PhD Dissertation

International Doctorate School in Information and Communication Technologies

DISI - University of Trento

Photo Indexing and Retrieval based on Content and Context

Mattia Broilo

Advisor:

Prof. Francesco G. B. De Natale

Università degli Studi di Trento

February 2011

Abstract

The widespread use of digital cameras, as well as the increasing popularity of online photo sharing has led to the proliferation of networked photo collections. Handling such a huge amount of media, without imposing complex and time consuming archiving procedures, is highly desirable and poses a number of interesting research challenges to the media community. In particular, the definition of suitable content based indexing and retrieval methodologies is attracting the effort of a large number of researchers worldwide, who proposed various tools for automatic content organization, retrieval, search, annotation and summarization. In this thesis, we will present and discuss three different approaches for content-and-context based retrieval. The main focus will be put on personal photo albums, which can be considered one of the most challenging application domains in this field, due to the largely unstructured and variable nature of the datasets. The methodologies that we will describe can be summarized into the following three points:

i. Stochastic approaches to exploit the user interaction in query-by-example photos retrieval. Understanding the subjective meaning of a visual query, by converting it into numerical parameters that can be extracted and compared by a computer, is the paramount challenge in the field of intelligent image retrieval, also referred to as the "semantic gap" problem. An innovative approach is proposed that combines a relevance feedback process with a stochastic optimization engine, as a way to grasp user's semantics through optimized iterative learning providing on one side a better exploration of the search space, and on the other side avoiding stagnation in local minima during the retrieval.

ii. Unsupervised event collection, segmentation and summarization. The need for automatic tools able to extract salient moments and provide automatic summary of large photo galleries is becoming more and more important due to the exponential growth in the use of digital media for recording personal, familiar or social life events. The multi-modal event segmentation algorithm faces the summarization problem in an holistic way, making it possible to exploit the whole available information in a fully unsupervised way. The proposed technique aims at providing such a tool, with the specific goal of reducing the need of complex parameter settings and letting the system be widely useful for as many situations as possible.

iii. Content-based synchronization of multiple galleries related to the same event. The large spread of photo cameras makes it quite common that an event is acquired through different devices, conveying different subjects and perspectives of the same happening. Automatic tools are more and more used to support the users in organizing such archives, and it is largely accepted that time information is crucial to this purpose. Unfortunately time-stamps may be affected by erroneous or imprecise set-

ting of the camera clock. The synchronization algorithm presented is the first which uses the content of pictures to estimate the mutual delays among different cameras, thus achieving an a-posteriori synchronization of various photo collections referring to the same event.

Keywords

[Content-Based, Particle Swarm Optimization, Relevance Feedback, Unsupervised, Clustering, Event, Summarization, Synchronization, Context, Photos]

Contents

Introduction

Content-Based Image Retrieval (CBIR) refers to any technology that in principle helps to organize digital picture archives by their visual content [32]. The year 1992 is considered the starting point of research and development on image retrieval by content [57]. The last two decades have witnessed great interest in research on content-based image retrieval. This has paved the way for a large number of new techniques and systems, and a growing interest in associated fields to support such systems. Content based image retrieval is a field of research differentiated in many facets which contain numerous unresolved issues. Despite the effort made in these years of research, there is not yet a universally acceptable algorithmic means of characterizing human vision, more specifically in the context of interpreting images. Some intrinsic technical problems are: how to mathematically describe an image (visual signature), how to assess the similarity between images (similarity metric) how to retrieve the desired content (search paradigm) how and what to learn from content or users (learning and classification). All these issues can be referred to as the so called "*semantic gap*" that is the gap between the subjective semantic meaning of a visual query and the numerical parameters extracted and analyzed by a computer [111]. Beyond the techniques adopted, the key aspects of a content-based system are the purpose and the domain of the application. It is possible to simplify the content-based applications types according to two main tasks: *search* that covers retrieval by association, target or category search; and *annotation* that includes face and object detection and recognition, and all the different level types in concept detection (from lower to higher sematic abstraction). Understanding the nature and scope of image data plays a key role in the complexity of image search system design. Along this dimension, it is possible to classify the content-based application domain into the following categories: for consumer or *personal collections*, for the *web* or for *specific domain* data such as biomedical, satellites or museum image databases. Many researchers agree that CBIR remains well behind content-based text retrieval [52], [128], [35] this is mainly due to three great unresolved problems.

Semantic gap. Up to now it has resulted impossible to find the semantic interpretation of an image using the statistics of the values of the pixels even if significant efforts have been put into using low-level image properties [34]. From simpler methods such as color and texture histograms to more sophisticated features such as global transforms or SIFT [86], the visual signatures

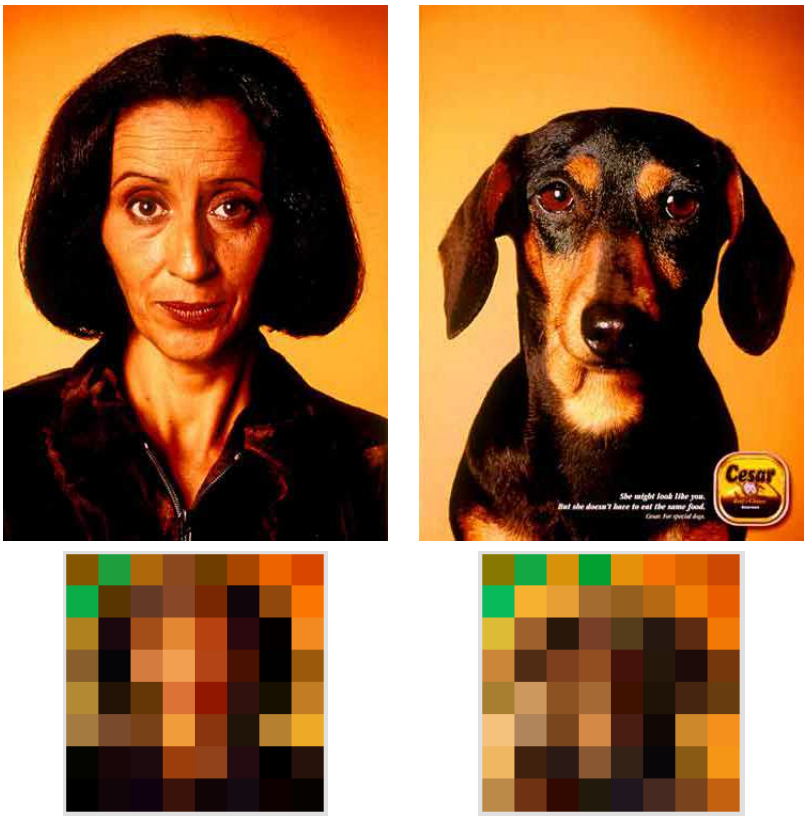
extracted from the pixel photo content fail in the description of the user perceived meaning (see figure 1).

Curse of dimensionality. Pattern classification studies demonstrate that increasing the number of features can be detrimental in a classification problem [39]. Ideally, images in a given semantic category should be projected in nearby points in the feature space. If the number of samples is small compared to the dimension of the space, then it becomes possible to find rules to associate the feature sets of "similar" images. But when new samples or new categories are added, it is unlikely that such an association will be confirmed [62] thus bringing generalization and scalability lacks in classifiers [106].

Role of the user and of the context. Each user is unique and while interacting with a system the interpretation of the data has many relationships with psychological affects. If the same photo is given to different people and they are asked to assign tags to represent the photo, there may be as many different tags as the number of people assigning them. Images content analysis is fundamentally a perception problem and the human perception is strictly connected to the context where the picture is used. If the same photo is given to a person at two different times and in different contexts, then the tags assigned could be different. If context is known, then it is possible to include that knowledge in the system design but this is possible only when we deal with a specific application area [66].

In this thesis, three different content-based applications will be presented. The tools domain is the personal photo collection while the tasks involve both the retrieval and the annotation issues. The above mentioned unresolved problems are faced using unsupervised approaches and involving the user as leading actor of the application. In the following chapters the three applications proposed will be described. Each chapter is introduced by a section depicting the related works and the state of art strictly connected to the methodologies adopted.
In chapter 1, an innovative approach is proposed which combines a relevance feedback process with a stochastic optimization engine, as a way to grasp users semantics through optimized iterative learning providing on one side a better exploration of the search space, and on the other side avoiding stagnation in local minima during the retrieval. The retrieval uses human interaction to achieve a twofold goal: (i) to guide the swarm particles in the exploration of the solution space towards the cluster of relevant images; (ii) to dynamically modify the feature space by appropriately weighting the descriptive features according to the users perception of relevance.
In chapter 2 a context and content summarization tool is presented. The application helps the user in annotation and organization tasks in an holistic way, making it possible to exploit the whole available information in a fully unsupervised way. Context expressed by time and space and content expressed by visual features and faces are fused together after an independent clustering analysis. The proposed technique aims at providing such a tool, with the specific objective of reducing the need of complex parameter settings and letting the system be widely useful for as

many situations as possible.
Chapter 3 presents a content-based synchronization algorithm to estimate the time delay among different photo galleries of the same event. Automatic tools are more and more used to support the users in organizing their photo archives, and it is largely accepted that time information is fundamental to this purpose. Unfortunately, time-stamps may be affected by erroneous or imprecise setting of the camera clock. The synchronization algorithm presented is the first to use the content of pictures to estimate the mutual delays among different cameras, thus achieving an a-posteriori synchronization of various photo collections referring to the same event.
The thesis ends with the conclusion on the work done and discusses about future challenges in content-based applications for personal photo management.



