin, and Sugata Ghosal. An image retrieval system with automatic query modification. *IEEE Transactions on Multimedia*, 4(2):201–214, 2002. [cited at p. 27]

[3] David W. Aha, Dennis Kibler, and Marc K. Albert. Instance-based learning algorithms. *Machine Learning*, 6(1):37–66, 1991. [cited at p. 25]

[4] Miguel Arevalillo-Herráez, Juan Domingo, and Francesc J. Ferri. Combining similarity measures in content-based image retrieval. *Pattern Recognition Letters*, 29(16):2174–2181, 2008. [cited at p. 33]

[5] Annalisa Barla, Emanuele Franceschi, Francesca Odone, and Alessandro Verri. Image kernels. In Seong-Whan Lee and Alessandro Verri, editors, *SVM*, volume 2388 of *Lecture Notes in Computer Science*, pages 83–96. Springer, 2002. [cited at p. 50, 51]

[6] Mikhail Belkin and Partha Niyogi. Laplacian eigenmaps and spectral techniques for embedding and clustering. In Thomas G. Dietterich, Suzanna Becker, and Zoubin Ghahramani, editors, *NIPS*, pages 585–591. MIT Press, 2001. [cited at p. 28]

[7] Wei Bian and Dacheng Tao. Biased discriminant euclidean embedding for content-based image retrieval. *IEEE Transactions on Image Processing*, 19(2):545–554, 2010. [cited at p. 10, 28]

[8] Alberto Del Bimbo. *Visual information retrieval*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1999. [cited at p. 10]

[9] Christopher M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer, October 2006. [cited at p. 26, 47]

[10] Christian Böhm, Stefan Berchtold, and Daniel A. Keim. Searching in high-dimensional spaces: Index structures for improving the performance of multimedia databases. *ACM Comput. Surv.*, 33(3):322–373, 2001. [cited at p. 27]

[11] Sabri Boughorbel, Jean-Philippe Tarel, and Nozha Boujemaa. Generalized histogram intersection kernel for image recognition. In *ICIP (3)*, pages 161–164, 2005. [cited at p. 50, 51]

[12] Eric Bruno, Nicolas Moënne-Loccoz, and Stéphane Marchand-Maillet. Learning user queries in multimodal dissimilarity spaces. In Marcin Detyniecki, Joemon M. Jose, Andreas Nürnberger, and C. J. van Rijsbergen, editors, *Adaptive Multimedia Retrieval*, volume 3877 of *Lecture Notes in Computer Science*, pages 168–179. Springer, 2005. [cited at p. 81]

[13] Savvas A. Chatzichristofis and Yiannis S. Boutalis. Cedd: Color and edge directivity descriptor: A compact descriptor for image indexing and retrieval. In Antonios Gasteratos, Markus Vincze, and John K. Tsotsos, editors, *ICVS*, volume 5008 of *Lecture Notes in Computer Science*, pages 312–322. Springer, 2008. [cited at p. 12, 15]

[14] Nitesh V. Chawla, Kevin W. Bowyer, Lawrence O. Hall, and W. Philip Kegelmeyer. Smote: Synthetic minority over-sampling technique. *J. Artif. Intell. Res. (JAIR)*, 16:321–357, 2002. [cited at p. 27, 38]

[15] Yanhua Chen, Ming Dong, and Wanggen Wan. Image co-clustering with multi-modality features and user feedbacks. In Wen Gao, Yong Rui, Alan Hanjalic, Changsheng Xu, Eckehard G. Steinbach, Abdulmotaleb El-Saddik, and Michelle X. Zhou, editors, *ACM Multimedia*, pages 689–692. ACM, 2009. [cited at p. 30]

[16] Yunqiang Chen, Xiang Sean Zhou, and T.S. Huang. One-class svm for learning in image retrieval. In *ICIP*, volume 1, pages 34 –37 vol.1, 2001. [cited at p. 25]

[17] Jian Cheng and Kongqiao Wang. Active learning for image retrieval with co-svm. *Pattern Recognition*, 40(1):330–334, 2007. [cited at p. 31]

[18] Zheru Chi, Hong Yan, and Tuan Pham. *Fuzzy Algorithms: With Applications to image processing and pattern recognition*, volume 10 of *Advances in Fuzzy Systems-Applications and Theory*. World Scientific Pub Co Inc, 1996. [cited at p. 15]

[19] David A. Cohn, Les E. Atlas, and Richard E. Ladner. Improving generalization with active learning. *Machine Learning*, 15(2):201–221, 1994. [cited at p. 31]

[20] Ingemar J. Cox, Matthew L. Miller, Thomas P. Minka, Thomas V. Papathomas, and Peter N. Yianilos. The bayesian image retrieval system, PicHunter: theory, implementation, and psychophysical experiments. *IEEE Transactions on Image Processing*, 9(1):20–37, 2000. [cited at p. 32]

[21] Koby Crammer and Gal Chechik. A needle in a haystack: local one-class optimization. In Carla E. Brodley, editor, *ICML*, volume 69 of *ACM International Conference Proceeding Series*. ACM, 2004. [cited at p. 32]

[22] Koby Crammer, Ofer Dekel, Joseph Keshet, Shai Shalev-Shwartz, and Yoram Singer. Online passive-aggressive algorithms. *J. Mach. Learn. Res.*, 7:551–585, 2006. [cited at p. 47, 48, 50]

[23] Nello Cristianini and John Shawe-Taylor. *An Introduction to Support Vector Machines and Other Kernel-based Learning Methods*. Cambridge University Press, 2000. [cited at p. 24, 25, 50]

[24] Ritendra Datta, Dhiraj Joshi, Jia Li, and James Z. Wang. Image retrieval: Ideas, influences, and trends of the new age. *ACM Computing Surveys*, 40(2):1–60, 2008. [cited at p. 1, 21]

[25] Thomas Deselaers, Daniel Keysers, and Hermann Ney. Features for image retrieval: an experimental comparison. *Inf. Retr.*, 11(2):77–107, 2008. [cited at p. 10, 13]

[26] Inderjit S. Dhillon. Co-clustering documents and words using bipartite spectral graph partitioning. In *KDD*, pages 269–274, 2001. [cited at p. 30]

[27] Gyuri Dorkó. *Selection of Discriminative Regions and Local Descriptors for Generic Object Class Recognition*. PhD thesis, Institut National Polytechnique de Grenoble, 2006. [cited at p. 13]

[28] Richard O. Duda, Peter E. Hart, and David G. Stork. *Pattern Classification*. John Wiley and Sons, Inc., New York, 2001. [cited at p. 25, 29, 30]

[29] John P. Eakins. Automatic image content retrieval - are we getting anywhere? In *ELVIRA3, Proc. of Third International Conference on Electronic Library and Visual Information Research*, pages 123–135, 1996. [cited at p. 9]

[30] Martin Ester, Hans-Peter Kriegel, Jörg Sander, and Xiaowei Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. In *KDD*, pages 226–231, 1996. [cited at p. 30]

[31] Giorgio Giacinto. A nearest-neighbor approach to relevance feedback in content based image retrieval. In *CIVR '07: Proceedings of the 6th ACM international conference on Image and video retrieval*, pages 456–463, New York, NY, USA, 2007. ACM. [cited at p. 22, 25, 40, 52, 66, 69, 75, 87]

[32] Giorgio Giacinto and Fabio Roli. Dissimilarity representation of images for relevance feedback in content-based image retrieval. In Petra Perner and Azriel Rosenfeld, editors, *MLDM*, volume 2734 of *Lecture Notes in Computer Science*, pages 202–214. Springer, 2003. [cited at p. 26, 81]

[33] Giorgio Giacinto and Fabio Roli. Bayesian relevance feedback for content-based image retrieval. *Pattern Recognition*, 37(7):1499–1508, 2004. [cited at p. 23, 24, 25, 39]

[34] Giorgio Giacinto and Fabio Roli. Nearest-prototype relevance feedback for content based image retrieval. In *ICPR (2)*, pages 989–992, 2004. [cited at p. 79, 88]

[35] Philippe Henri Gosselin and Matthieu Cord. Active learning methods for interactive image retrieval. *IEEE Transactions on Image Processing*, 17(7):1200–1211, 2008. [cited at p. 31]