(a) (b)

Figure 1: Example of two images with similar color and textures statistics but different semantic meaning.

Chapter 1

Retrieval in Photos Database

Understanding the subjective meaning of a visual query, by converting it into numerical parameters that can be extracted and compared by a computer, is the paramount challenge in the field of intelligent image retrieval, also referred to as the semantic gap problem. In this chapter, an innovative approach is proposed that combines a relevance feedback (RF) approach with an evolutionary stochastic algorithm, called Particle Swarm Optimizer (PSO), as a way to grasp users semantics through optimized iterative learning. The retrieval uses human interaction to achieve a twofold goal: (i) to guide the swarm particles in the exploration of the solution space towards the cluster of relevant images; (ii) to dynamically modify the feature space by appropriately weighting the descriptive features according to the users perception of relevance. Extensive simulations showed that the proposed technique outperforms traditional deterministic RF approaches of the same class, thanks to its stochastic nature, which allows a better exploration of complex, non-linear and highly-dimensional solution spaces.

1.1 A Stochastic Approach using Relevance Feedback and Particle Swarm Optimization

Content-Based image retrieval (CBIR) systems analyze the visual content description to organize and find images in databases. The retrieval process usually relies on presenting a visual query (natural or synthetic) to the systems, and extracting from a database the set of images that best fit the user request. Such mechanism, referred to as query-by-example, requires the definition of an image representation (a set of descriptive features) and of some similarity metrics to compare query and target images. Several years of research in this field [104],[111],[32] highlighted a number of problems related to this (apparently simple) process. First, how good is the description provided by the adopted feature set, i.e., are the selected features able to provide a good clustering of the requested images, retrieving a sufficient number of desired images and avoiding false positives? Second, is the query significant

enough to represent the conceptual image that the user has in mind, i.e., does the query capture the semantics of the user? Third, is there a reliable method to cluster relevant and irrelevant images, taking into account that, even if relevant images may luckily represent a compact cluster, irrelevant ones for sure will not? Simple minimum-distance-based algorithms are usually unable to provide a satisfactory answer to all such problems. According to this, several additional mechanisms have been introduced to achieve better performance. Among them, relevance feedback (RF) proved to be a powerful tool to iteratively collect information from the user and transform it into a semantic bias in the retrieval process [105]. RF increases the retrieval performance thanks to the fact that it enables the system to learn what is relevant or irrelevant to the user across successive retrieval-feedback cycles. Nevertheless, RF approaches so far proposed show some critical issues yet unsolved. First, user interaction is time consuming and tiring, and it is therefore desirable to reduce as much as possible the number of iterations to convergence. This is particularly difficult when only a few new images (possibly none) are retrieved during the first RF steps, so that no positive examples are available for successive retrieval [59]. In this case, the method should introduce some alternative strategy to explore the solution space (e.g., some perturbation of the solution). Another critical problem concerns the risk of stagnation, where the search process converges to a very suboptimal local solution, thus being unable to further explore the image space. This problem is more and more evident when the size of the database increases. Again, additional mechanisms to allow enlarging the exploration are usually needed. In order to overcome the above problems, we investigate the possibility of embedding the RF process into a stochastic optimization engine able to provide on one side a better exploration of the search space, and on the other side to avoid the stagnation in local minima. We selected a particle swarm optimizer [71], for it provides not only a powerful optimization tool, but also an effective space exploration mechanism. We would like to point out that the optimal choice of the features used for image description is outside the scope of this work, and then all the tests presented are based on a very standard description based on colors and textures.

1.2 Related Work

1.2.1 Relevance Feedback

As already mentioned, the proposed technique is based on the well-known concept of relevance feedback. The basic RF mechanism consists in iteratively asking the user to discriminate between relevant and irrelevant images on a given set of results [10]. The collected feedback is then used to drive different adaptation mechanisms which aim at better separating the relevant image cluster or at reformulating the query based on the additional user input. In the first case, we may apply feature re-weighting [72] or adaptation [50] algorithms, which modify the solution space metrics, giving more importance to some features with respect to others. In the second case, also known as query shifting, we will move the initial user query towards the center of the relevant image cluster to obtain a new virtual query, which

takes into account the multiple inputs of the user across iterations [43]. Feature re-weighting and query shifting are often used jointly. A binary RF is used to train neural network systems as in PicSOM [74] or in the work of Bordogna and Pasi [13]. In [130], a fuzzy RF is described, where the user provides the system with a fuzzy judgment about the relevance of the images. It was also proposed to exploit a RF approach to model an SVM-based classifier: this is the case of the work by Djordjevic and Izquierdo [36], and of the system developed by Tian et al. [117]. A thorough survey on the existing RF techniques for image retrieval is presented in [135], while two different evaluations and comparisons of several RF methods and schemas are reported in [60] and [38].

1.2.2 Particle Swarm Optimization

In the last years, the development of optimization algorithms has been inspired and influenced by natural and biological behaviors [123]. Bio-inspired optimization approaches provided new ways to achieve nearly-optimal solutions in highly nonlinear, multidimensional solution spaces, with lower complexity and faster convergence than traditional algorithms. In this chapter, we investigate the use of a popular bio-inspired stochastic optimization algorithm called particle swarm optimization (PSO) to achieve an efficient interactive CBIR algorithm. PSO was introduced in the field of computational intelligence by Kennedy and Eberhart [70] in 1995 and is a population-based stochastic technique that allows solving complex optimization problems [40]. It is inspired by the behavior of swarms of bees, where a particle corresponds to a single bee that flies inside a problem solution space searching for the optimal solution. During the iterations, the particles move towards the swarms global best and the particles personal best, which are known positions in the solution space. These positions represent the social and the cognitive factor of the swarm and are weighted by two parameters that influence the swarm behavior in the search process. PSO has been successfully applied as an optimization tool in several practical problems [94] and in many different domains such as image classification [19], ad-hoc sensor network [131], design of antennas array [44], and neural networks [82]. PSO has been used in many cases as a way to generate optimized parameters for other algorithms. As such, it has been proposed also in the field of CBIR. In particular, in [21] and [20], it is used to build a supervised classifier based on self-organizing features maps (SOM), while in [93], it is applied to the tuning of parameters of a similarity evaluation algorithm. An in-depth study on the use of statistical methods in image retrieval problem was recently done by Chandramouli and Izquierdo [96], where the image retrieval task is treated as a classification problem.

1.3 Motivations

Although RF is for sure not new in itself, no viable alternative solutions have been proposed so far to capture the user semantics in content-based image retrieval through user interaction, unless additional information is available (e.g., textual

keywords, such as tags or annotations). RF has been used in different fields of information retrieval, but its current moderate success in the media domain is mainly due to the limited performance achieved by available algorithms, which require numerous iterations to achieve a significant number of relevant images. As a matter of fact, RF mechanisms currently used in some beta or demo version of online search engines usually rely on a simple substitution of the query with one of the images found in the previous search. In this case, the history of the search is not maintained, making impossible to achieve a real adaptation of the search. It is our opinion that more sophisticated methods are likely to be adopted in the future if effective methods to exploit the history of the search will be available. The proposed method converts the RF into an optimization algorithm, thus opening a new perspective for the development of more efficient and computationally-effective RF approaches. To this purpose, we restate the problem of finding the images that match a given user query as an optimization problem where the requested images are the ones that minimize a given fitness function. A swarm particles fly in a multidimensional feature space populated by the database images. The features provide the image description, and each image is uniquely represented by a feature vector, corresponding to a point in the space. The fitness function is defined such that minimum values are achieved when particles approach images which fit the users request. Then, swarm migration process is run so that particles may iteratively converge to the solution that minimizes the global fitness, i.e., to the cluster of images that best fit the user query. We will demonstrate that using swarm intelligence of the PSO algorithm, it is possible to substitute a generic query shifting [102] by using completely different process, where the particles of the swarm can be seen as many single retrieval queries that search in parallel, locally and globally, moving towards relevant samples and far from irrelevant ones. Practically speaking, this can be seen as a generalized query shifting algorithm, where each particle of the swarm can be considered as a query point that searches in parallel the best solution inside a local area of the feature space, and the different queries (particles) are combined by taking into account the global and the personal best. A further added value of the proposed PSO-RF with respect to other proposed CBIR algorithms is in the fact that it introduces a stochastic component to the process, thus allowing to explore the solution space in different ways, thus making it possible to climb local minima and to converge to a good solution independently of the starting point and of the path followed. This is achieved by making three components cooperate in the convergence process with appropriate weights: a social factor (where the swarm did find the best solution), a cognitive factor (where the particle did find its best solution), and an inertial factor (towards where the particle is moving).