[36] David Grangier and Samy Bengio. A discriminative kernel-based approach to rank images from text queries. *IEEE Trans. Pattern Anal. Mach. Intell.*, 30(8):1371–1384, 2008. [cited at p. 47, 48]

[37] Gregory Griffin, Alex Holub, and Pietro Perona. Caltech-256 object category dataset. Technical Report 7694, California Institute of Technology, 2007. [cited at p. 18]

[38] Venkat N. Gudivada and Vijay V. Raghavan. Content-based image retrieval systems - guest editors' introduction. *IEEE Computer*, 28(9):18–22, 1995. [cited at p. 9]

[39] Robert M. Haralick, K. Shanmugam, and Its'Hak Dinstein. Textural features for image classification. *Systems, Man and Cybernetics, IEEE Transactions on*, 3(6):610 –621, nov. 1973. [cited at p. 12]

[40] Chris Harris and Mike Stephens. A combined corner and edge detector. In *Alvey Vision Conference*, page 147Ð152, 1988. [cited at p. 13]

[41] Haibo He and E.A. Garcia. Learning from imbalanced data. *Knowledge and Data Engineering, IEEE Transactions on*, 21(9):1263 –1284, sep. 2009. [cited at p. 26]

[42] Xiaofei He. Incremental semi-supervised subspace learning for image retrieval. In Schulzrinne et al. [88], pages 2–8. [cited at p. 28]

[43] Xiaofei He, Deng Cai, and Jiawei Han. Learning a maximum margin subspace for image retrieval. *IEEE Trans. Knowl. Data Eng.*, 20(2):189–201, 2008. [cited at p. 28]

[44] Xiaofei He, Wei-Ying Ma, and HongJiang Zhang. Learning an image manifold for retrieval. In Schulzrinne et al. [88], pages 17–23. [cited at p. 28]

[45] Xiaofei He and Partha Niyogi. Locality preserving projections. In Sebastian Thrun, Lawrence K. Saul, and Bernhard Schölkopf, editors, *NIPS*. MIT Press, 2003. [cited at p. 28]

[46] Mark Herbster. Learning additive models online with fast evaluating kernels. In David P. Helmbold and Bob Williamson, editors, *COLT/EuroCOLT*, volume 2111 of *Lecture Notes in Computer Science*, pages 444–460. Springer, 2001. [cited at p. 47]

[47] Chu-Hong Hoi and Michael R. Lyu. Group-based relevance feedback with support vector machine ensembles. In *ICPR (3)*, pages 874–877, 2004. [cited at p. 32]

[48] Steven C. H. Hoi, Rong Jin, Jianke Zhu, and Michael R. Lyu. Semisupervised svm batch mode active learning with applications to image retrieval. *ACM Trans. Inf. Syst.*, 27(3):16:1–16:29, 2009. [cited at p. 32]

[49] Steven C. H. Hoi and Michael R. Lyu. A semi-supervised active learning framework for image retrieval. In *CVPR (2)*, pages 302–309. IEEE Computer Society, 2005. [cited at p. 31]

[50] Derek Hoiem, Rahul Sukthankar, Henry Schneiderman, and Larry Huston. Object-based image retrieval using the statistical structure of images. In *CVPR (2)*, pages 490–497, 2004. [cited at p. 27]

[51] Dionysius P. Huijsmans and Nicu Sebe. How to complete performance graphs in content-based image retrieval: Add generality and normalize scope. *IEEE Trans. Pattern Anal. Mach. Intell.*, 27(2):245–251, 2005. [cited at p. 16]

[52] Mark J. Huiskes and Michael S. Lew. The MIR flickr retrieval evaluation. In Michael S. Lew, Alberto Del Bimbo, and Erwin M. Bakker, editors, *Multimedia Information Retrieval*, pages 39–43. ACM, 2008. [cited at p. 19]

[53] Mark J. Huiskes, Bart Thomee, and Michael S. Lew. New trends and ideas in visual concept detection: the mir flickr retrieval evaluation initiative. In James Ze Wang, Nozha Boujemaa, Nuria Oliver Ramirez, and Apostol Natsev, editors, *Multimedia Information Retrieval*, pages 527–536. ACM, 2010. [cited at p. 19]

[54] Anil K. Jain and Richard C. Dubes. *Algorithms for clustering data*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1988. [cited at p. 30]

[55] Prateek Jain and Ashish Kapoor. Active learning for large multi-class problems. In *CVPR*, pages 762–769. IEEE, 2009. [cited at p. 32]

[56] Jerzy W. Jaromczyk and Godfried T. Toussaint. Relative neighborhood graphs and their relatives. In *Proc. IEEE*, volume 80, pages 1502–1517, September 1992. [cited at p. 23, 30]

[57] Feng Jing, Mingjing Li, HongJiang Zhang, and Bo Zhang. Entropy-based active learning with support vector machines for content-based image retrieval. In *ICME*, pages 85–88. IEEE, 2004. [cited at p. 31]

[58] Feng Jing, Mingjing Li, Lei Zhang, HongJiang Zhang, and Bo Zhang. Learning in region-based image retrieval. In Erwin M. Bakker, Thomas S. Huang, Michael S. Lew, Nicu Sebe, and Xiang Sean Zhou, editors, *CIVR*, volume 2728 of *Lecture Notes in Computer Science*, pages 206–215. Springer, 2003. [cited at p. 27]

[59] Eiji Kasutani and Akio Yamada. The mpeg-7 color layout descriptor: a compact image feature description for high-speed image/video segment retrieval. In *ICIP (1)*, pages 674–677, 2001. [cited at p. 15]

[60] Toshikazu Kato. Database architecture for content-based image retrieval. In *Image Storage and Retrieval Systems (SPIE)*, volume 1662, pages 112–123, 1992. [cited at p. 9]

[61] Mohammed Lamine Kherfi and Djemel Ziou. Relevance feedback for cbir: a new approach based on probabilistic feature weighting with positive and negative examples. *IEEE Transactions on Image Processing*, 15(4):1017–1030, 2006. [cited at p. 9, 33]

[62] Michael S. Lew, Nicu Sebe, Chabane Djeraba, and Ramesh Jain. Content-based multimedia information retrieval: State of the art and challenges. *ACM Trans. Multimedia Comput. Commun. Appl.*, 2(1):1–19, 2006. [cited at p. 1, 21, 39]

[63] Yen-Yu Lin, Tyng-Luh Liu, and Hwann-Tzong Chen. Semantic manifold learning for image retrieval. In HongJiang Zhang, Tat-Seng Chua, Ralf Steinmetz, Mohan S. Kankanhalli, and Lynn Wilcox, editors, *ACM Multimedia*, pages 249–258. ACM, 2005. [cited at p. 28]

[64] Michael Lindenbaum, Shaul Markovitch, and Dmitry Rusakov. Selective sampling for nearest neighbor classifiers. *Machine Learning*, 54(2):125–152, 2004. [cited at p. 32]

[65] David G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004. [cited at p. 13]

[66] Mathias Lux and Savvas A. Chatzichristofis. Lire: lucene image retrieval: an extensible java cbir library. In *MM '08: Proceeding of the 16th ACM international conference on Multimedia*, pages 1085–1088, New York, NY, USA, 2008. ACM. [cited at p. 40, 51, 75]

[67] Krystian Mikolajczyk and Cordelia Schmid. A performance evaluation of local descriptors. *IEEE Trans. Pattern Anal. Mach. Intell.*, 27(10):1615–1630, 2005. [cited at p. 13]

[68] Pierre Moreels and Pietro Perona. Evaluation of features detectors and descriptors based on 3d objects. *International Journal of Computer Vision*, 73(3):263–284, 2007. [cited at p. 13]

[69] Henning Müller, Wolfgang Müller 0002, David Squire, Stéphane Marchand-Maillet, and Thierry Pun. Performance evaluation in content-based image retrieval: overview and proposals. *Pattern Recognition Letters*, 22(5):593–601, 2001. [cited at p. 15]

[70] Giang P. Nguyen, Marcel Worring, and Arnold W. M. Smeulders. Similarity learning via dissimilarity space in cbir. In James Ze Wang, Nozha Boujemaa, and Yixin Chen, editors, *Multimedia Information Retrieval*, pages 107–116. ACM, 2006. [cited at p. 26, 81]