1.4 Proposed Approach

In this section, we describe the proposed retrieval algorithm that we will call PSO-RF. PSO-RF is based on two iterative processes: feature re-weighing and the swarm updating. Both processes use the information gathered from the user, who is iter-

atively involved in the image search process. Fig. 1 shows the block diagram of the proposed algorithm. According to the classic "*query-by-example*" approach, the user selects the query image, and based on that, the system ranks the whole dataset according to a minimum distance criterion. To this purpose, each image (included the query) is mapped into a feature vector and the distance between query and image is calculated as a weighted Euclidean distance computed among feature vector pairs [121]. Initially, the weights are all equal to 1. Then, the nearest images are presented to the user, and the first feedback is requested. The feedback is binary, and labels each retrieved image as relevant or irrelevant. Accordingly, two image subsets are created, which will be progressively populated across iterations thanks to human interaction. The definition of relevant and irrelevant image subsets makes it possible to perform a first re-weighting of the features and a first updating of the swarm. Details about such procedures are provided in section 1.4.1. After that, a new ranking is calculated based on the weighted Euclidean distance (with updated weights) and the N_{FB} nearest images are again proposed to the user to collect a new feedback. The process is then iterated until convergence. During this process, the feature weights are iteratively specified to fit the users mental idea of the query, i.e., the two classified image subsets allow the system to understand which features are more important to discriminate between relevant and irrelevant images. In parallel, the optimization process is carried out by constantly updating the swarm, which progressively converges to the image cluster that contains the best solutions found across iterations.

1.4.1 Query selection and Distance calculation

The first operation is to describe the images in terms of features. The visual signature of the i-th image is made of four different feature vectors, composed by: N_{cm} color moments $\underline{x}_i^{cm}$, N_{ch} color histogram bins $\underline{x}_i^{ch}$, N_{eh} edge histogram bins $\underline{x}_i^{eh}$, and N_{wt} wavelet texture energy values $\underline{x}_i^{wt}$. The vector $\underline{x}_i = [\underline{x}_i^{cm}, \underline{x}_i^{ch}, \underline{x}_i^{eh}, \underline{x}_i^{wt}]$ of dimension $F = N_{cm} + N_{ch} + N_{eh} + N_{wt}$, provides then the overall description of the image. The computation of the parameters is usually performed off-line for database images and on-line for the user query. From that point on, each image is completely represented by its visual signature, or equivalently by a point $\underline{x}_j$ in a F-dimensional space. After the selection of the query and its mapping $\underline{x}_q$ in the feature space, the system shows the user the most similar N_{FB} images in the entire database according to equation 1.1:

$$\begin{aligned} Dist\left(\underline{x}_q;\underline{x}_j\right) = \quad & WMSE\left(\underline{x}_q^{cm};\underline{x}_j^{cm}\right) + \\ & + WMSE\left(\underline{x}_q^{ch};\underline{x}_j^{ch}\right) + \\ & + WMSE\left(\underline{x}_q^{eh};\underline{x}_j^{eh}\right) + \\ & + WMSE\left(\underline{x}_q^{wt};\underline{x}_j^{wt}\right); \end{aligned} \tag{1.1}$$

where $WMSE$ is the weighted Euclidean distance calculated between a pair of corresponding feature vectors:

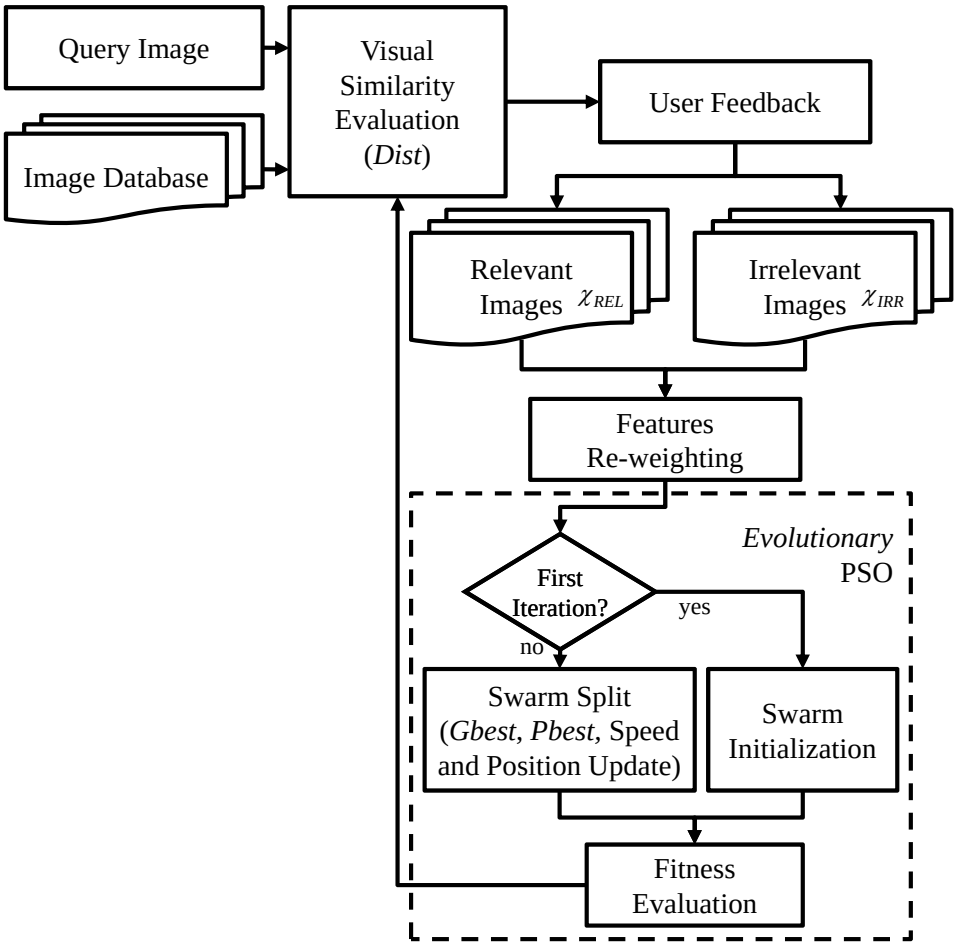


Figure 1.1: Flowchart of the proposed CBIR system.

$$WMSE\left(\underline{x};\underline{y}\right) = \frac{1}{S} \cdot \sum_{s=1}^{S} \left(x_s - y_s\right)^2 \cdot w_s^k \tag{1.2}$$

where $\underline{x}$ and $\underline{y}$ are two generic feature vectors, $\underline{w}^k$ is a vector of weights associated to the features ($s = 1, \ldots, S$, where S is equal to N_{cm} or N_{ch}or N_{eh} or N_{wt}), and k marks the iteration number. At the first iteration ($k = 1$) all the features are equally important, i.e., $\underline{w}^k = 1; s = 1, \ldots, S$. After computing $Dist\left(\underline{x}_q;\underline{x}_j\right)$; with $\underline{x}_j; j = 1, \ldots, N_{FB}$, where N_{DB} is the number of database images, a ranking is performed to sort the database according to the distance from the query. Then, the ranked list is shown to the user to collect the feedback.

1.4.2 User feedback and features reweighting

The above metric provides a quantitative measure of the visual dissimilarity of two images. It is then used to compare each of the N_{DB} images in the database with the query, and to sort them in increasing distance order. After that, the first N_{FB} results are shown to the user to collect the relevance feedback. In particular, the user is asked to tag the N_{FB} presented images as relevant or irrelevant according to his mental idea of query. Two image subsets are then created, namely relevant χ^k_{REL} and irrelevant χ^k_{IRR} sets. From this point on, the two sets are maintained and updated during all the iterations, preserving the history of the retrieval process. The aim of weight updating is to emphasize the most discriminating parameters. In practice, the idea is to perform a dynamic feature selection, driven by the user feedbacks (used as a supervision input). The feature re-weighing algorithm used in this work is similar to the one proposed in [125], and is based on a set of statistical characteristics. Taking into account the concept of *dominant range* (that is the range of a single feature of the image subset χ^k_{REL}) it is possible to calculate the *discriminant ratio* $\delta^{k,f}$ on the f-th feature ($f = 1, \ldots, F$), at the iteration k-th, which indicates the ability of this feature to separate relevant images from irrelevant ones:

$$\delta^{k,f} = 1 - \frac{\Phi^{k,f}_{CIrr}}{\Phi^k_{Irr}} \tag{1.3}$$

where Φ^k_{Irr} is the number of irrelevant images at the k-th iteration, while $\Phi^{k,f}_{CIrr}$ is the number of images in χ^k_{IRR} that have the feature f within the range associated to the corresponding feature in χ^k_{REL} . The weights are then updated as follows:

$$w^{k,f} = \frac{\delta^{k,f}}{\sigma^{k,f}_{REL}} \tag{1.4}$$

where $\sigma^{k,f}_{REL}$ is the standard deviation of the f-th feature in χ^k_{REL} at k-th iteration. To avoid problems when $\sigma^{k,f}_{REL}$ is close to zero, the method implemented in [125] has been modified with a normalization factor that limits the maximum weight to 1.