[71] Roberto Paredes, Adrian Ulges, and Thomas Breuel. Fast discriminative linear models for scalable video tagging. *Machine Learning and Applications, Fourth International Conference on*, 0:571–576, 2009. [cited at p. 47]

[72] Roberto Paredes and Enrique Vidal. Learning weighted metrics to minimize nearest-neighbor classification error. *IEEE Trans. Pattern Anal. Mach. Intell.*, 28(7):1100–1110, 2006. [cited at p. 73]

[73] Massimiliano Pavan and Marcello Pelillo. Dominant sets and pairwise clustering. *IEEE Trans. Pattern Anal. Mach. Intell.*, 29(1):167–172, 2007. [cited at p. 31, 59]

[74] Elzbieta Pekalska and Robert P. W. Duin. *The Dissimilarity Representation for Pattern Recognition: Foundations And Applications (Machine Perception and Artificial Intelligence)*. World Scientific Publishing Co., Inc., River Edge, NJ, USA, 2005. [cited at p. 26, 63, 78, 79]

[75] Jing Peng, Bir Bhanu, and Shan Qing. Probabilistic feature relevance learning for content-based image retrieval. *Computer Vision and Image Understanding*, 75(1/2):150–164, July/August 1999. [cited at p. 29, 32, 66]

[76] Luca Piras and Giorgio Giacinto. Neighborhood-based feature weighting for relevance feedback in content-based retrieval. In *WIAMIS*, pages 238–241. IEEE Computer Society, 2009. [cited at p. 33, 66]

[77] Luca Piras and Giorgio Giacinto. K-nearest neighbors directed synthetic images injection. In *Image Analysis for Multimedia Interactive Services (WIAMIS), 2010 11th International Workshop on*, pages 1–4, 2010. [cited at p. 36]

[78] Luca Piras and Giorgio Giacinto. Unbalanced learning in content-based image classification and retrieval. In *ICME*, pages 36–41. IEEE, 2010. [cited at p. 35]

[79] Yong Rao, Padmavathi Mundur, and Yelena Yesha. Fuzzy svm ensembles for relevance feedback in image retrieval. In Hari Sundaram, Milind R. Naphade, John R. Smith, and Yong Rui, editors, *CIVR*, volume 4071 of *Lecture Notes in Computer Science*, pages 350–359. Springer, 2006. [cited at p. 32]

[80] J. J. Rocchio. Relevance feedback in information retrieval. In Gerard Salton, editor, *The SMART Retrieval System - Experiments in Automatic Document Processing*, pages 313–323. Prentice Hall, Englewood, Cliffs, New Jersey, 1971. [cited at p. 23, 32]

[81] Sam T. Roweis and Lawrence K. Saul. Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290:2323–2326, December 2000. [cited at p. 27, 28, 36, 61, 62]

[82] Yong Rui and Thomas S. Huang. Relevance feedback techniques in image retrieval. In *Lew M.S. (ed.): Principles of Visual Information Retrieval*, pages 219–258, Springer-Verlag, London, 2001. [cited at p. 21, 33, 80]

[83] Yong Rui, Thomas S. Huang, and Sharad Mehrotra. Content-Based image retrieval with relevance feedback in MARS. In *International Conference on Image Processing Proceedings*, pages 815–818, October 1997. [cited at p. 23, 29, 32, 65]

[84] Yong Rui, Thomas S. Huang, and Sharad Mehrotra. Relevance feedback: A power tool in interactive content-based image retrieval. *IEEE Trans. on Circuits and Systems for Video Technology*, 8(5):644–655, September 1998. [cited at p. 64]

[85] Hichem Sahbi, Jean-Yves Audibert, and Renaud Keriven. Graph-cut transducers for relevance feedback in content based image retrieval. In *ICCV*, pages 1–8. IEEE, 2007. [cited at p. 31]

[86] Simone Santini and Ramesh Jain. Similarity measures. *IEEE Trans. Pattern Anal. Mach. Intell.*, 21(9):871–883, 1999. [cited at p. 13, 26]

[87] Bernhard Scholkopf and Alexander J. Smola. *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond.* MIT Press, Cambridge, MA, USA, 2001. [cited at p. 25]

[88] Henning Schulzrinne, Nevenka Dimitrova, Martina Angela Sasse, Sue B. Moon, and Rainer Lienhart, editors. *Proceedings of the 12th ACM International Conference on Multimedia, October 10-16, 2004, New York, NY, USA.* ACM, 2004. [cited at p. 103]

[89] H. Sebastian Seung and Daniel D. Lee. The manifold ways of perception. *Science*, 290:2268–2269, December 2000. [cited at p. 27]

[90] Jianbo Shi and Jitendra Malik. Normalized cuts and image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 22(8):888–905, 2000. [cited at p. 30]

[91] Rongjie Shi, I-Fan Shen, and Su Yang. Supervised graph-theoretic clustering. In *Neural Networks and Brain, 2005. ICNN&B '05. International Conference on*, volume 2, pages 683 –688, 2005. [cited at p. 59]

[92] Luo Si, Rong Jin, Steven C. H. Hoi, and Michael R. Lyu. Collaborative image retrieval via regularized metric learning. *Multimedia Syst.*, 12(1):34–44, 2006. [cited at p. 26]

[93] M. Skurichina, S. Raudys, and R. P. W. Duin. K-nearest neighbors directed noise injection in multilayer perceptron training. *IEEE Trans. on Neural Networks*, 11(2):504–511, March 2000. [cited at p. 36, 38]

[94] Arnold W. M. Smeulders, Marcel Worring, Simone Santini, Amarnath Gupta, and Ramesh Jain. Content-based image retrieval at the end of the early years. *IEEE Trans. Pattern Anal. Mach. Intell.*, 22(12):1349–1380, 2000. [cited at p. 1, 8, 21, 39]

[95] Zhong Su, Stan Z. Li, and HongJiang Zhang. Extraction of feature subspaces for content-based retrieval using relevance feedback. In *ACM Multimedia*, pages 98–106, 2001. [cited at p. 29]

[96] Michael J. Swain and Dana H. Ballard. Color indexing. *International Journal of Computer Vision*, 7(1):11–32, 1991. [cited at p. 11, 51]

[97] Hideyuki Tamura, Shunji Mori, and Takashi Yamawaki. Textural features corresponding to visual perception. *IEEE Trans. Systems, Man and Cybernetics*, 8(6):460–473, June 1978. [cited at p. 12]

[98] D. Tao, X. Tang, X. Li, and X. Wu. Asymmetric bagging and random subspace for support vector machines-based relevance feedback in image retrieval. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 28:1088–1099, 2006. [cited at p. 25, 26, 32]

[99] David M.J. Tax. *One-class classification.* PhD thesis, Delft University of Technology, Delft, The Netherlands, June 2001. [cited at p. 23]

[100] Walter ten Brinke, David McG. Squire, and John Bigelow. Similarity: Measurement, ordering and betweenness. In Mircea Gh. Negoita, Robert J. Howlett, and Lakhmi C. Jain, editors, *KES*, volume 3214 of *Lecture Notes in Computer Science*, pages 996–1002. Springer, 2004. [cited at p. 26]

[101] Walter ten Brinke, David McG. Squire, and John Bigelow. The meaning of an image in content-based image retrieval. In Esperanza Marcos, Mark Lycett, César J. Acuña, and Juan M. Vara, editors, *PhiSE*, volume 240 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2006. [cited at p. 9]

[102] Joshua B. Tenenbaum. Mapping a manifold of perceptual observations. In Michael I. Jordan, Michael J. Kearns, and Sara A. Solla, editors, *NIPS*. The MIT Press, 1997. [cited at p. 27]

[103] Joshua B. Tenenbaum, Vin deSilva, and John C. Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290:2319–2322, December 2000. [cited at p. 27]

[104] Charles W. Therrien. *Decision estimation and classification: an introduction to pattern recognition and related topics.* John Wiley & Sons, Inc., New York, NY, USA, 1989. [cited at p. 27]