1.4.3 Swarm initialization and fitness evaluation

As previously mentioned, in our work the retrieval is formulated as an optimization process. We have therefore to model the retrieval problem in terms of a PSO. To this purpose, we define the swarm particles $\underline{p}_n$ as points inside the feature space, i.e., a F-dimensional vectors in the feature space. Given a number P of particles with $N_{FB} \leq P < N_{DB}$, we initialize the swarm by associating each particle to one the P nearest neighbors of the original query, according to the ranking performed at first iteration. Then, we generate a random speed vector $\underline{v}^k_n$; $n = 1, \ldots, P$ independently for each swarm particle, to initialize the stochastic exploration. One of the most important points in an optimization process is the definition of the target function to be minimized or maximized, called *fitness*. The fitness function should represent

the effectiveness of the solution reached by the swarm particles. Taking into account the relevant and irrelevant image sets, equation 1.5 defines the weighed cost function $\Psi^k\left(\underline{p}_n\right)$ that expresses the fitness associated to the solution found by the generic particle $\underline{p}_n$:

$$\Psi^k\left(\underline{p}_n\right) = \frac{1}{N_{rel}^k}\sum_{r=1}^{N_{rel}^k} Dist\left(\underline{p}_n^k;\underline{x}_r^k\right) + \left(\frac{1}{N_{irr}^k}\sum_{i=1}^{N_{irr}^k} Dist\left(\underline{p}_n^k;\underline{x}_i^k\right)\right)^{-1} \tag{1.5}$$

where $\underline{x}_r^k;\ r = 1,\ldots,N_{rel}^k$ and $\underline{p}_n^k;\ i = 1,\ldots,N_{irr}^k$ are the images in the relevant and irrelevant image subsets respectively. The weight vector needed to compute $Dist(\cdot)$ is the one calculated at the previous step. It is to be observed that the function $\Psi^k\left(\underline{p}_n\right)$ produces lower values when the particle is close to the relevant set and far from the irrelevant one. Therefore, the lower the fitness, the better the position of the particle is. According to the fitness value it is possible to reorder the particle swarm obtaining a new ranking. It is also worth noting that both the weights and the fitness function change across iterations because of the dynamic nature of χ_{REL}^k and χ_{IRR}^k subsets. Accordingly, features that were relevant to discriminate some images can lose importance and particles that were considered very close to the global best can become far from the relevant zone of the solution space. In most cases, the number of irrelevant images collected across iterations is greater than the number of relevant ones; this aspect has been taken into consideration during the formulation of the fitness function making it dependent on the inverse of the distance from irrelevant images. In this way, the more the average distance of the particle from irrelevant images grows, the more the fitness depends only from relevant images.

1.4.4 Evolution and termination criteria

Having defined all the elements of the optimization process, we still need to identify how to make the swarm evolve in time. To do that, we have to define some attributes of the particles. Each particle $\underline{p}_n$ holds a *personal best* $\underline{l}_n^k$ (a relevant position) and a *global best* $\underline{g}^k$ that is the best position among all the solutions found during the whole retrieval process (the same position for all the swarm particles). In our approach, the selection and the update of *personal (pbest)* and *global best (gbest)* are very different from a typical PSO implementation [99]. The global best is updated at each iteration as an image of the "relevant" set χ_{REL}^k . The image is selected according to equation 1.6: for each relevant image, the sum of the distances from the other relevant images is calculated; then, the image resulting nearest to the others is chosen as *gbest*. If there are no relevant images except for the query, *gbest* remains the position of the query.

$$\underline{g}^k = arg\min_{\underline{x}_r}\left\{\sum_{j=1}^{\chi_{REL}^k} Dist\left(\underline{x}_r;\underline{x}_j\right)\right\};\underline{x}_r,\underline{x}_j \in \chi_{REL}^k \tag{1.6}$$

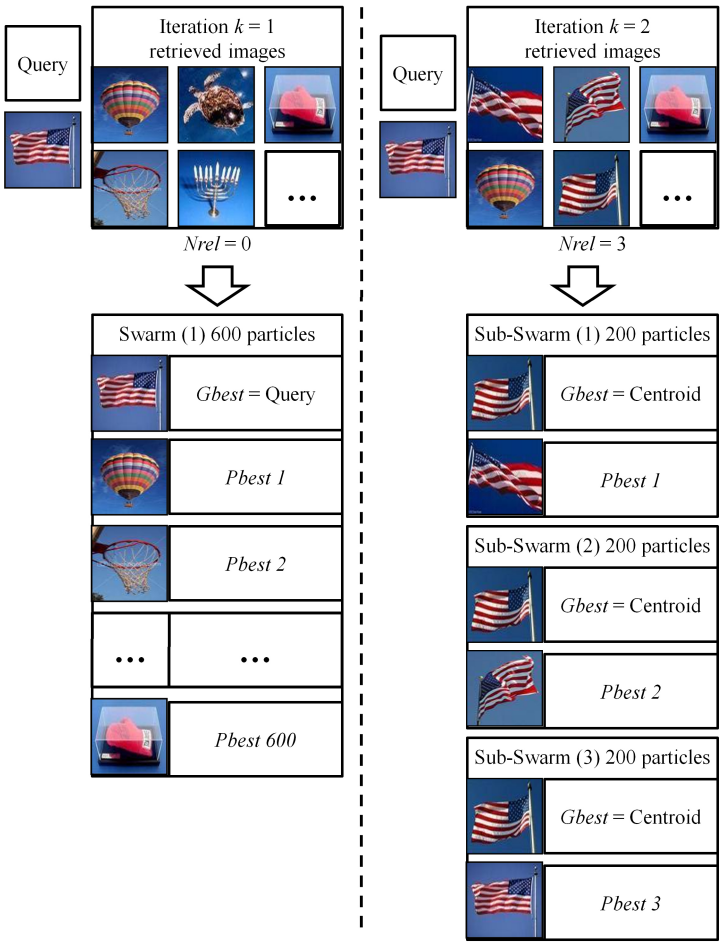


Figure 1.2: Example of *gbest* and *pbest* evolution.

*pbest*is different for each particle and is initialized as the feature vector originally associated to the particle. Until the retrieval algorithm does not find any relevant images, *pbest* will be updated according to the fitness function (equation 1.5) only if $\Psi^k\left(\underline{p}_n\right) \leq \Psi^{k-1}\left(\underline{p}_n\right)$. If the user tags some retrieved images as relevant, the swarm will be split into sub-swarms. While *gbest* position (relevant centroid) is shared among all the sub-swarms to guarantee the continuity of the convergence process, *pbest* is forced in the position of the relevant images. In this way, it is possible to better explore the features space near the relevant points, while maintaining a global reference point. An example of these concepts is provided in figure 1.2. In our tests the initial swarm evolves splitting itself till a maximum of $N_{FB}/2$ sub-swarms. Using the knowledge of the global and individual best, the speed of each particle is set, according to the following equation 1.7:

$$\underline{v}_n^k = \varphi \cdot \underline{v}_n^{k-1} + C_1 r_1 \left\{\underline{l}_n^k - \underline{p}_n^{k-1}\right\} + C_2 r_2 \left\{\underline{g}^k - \underline{p}_n^{k-1}\right\} \tag{1.7}$$

where r_1 and r_2 are uniform random numbers in the range [0,1]; and φ is an inertial weight parameter in [0.2,0.7]; progressively decreasing along iterations [56]. The inertial weight is calculated in such a way to decrease proportionally to the number of retrieved relevant images, thus allowing to slow down the swarm when approaching to convergence. The parameters C_1 and C_2 are two positive constants called *acceleration coefficients*, aimed at pulling the particle towards the position related to the cognitive (i.e., personal best $\underline{l}_n^k$) or social part (i.e., global best $\underline{p}_n^{k-1}$). Further details on the PSO parameter choice are reported in section 3.3. Finally, the position of each particle is updated as follows:

$$\underline{p}_n^k = \underline{p}_n^{k-1} + \underline{v}_n^k \tag{1.8}$$

where the sign of the speed $\underline{v}_n^k$ is changed according to the "reflecting wall" boundary condition in order to limit the search of the relevant images inside the space of admissible solutions [101]. It is worth noting that it is possible to view the particles of the swarm like query points that will explore the F-dimensional solutions space, made of the image features $f = 1, \ldots, F$, with an own speed and direction. It is useful to point out that the images of the database ($\underline{x}_j$; $j = 1, \ldots, N_{DB}$) represent a discrete and fixed set of points, while the particles can move in a continuous way inside the features space. After the swarm initialization at first iteration (setting the initial position, the random speed, the *pbest* and the *gbest*), an updating operation is done at every following iteration. The *gbest* image is recalculated if new relevant images are tagged by the user, and the new speed and position of each particle are calculated according to equation 1.7 and equation 1.8, respectively. Consequently, after every user feedback the swarm moves towards new areas in the solution space where other relevant images may be found. To complete a single iteration, a further operation is needed: to associate to particles placed in a "good position" (the lower the fitness, the better the position of the particle is) the nearest images in the database according to equation 1.1. In fact, the particles move semi-randomly in a continuous space, while database images are in discrete positions. Then, the first N_{FB} particles of the swarm ranked according to equation 1.5 are associated to their closest image, thus obtaining the new set of images to be presented to the user. If more than one particle points to the same image or a particle points to an image already classified as irrelevant, the corresponding image is discarded and next nearest neighbor is considered, until N_{FB} different images are collected. After the user feedback, the above described process of re-weighting and swarm updating is iterated. The process ends when one of the following conditions is verified:

1. the user is satisfied with the result of the search,
2. a target number of relevant images is achieved (in general N_{FB}),
3. a predefined number of iterations is reached.

After termination, all relevant images found are shown to the user.

1.4.5 Remarks on the optimization strategy

It has to be pointed out that our use of PSO substantially differs from previous works in the same field. First, we have no complete knowledge about the problem, e.g., the class "irrelevant images" is highly variable, and the supervision information is very limited and not completely available from the beginning. Second, the representation (description) of our objects is largely uncorrelated with the classification of the objects themselves (from where, the semantic gap). Finally, we do not aim at finding an optimal point in the solution space, but we want to use the swarm as a way to explore a feature space to find the best matching points, according to a cost function to be optimized. Then, the maximization of the fitness becomes a side effect (although fundamental to achieve the solution) of the convergence of the swarm to the set of relevant images. Furthermore, the basic PSO scheme is modified by introducing a "divide and conquer" schema, where the swarm can be split according to the number of relevant images retrieved. We will show that this procedure has a twofold goal: to avoid stagnation, and to allow the convergence of the swarm to multiple sub-clusters where relevant images are. As to the first point, we have to consider that we could have only a very limited number of iterations, and just very partial information on the ground truth (the initial query and the images labeled by the user across iterations). If the number of relevant images retrieved in the beginning of the process is low, the risk of stagnation is very high [15]. The adopted methodology sharply reduces such problem, increasing the exploration capabilities of the algorithm. As to the second point, we observed that, being the representation of images just based on generic visual features while classes are usually organized on the basis of visual concepts, often relevant images are not grouped in a single cluster in the feature space. Therefore, sub-swarms are more suited to handle this problem. Finally, another important difference from typical optimization problems is that the data are not completely available since the beginning, but they are collected from the user feedback across iterations. This "on-line supervision" makes the convergence faster (as compared to standard implementations), since the learning procedure is directly driven by the user knowledge.

1.5 Experimental Setup

In this section, we provide some details about the setup of the tests described and commented in Section VI. In particular, we illustrate the databases used for experimental testing, the feature set adopted for image description, and the setting of PSO parameters.

1.5.1 Image databases and image classification

At present, common standards and universally accepted data corpora to assess the performance of image retrieval systems are not available. Furthermore, it is commonly accepted that introducing a subjective feedback in the testing of RF systems makes it extremely difficult to evaluate the performances and to make comparisons

with other state-of-the-art approaches. In fact, the user can change his mind during the retrieval, make errors, and can give a personal interpretation to the data. This last problem is even more relevant in our approach, due to the stochastic nature of the algorithm, which generates at each run different results across iterations, thus making possible for a real user to follow different paths to the solution. Consequently, we made two major hypotheses:

1. we adopted two commonly used databases that, although limited in the image variety and number, provide a trustable pre-classification of images and allow an easier comparison with other state-of-the-art methods;

2. we adopted the usual procedure of providing automatic feedbacks based on pre-classified datasets, widely used in the literature.

Also in this case, this choice guarantees significant results in comparative evaluations, thanks to a consistent use of data and classification criteria. Of course, these assumptions may lead to results which do not correspond to the subjective behavior of a generic user. In fact, an image may encompass several coexisting visual concepts or even in the same object. As an example, the image class "cats" can be included at large in the image class "animals", or be further specified in a subclass "black cats". At the same time, an image containing a cat may be classified according to another subject present in the image, e.g., a dog, which is considered the main subject by the user. Additionally, an image can be classified according to some abstract concept connected to the subject (e.g., the activity it is performing, or the way it is behaving), making it even more difficult to ensure a significant labeling. In all those cases, a relevant image can be classified as irrelevant (or vice-versa) during the process and in the final evaluation, thus preventing the convergence and in any case showing suboptimal performance. Several studies are being carried on how to manage these problems through the use of appropriate knowledge (for instance, tagging, taxonomies, and ontology), which are beyond the scope of this work. In order to limit such problems, we were very careful in checking the consistency of the dataset we used for testing. Such datasets were achieved by merging two different and widely used databases, and selecting the largest possible number of image categories that presented a sufficient consistency. The selected databases were the Caltech-256 [49] and the Corel Photo Galleries[1], for their large use in the scientific literature (see, e.g., [116] and [58]). The resulting dataset includes 150 categories, each one represented by 80 images, for a total of 12 000 images. Of course, this may turn out to be limited as compared to the huge number of images used in large-scale applications. However, our objective here is to demonstrate that our method compares favorably to previously proposed approaches, on the same type of dataset used thereby. Additionally, a user interface has been developed to allow user-oriented tests and a demo of the retrieval system. Further experiments to attain subjective analysis about user satisfaction are being currently carried on.

[1] http://www.corel.com

1.5.2 Visual Signature

The selection of a significant set of descriptive features is crucial in CBIR. Specific retrieval problems and different application domains may require a careful selection of the features that best describe the image database, such as colors, textures, contours, shapes, etc. [87], [37], [119]. As previously mentioned, it is out of the scope of this work an in-depth analysis about image descriptors. Thus, a quite common set of descriptors was adopted based on low-level visual features selected according to current multimedia standards. Such features well adapt to photographic picture retrieval, and our goal will be to demonstrate that the proposed approach provides a better performance than other competing methods given an equal description. As usual, feature extraction was performed off-line, but for the query image, thus affecting to a negligible extent the retrieval speed. The feature vectors associated to each image were stored in a database for runtime access. The size of the feature vector N was set to 75, and specifically: $N_{ch} = 32$ color histogram bins, $N_{cm} = 9$ color moments, $N_{eh} = 16$ edge histogram bins, and $N_{wt} = 18$ wavelet texture energy values [34]. Color moments included first, second and third-order central moments of each color channel in the HVS color space (mean value, standard deviation, and skewness, respectively. The color histogram is calculated in the RGB color space, while the edge histogram is obtained dividing the image into four parts and the edges into four main orientations. Finally, the 18 texture features represent the coarse, middle, and fine level frequency content of the image in the wavelet domain. Features are normalized in the range [-1,1] according to their variance [5].

1.6 Results

In this section, a selection of experimental results is presented to demonstrate the effectiveness of the proposed approach. As far as parameter tuning is concerned, PSO has a few operating parameters, and we will show that its performance is very stable across a large set of experiments for a fixed setting. For all our experiments we set the number of feedback images $N_{FB} = 16$ (constant across iterations) which is a convenient number to avoid confusing or bothering the user during the interactive phase. The performance of the system was assessed in terms of precision (number of retrieved relevant images over N_{FB}) and recall (percentage of relevant images retrieved across all iterations with respect to the number of class samples). The precision-recall curves are calculated after 10 iterations, but additional charts are provided to show the convergence across iterations. All the precision-recall curves are calculated considering the first 80 ranked images (corresponding to the number of images per category in the dataset).

1.6.1 Comparison methods

The experiments presented in this section are organized as follows. First we analyze the impact of different settings of the PSO-RF parameters on retrieval performance. Then, using the best configuration, we provide the global performance

on the selected databases. The charts will also provide a comparison between our Evolutionary-PSO-RF method and 3 different retrieval algorithms. The first one is an earlier version of PSO-RF we proposed in [15], using different fitness and distance functions, and no swarm splitting. Furthermore, in that algorithm *gbest* and *pbest* were calculated as positions in the feature space, which usually do not correspond to real images. The other two methods are deterministic, and derive from a traditional query shifting method based on the Rocchio equation [10]. The first deterministic method uses a single query, and a setting of Rocchio equation parameter as follows: $\alpha = 2$; $\beta = 0.75$; and $\gamma = 0.25$, thus stressing the importance of the initial query and of the relevant images found. The second exploits the same evolution principle of our method, creating a new query for each relevant image found. To achieve a fair comparison, all the four compared CBIR systems use the same feature re-weighting function [125], except for normalization, which is set in the range [0,1] for all methods. Furthermore, all compared techniques exploit both relevant and irrelevant images tagged by the user. Finally, they use the same image similarity metric (equation 1.1), which is also used by the two deterministic algorithms to rank the database at each iteration. Due to the stochastic nature of the PSO-based algorithms, all precision and recall values plotted for those algorithms are obtained by averaging five consecutive runs for each query. An example of two different runs using the same query is showed in figure 1.3. The retrieved images are different both in amount and in retrieved temporal order.