[105] Bart Thomee. *A picture is worth a thousand words: content-based image retrieval techniques.* PhD thesis, Leiden University, The Netherlands, November 2010. [cited at p. 1, 21]

[106] Bart Thomee, Mark J. Huiskes, Erwin M. Bakker, and Michael S. Lew. An artificial imagination for interactive search. In Michael S. Lew, Nicu Sebe, Thomas S. Huang, and Erwin M. Bakker, editors, *ICCV-HCI*, volume 4796 of *Lecture Notes in Computer Science*, pages 19–28. Springer, 2007. [cited at p. 27]

[107] Bart Thomee, Mark J. Huiskes, Erwin M. Bakker, and Michael S. Lew. Using an artificial imagination for texture retrieval. In *Pattern Recognition, 2008. ICPR 2008. 19th International Conference on*, pages 1–4, Dec. 2008. [cited at p. 27]

[108] Kinh Tieu and Paul A. Viola. Boosting image retrieval. In *CVPR*, pages 1228–1235. IEEE Computer Society, 2000. [cited at p. 33]

[109] Simon Tong and Edward Y. Chang. Support vector machine active learning for image retrieval. In *ACM Multimedia*, pages 107–118, 2001. [cited at p. 31]

[110] Yiqing Tu, Gang Li, and Honghua Dai. Integrating local one-class classifiers for image retrieval. In Xue Li, Osmar R. Zaïane, and Zhanhuai Li, editors, *ADMA*, volume 4093 of *Lecture Notes in Computer Science*, pages 213–222. Springer, 2006. [cited at p. 32]

[111] Vladimir N. Vapnik. *Statistical Learning Theory*. Wiley, New York, 1998. [cited at p. 24]

[112] James Ze Wang, Jia Li, and Gio Wiederhold. Simplicity: Semantics-sensitive integrated matching for picture libraries. *IEEE Trans. Pattern Anal. Mach. Intell.*, 23(9):947–963, 2001. [cited at p. 18]

[113] Man Wang, Zheng-Lin Ye, Yue Wang, and Shu-Xun Wang. Dominant sets clustering for image retrieval. *Signal Processing*, 88(11):2843–2849, 2008. [cited at p. 31]

[114] John M. Winn, Antonio Criminisi, and Thomas P. Minka. Object categorization by learned universal visual dictionary. In *ICCV*, pages 1800–1807. IEEE Computer Society, 2005. [cited at p. 18]

[115] Chee Sun Won, Dong Kwon Park, and Soo-Jun Park. Efficient use of mpeg-7 edge histogram descriptor. *ETRI Journal*, 24(1):23–30, Feb. 2002. [cited at p. 15]

[116] Yimin Wu and Aidong Zhang. Interactive pattern analysis for relevance feedback in multimedia information retrieval. *Multimedia Syst.*, 10(1):41–55, 2004. [cited at p. 29]

[117] Peng-Yeng Yin, Bir Bhanu, Kuang-Cheng Chang, and Anlei Dong. Integrating relevance feedback techniques for image retrieval using reinforcement learning. *IEEE Trans. Pattern Anal. Mach. Intell.*, 27(10):1536–1551, 2005. [cited at p. 32]

[118] Lei Zhang, Fuzong Lin, and Bo Zhang. Support vector machine learning for image retrieval. In *ICIP (2)*, pages 721–724, 2001. [cited at p. 22, 25, 87]

[119] Ning Zhang and Ling Guan. Graph cuts in content-based image classification and retrieval with relevance feedback. In Horace Ho-Shing Ip, Oscar C. Au, Howard Leung, Ming-Ting Sun, Wei-Ying Ma, and Shi-Min Hu, editors, *PCM*, volume 4810 of *Lecture Notes in Computer Science*, pages 30–39. Springer, 2007. [cited at p. 30]

[120] Xiang Sean Zhou and Thomas S. Huang. Small sample learning during multimedia retrieval using biasmap. In *CVPR (1)*, pages 11–17. IEEE Computer Society, 2001. [cited at p. 23, 29]

[121] Xiang Sean Zhou and Thomas S. Huang. Relevance feedback in image retrieval: A comprehensive review. *Multimedia Syst.*, 8(6):536–544, 2003. [cited at p. 21, 39]