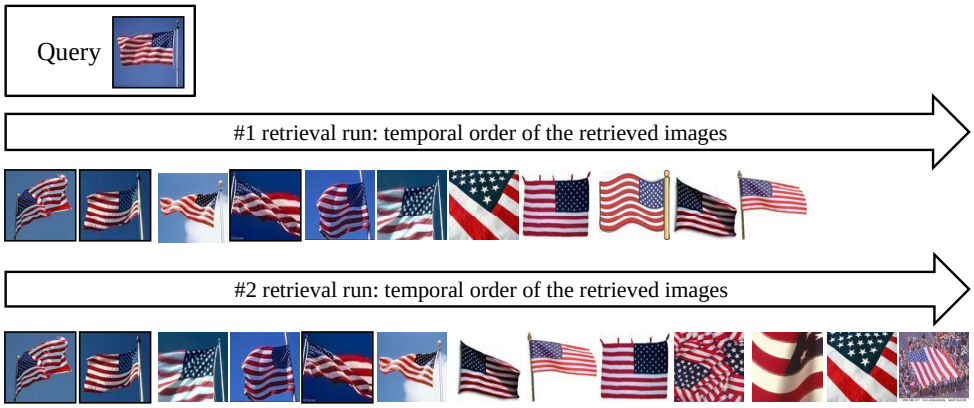


Figure 1.3: Example of two different retrieval runs with the same query.

1.6.2 Parameters tuning

PSO optimization is ruled by 4 parameters: the inertial weight φ, the cognitive and social acceleration constants C_1 and C_2, and the swarm dimension P. Such parameters have a unique goal, i.e., to determine the trade-off between global and local exploration. The inertial term spins the particles off to explore new areas of the solution space, while the cognitive and social terms attract the solution toward

previously found good points (personal and global, respectively). The dimension of the swarm is typically related to the dimensionality of the problem (in particular, the dimension of the solution space), so that the number of "agents" exploring the space is significant with respect to the extension of the space to be explored. Some restrictions on C_1 and C_2 values have been defined by Kennedy [29], based on the swarm convergence. Such restrictions force particles to avoid escalating oscillatory behaviors, this can be achieved by imposing:

$$C_1 + C_2 \leq 4 \tag{1.9}$$

The chart in figure 1.4 provides an experimental confirmation of this rule, where different combinations of C_1 and C_2 are tested with $C_1 + C_2 = 4$, providing similar results, while the two tested combinations that exceed the threshold have a dramatic loss in performance. Best results are achieved when $C_1 = C_2$, i.e., local and global attraction coincide. The curves are averaged over 2500 queries. As far as the inertial term is concerned, it determines which percentage of the particle velocity at the previous iteration is transmitted to the current one, and it is fundamental for convergence rate. Its value typically has to decrease progressively to ensure a larger exploration in the beginning of the process and a better convergence in the end. Several studies are reported on the setting of the inertial weight to ensure the swarm convergence. In [107] Eberhart and Shi suggested to vary the inertial weight linearly from 0.7 to 0.4 along iterations, to achieve a large-scale exploration in the early stages of the algorithm, and a refined exploration of the local basin at convergence [118]. In our case, we used an initial inertia of 0.7 and we decrease it till 0.2 depending on the number of the relevant retrieved images. If no relevant examples are found, the inertial weight remains constant to avoid stagnation, otherwise the inertia decreases in order to slow down the swarm speed, thus allowing a better local search when approaching the convergence. Figure 1.5 shows the results of some studies we performed on the setting of φ. The resulting performances are quite similar, except for the case when a large constant value is maintained till the end of the process, thus preventing the convergence. The best solution is achieved when the inertia decreases proportionally to the number of retrieved relevant images. Also in this case, the results are averaged over 2500 examples.

1.6.3 Swarm evolution

A further consideration concerns the impact of the swarm size and of the splitting process on the retrieval performances. During years, empirical results have shown that the swarm size can influence to some extent the PSO performance, but no general rule has been found to optimize the number of particles for every problem. Many factors contribute to the optimal swarm size setting. Among them, the problem statement with its consequent fitness formulation and the number of iterations are the most relevant. In many cases, a large swarm allows covering more easily large search spaces, with lower sensitivity to local minima. On the other side, it may create convergence problems, and increases storage and computation needs

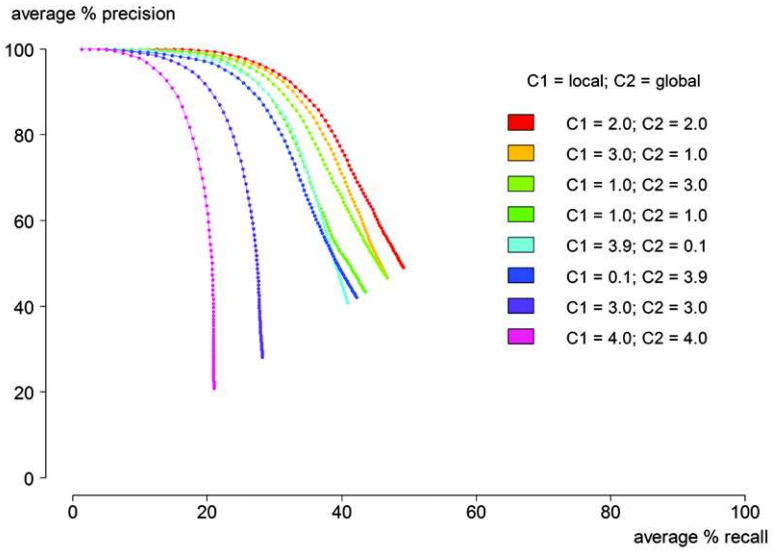


Figure 1.4: C_1 and C_2 calibration at the end of the tenth iteration: 2500 queries, linear decreasing inertia, swarm with 500 particles.

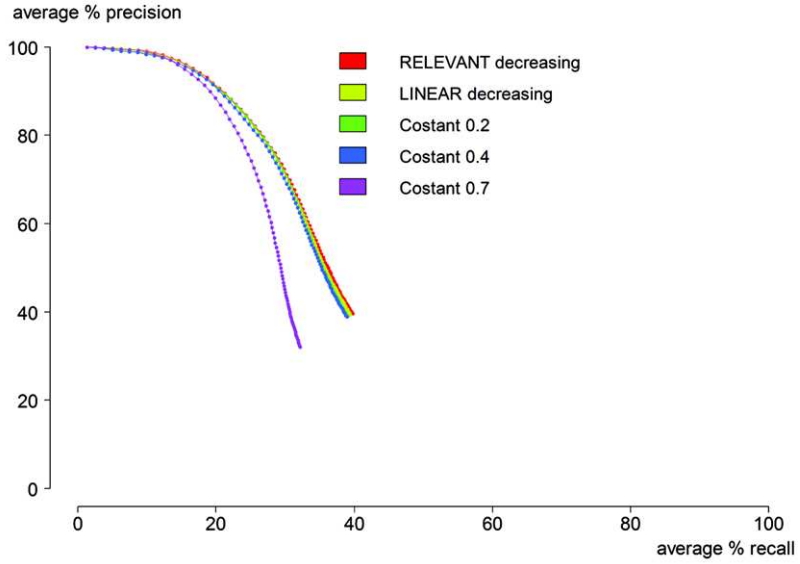


Figure 1.5: Inertial weight φ calibration at the end of the tenth iteration: 2500 queries, $C_1 = C_2 = 2.0$, swarm with 500 particles.

[14]. Figure 1.6 shows the results of some tests performed varying the swarm size in the range [16,1000]; 16 being the number of feedback images N_{FB} and 1000 being around 10% of the database size. It is possible to notice that the performance grows

with the swarm size, but there is a saturation value after which there is no further improvement (corresponding to around 100 particles). To underline the great potential of the swarm evolution in the algorithm, a representative precision-recall test is reported in figure 1.7. During iterations, the swarm splits into sub-swarms to better explore the solution space. The chart clearly shows the performance increase from single swarm to multiple sub-swarms, and also in this case a saturation parameter is clearly identifiable when the number of sub-swarms approaches N_{FB}. Another important factor for the viability of RF procedures concerns the capability of reaching a sufficient number of retrieved images with a limited number of feedbacks. To give an idea of the results achieved iteration by iteration, figure 1.8 plots the precision/recall graphs across iterations, calculated each time on the basis of the 80 best-ranking images at that stage. Analyzing that chart it is possible to see how the swarm evolves rapidly from the second iteration (the first is deterministic) to reach an asymptote after 6-7 iterations. For instance, after 5 iterations, the user has been presented less than 80 images (due to possible multiple presentation of the same relevant images), and is able to retrieve in the average 25 relevant images among the first 80 ranked in the whole dataset.

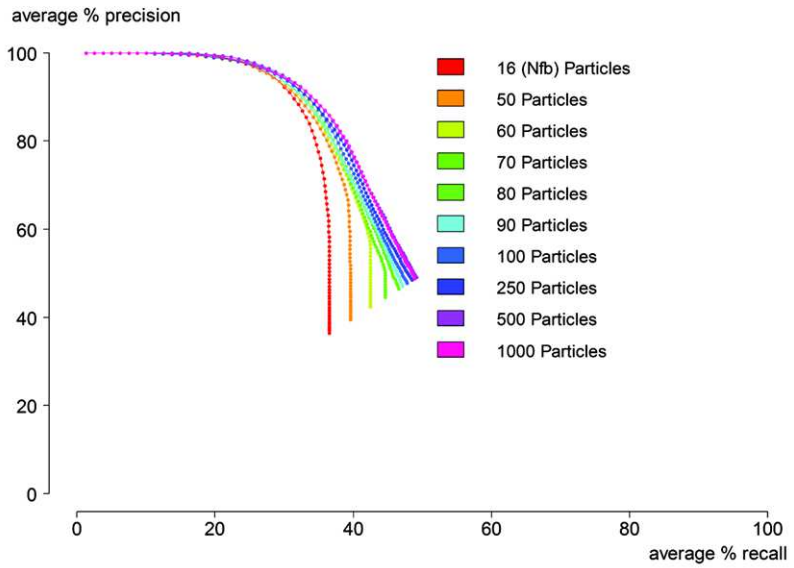


Figure 1.6: Particle number calibration at the end of the tenth iteration: 2500 queries, $C_1 = C_2 = 2.0$, decreasing inertial weight φ.

1.6.4 Final evaluation and comparisons

A comprehensive comparative test is finally presented that uses a total of 2500 queries selected from 150 classes (20 images per class). The chart in figure 1.9 compares the average percentage precision of the four methods for increasing iteration number. It is possible to observe that the compared methods are rapidly trapped by

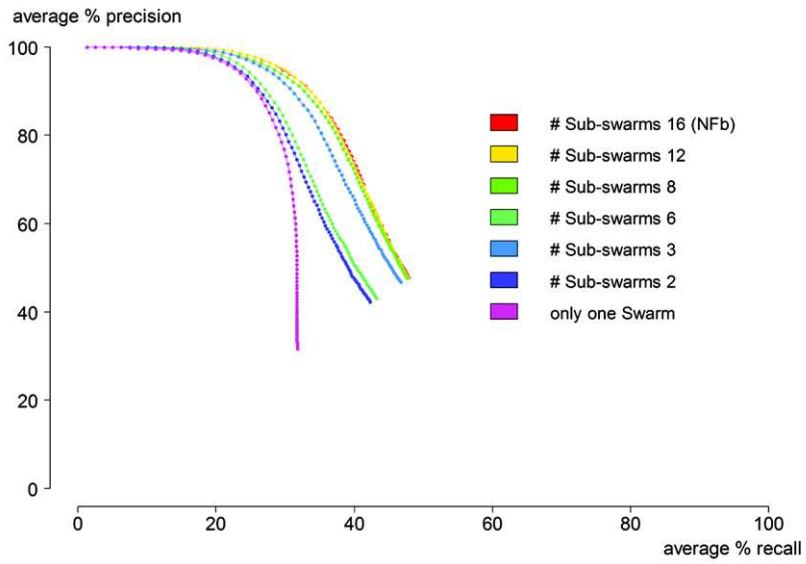


Figure 1.7: Sub-swarms number calibration at the end of the tenth iteration: 2500 queries, $C_1 = C_2 = 2.0$, swarm with 500 particles, decreasing inertial weight φ.

local minima, while the proposed one grows much faster and continues to grow for some more iterations. Figure 1.10 shows the precision-recall curve of the same experiment. From these two graphs it is possible to note that Evolutionary-PSO clearly outperforms the other three approaches. It is to be pointed out that our dataset was built as a combination of two test databases, as explained in section 3.3. This makes the test particularly realistic and challenging, since the image types and the relevant categories are not homogeneous. Being the PSO-RF a stochastic algorithm, one is not guaranteed that successive runs of the process (even with the same query) will produce equal results (in terms of retrieved images and order of retrieval). For sure, starting from different query images the result is different. What was observed to be very stable is the convergence, i.e., the number of retrieved images at the end of the process for the same image class. To show this fact we introduced a chart (figure 1.11) that plots the statistics for a reduced set of representative classes. Each curve is calculated using in turn 60 images of the same class as queries, to demonstrate the low sensitivity to a bad starting point. The tests referring to the Corel archive typically perform better, in particular for some image classes commonly used in the literature. On the other side, the Caltech database contains some particularly difficult categories, where the performance drops dramatically, mainly due to the fact that the automatic mechanism used for feedback and match analysis requires a very well defined classification of the database, which is not the case for some Caltech database classes. Furthermore, some particularly difficult classes, such as starfish, would require much more complex descriptive features to be identified than those used in our tests. Nevertheless, it is important to point out that also in these cases

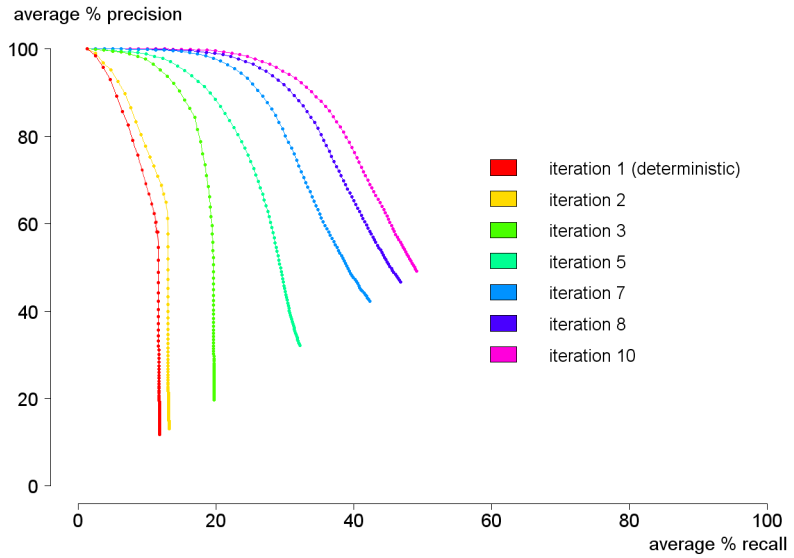


Figure 1.8: Precision-Recall improvement graph during the iterations using 2500 query images.

the proposed method behaves better than the compared ones. One last important consideration concerns the convergence of the RF, and therefore the complexity of the proposed approach. The chart in figure 1.9 clearly shows how the proposed method outperforms the other compared methods even in terms of convergence. As a matter of fact, the best compared method reaches it peak performance at convergence after 10 iterations, while Evolutionary PSO-RF reaches the same result after half of the iterations, then continuing its growth. A classical deterministic algorithm such as query shifting remains trapped in a local minimum very soon, and the proposed method is able to achieve a similar result after just 3-4 iterations. Significant differences are also reported with respect to other PSO implementations. Typical PSO algorithms may require hundreds of iterations to converge, while the proposed method converges in a few ones. This is mainly due to the interactive nature of our solution, where the convergence is guided by the repeated user feedbacks, which progressively add supervising knowledge to the problem.

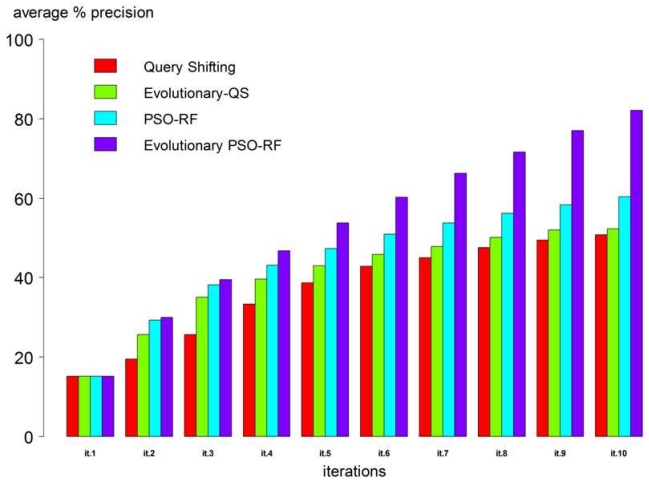


Figure 1.9: Average precision comparison of the four methods considered during 10 iterations.

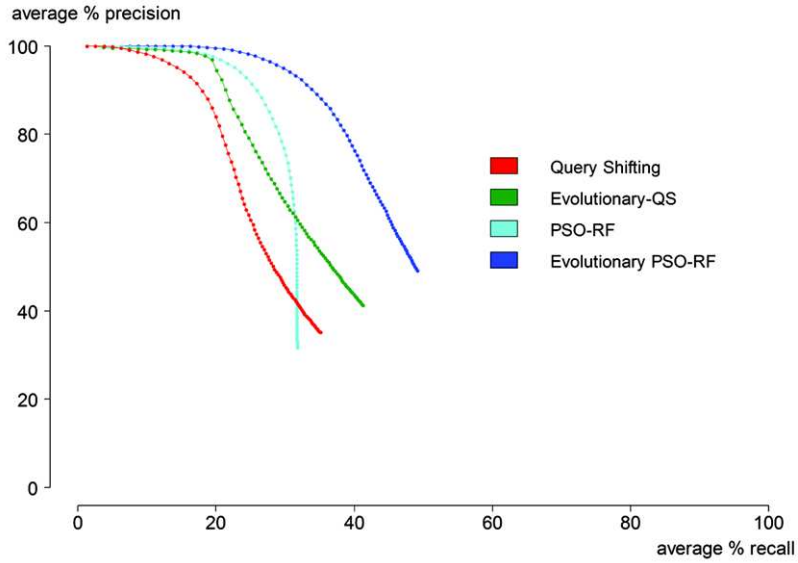


Figure 1.10: Precision-recall comparison of the four methods considered at the end of the tenth iteration.