List of Works Related to the Thesis

International Journal

- Luca Piras and Giorgio Giacinto, *Synthetic Pattern Generation for Imbalanced Learning in Content-Based Image Retrieval.* In preparation.
  (Related to Section 4.1)

International Conference Papers

- Luca Piras and Giorgio Giacinto, *Neighborhood-based feature weighting for relevance feedback in content-based retrieval.* In *WIAMIS*, pages 238–241. IEEE Computer Society, 2009.
  (Related to Section 5.1.3)

- Luca Piras and Giorgio Giacinto, *K-nearest neighbors directed synthetic images injection.* In *WIAMIS*, pages 1–4. IEEE Computer Society, 2010.
  (Related to Section 4.1)

- Luca Piras and Giorgio Giacinto, *Unbalanced learning in content-based image classification and retrieval.* In *ICME*, pages 36–41. IEEE Computer Society, 2010.
  (Related to Section 4.1)

- L. Piras, G. Giacinto, and R. Paredesv, *On-line Learning in Content Based Image Retrieval.* Submitted to ICMR, ACM International Conference on Multimedia Retrieval, April 2011.
  (Related to Section 4.2)

- Luca Piras and Giorgio Giacinto, *Dissimilarity representation from multi-feature spaces.* In preparation.
  (Related to Section 5.2.2)

Acknowledgements

Finding the words to thank everyone who helped, encouraged and have been close to me during this journey is perhaps the most difficult thing, especially because they are many and are not limited to the short list I am going to do. My first thank surely goes to Prof. Giorgio Giacinto, that gave me the opportunity to make this wonderful experience. His availability, his support and his advices made for sure these years more than a simple course of studies.

A sincere thank also goes to Prof. Roberto Paredes by making the time spent at the *Universidad Politécnica de Valencia* a profitable time in my training. I must thank all my colleagues and friends that I knew during my stay in Spain, in particular Vladimir, Ihab, Germán and Míriam, without them the months spent away from home would have been definitely harder.

I would also like to thank all my friends who have stood with me all these years, the list is very long but in particular my thanks go to Davide and Francesca with whom I shared the moments of hard work and satisfactions, over the university years and then during the doctorate.

A heartfelt thank to my family that is always so patient with me even at times when work commitments do not allow me to give them all the attentions that they deserve.

Finally a special thanks goes to Iole, for his steadfast support in difficult times and for being by my side in those happier.

To all, thank you.

Ringraziamenti

Trovare le parole per ringraziare tutte le persone che mi hanno aiutato, incoraggiato e mi sono state vicine durante questo percorso è forse la cosa più difficile, soprattutto perché sono tante e non si esauriscono nel breve elenco che mi accingo a fare. Il mio primo grazie va sicuramente al Prof. Giorgio Giacinto, che mi ha dato la possibilità di fare questa fantastica esperienza. La sua disponibilità, il suo sostegno e i consigli di cui è sempre stato prodigo, hanno fatto si che questi anni siano stati più che un semplice percorso di studi.

Un sentito grazie va anche al Prof. Roberto Paredes per aver reso il periodo trascorso all'*Universidad Politécnica de Valencia* un momento proficuo nel mio percorso formativo. Non posso esimermi dal ringraziare tutti i colleghi e gli amici che ho avuto modo di conoscere durante il mio soggiorno in Spagna, in particolare Vladimir, Ihab, Germán e Míriam, senza i quali i mesi trascorsi lontano da casa sarebbero stati sicuramente più duri.

Vorrei anche ringraziare tutti gli amici che mi sono stati accanto in tutti questi anni, l'elenco sarebbe molto lungo ma in particolare il mio grazie va a Davide e Francesca con cui ho condiviso i momenti di duro lavoro e le soddisfazioni, prima durante gli anni universitari e poi di dottorato.

Un grazie di cuore alla mia famiglia che ha sempre tanta pazienza con me anche nei momenti in cui gli impegni di lavoro non mi permettono di dare loro tutte le attenzioni che meritano.

Infine un grazie speciale va a Iole, per il suo sostegno incrollabile nei momenti più difficili e per essere al mio fianco in quelli più felici.

A tutti, grazie.