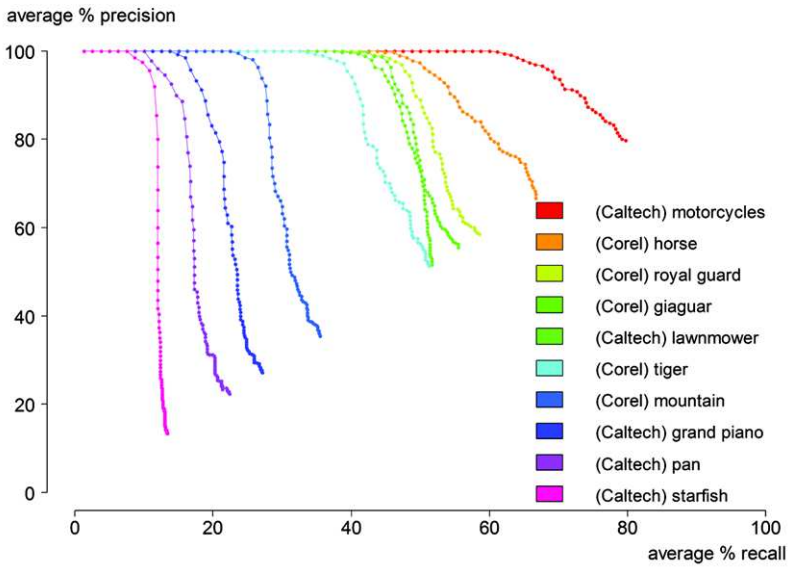


Figure 1.11: Precision-recall graph of image classes coming from different databases at the end of the tenth iteration.

Chapter 2

Photos Event Summarization

The need for automatic tools able to extract salient moments and provide automatic summarization of large photo galleries is becoming more and more important due to the exponential growth in the use of digital media for recording personal, familiar or social life events. In this chapter we present an unsupervised multimodal event segmentation method that exploits different types of information in order to automatically cluster a photo album into salient moments, as a basis for the creation of a storyboard. The algorithm analyzes the consistency in terms of time of acquisition, GPS coordinates and visual content across the gallery to detect the major points of discontinuity, which may identify the transition to a different episode in the event description. Experimental results show that the proposed system is able to produce an effective segmentation of the gallery, which well approximates the intuitive clustering made by a human operator.

2.1 Motivations

Taking pictures is the most popular way to maintain memories of an event, a travel, a person. Modern digital cameras made it easier and cheaper to collect large photo galleries of daily life. Different tools are available to organize and share all those contents, (e.g. Picasa[1] or iLife[2]); such tools provide basic functionalities to ease the user in image cataloguing, such as face recognition or geo-referencing. Nevertheless, there is still a lack of summarization facilities able to automatically select, from huge collections, significant pictures for online sharing or for further showing to different type of audiences. The need for automatic tools able to extract salient moments and provide automatic summarization of large photo galleries is becoming more and more important due to the exponential growth in the use of digital media for recording personal, familiar or social life events [108]. The creations of salient moment, and, more in general, the detection of photos belonging to the same event referable to a

[1] http://picasa.google.com
[2] www.apple.com/iLife

precise semantic concept, is of key importance in order to correctly annotate photos for any indexing and retrieval applications. Many years of machine learning research have brought to the development of different image auto-annotation systems using visual features, ontology and hierarchies. By exploiting the huge amount of labeled data available on the internet, it is possible to train extremely accurate classifiers able to detect landmarks [134], general tags [124] or particular situations [1], [73] . The main limitation of such methods is in the presence of keywords recalling abstract concepts (e.g., happiness, sadness), activities (e.g., reading, singing), or given names (e.g., names of persons or pets). Another problem of automatic annotation is the level of diversity due to personal experience, as well as linguistic and cultural differences. Since it is not possible to define a standard annotation or a fixed vocabulary, and to create a system able to automatically detect all the situations and objects that photos can represent, it is necessary to switch the focus on the user, and let him/her interact with the system to obtain a personalized result without bothering him. Event segmentation process is of key importance in order to create a meaningful and complete summary of a photo collection and it results a really tough task since it is difficult to interpret the subjectivity of each different user [84]. An unsupervised multimodal event segmentation method is proposed in this chapter, with the aim of summarizing personal photo albums in salient moments detecting the most semantically significant separation points of the collection analyzed. The summarization is based on a bottom-up hierarchical clustering that exploits matrices of visual, space and time distances. As a result, the user obtains the summary in form of a set of temporally ordered events, each one described by a representative image collage. The system fuses together different types of information analyzing them first separately and then jointly in a completely new way and it is structured in such a modular form that it results ductile for further plugged-in new kind of information. The way of combining time and content could also be managed from the user that can stress more the importance on the two different components. A second advantage is the fact that the system is fully unsupervised and the algorithm does not need any kind of training phase or parameter settings, as the needed information is extracted from the data analyzed on the fly. Personal photo summaries, events or sub-events concepts are by definition subjective in nature. Hence we conduct experiments to evaluate event segmentation both quantitatively and qualitatively involving user judges and comparing the user way of thinking with the solution proposed by the algorithm, discussing the results obtained.

2.2 Related Work

2.2.1 Summarization

Automatic summarization of digital photo sets has received increasing attention in recent years. How people manage their own photos is becoming more and more a research topic due to the large diffusion of acquisition devices that allow to collect a massive amount of digital images [103], [69]. Picture timestamp is one of the most exploited features to achieve this task [47], [88]. Also Platt et al. [97] used the inter-

photo time interval to group the pictures using an adaptive threshold. Loui and Savakis [83] proposed a K-means clustering algorithm of the time differences combined with a content-based post-processing, to divide photo collections into events. Cooper et al. [30], [45], presented a multi-scale temporal similarity to define salient moments in a digital photo library. Space, more and more available in terms of GPS coordinates or geographic landmarks, is another important information exploited to browse and organize picture archives [31]. Naaman et al. presented different frameworks [90], [130] for generating summaries of geo-referenced photographs with a map visualization. Content-based features have been also used to build systems able to summarize photos into events: Lim et al. [78], [80] summarize collections combining content and time information and use a predefined event taxonomy to annotate new photos, Chu et al. [27] exploit a near duplicate detection technique to represent a sequence of photos, while Sinha et al. [109] proposed a multimodal summarization framework at the CeWe[3] challenge for next generation tangible multimedia products in 2009. Furthermore, Li et al. [77] proposed an automatic photo collection organization based on image content and in particular based on human faces, together with corresponding clothes and nearby image regions. A top down clustering algorithm divides the photo collection into events and, introducing a contrast context histogram technique, duplicated subjects are extracted to create the summary. Ardizzone et al. [8], [41], [6] proposed a novel approach to the automatic representation of pictures, achieving a more effective organization of personal photo albums. Images are analyzed and described in multiple representation spaces, namely, faces, background, and timestamp. These three different image representations are automatically organized using a mean-shift clustering technique. Many different applications for summarizing and browsing personal photos albums with user interaction have been presented in these recent years [61], [129], [122], [28], [115]. Although the above techniques use many type of features and different algorithms to achieve photo gallery summarization, none of them faces the problem in an holistic way, making it possible to exploit the whole available information in a fully unsupervised way. The technique proposed in this work aims at providing such a tool, with the specific objective of reducing the need of complex parameter settings and letting the system be widely useful for as many situations as possible.

2.2.2 Hierarchical Clustering

Summarization and event segmentation applications deal in many cases with clustering algorithms [7]. These algorithms partition data into a certain number of clusters (groups, subsets, or categories) [65], [55]. Most researchers describe a cluster by considering the internal homogeneity and the external separation, i.e. patterns in the same cluster should be similar to each other, while patterns in different cluster should not [53]. A rough but widely agreed frame is to classify clustering techniques as hierarchical clustering and partitioning clustering, based on the properties of clusters generated [63]. Hierarchical clustering groups data objects with a sequence

[3] http://www.cewecolor.de

of partitions, either from singleton clusters to a cluster including all individuals or vice versa [68], while partitioning clustering directly divides data objects into some pre-specified number of clusters without the hierarchical structure [54], [64].

Hierarchical clustering algorithms organize data according to the proximity matrix. The results of the clustering are usually depicted by a binary tree or dendrogram. The root node of the dendrogram represents the whole data set and each leaf node is regarded as a data object. The intermediate nodes, thus, describe the extent that the objects are proximal to each other; and the height of the dendrogram usually expresses the distance between each pair of objects or clusters, or an object and a cluster. The ultimate clustering results can be obtained by cutting the dendrogram at different levels. This representation provides very informative descriptions and visualization for the potential data clustering structures, especially when real hierarchical relations exist in the data, like photos related to events, places and faces. Hierarchical clustering algorithms are mainly classified as agglomerative methods and divisive methods. Agglomerative clustering starts with clusters and each of them includes exactly one object. A series of merge operations are then followed out that finally lead all objects to the same group. Divisive clustering proceeds in an opposite way. In the beginning, the entire data set belongs to a cluster and a procedure successively divides it until all clusters are singleton clusters. For a cluster with N objects, there are $2^{N-1} - 1$ possible two-subset divisions, which is very expensive in computation [16]. Therefore, divisive clustering is not commonly used in practice. In the following discussion We focus on the agglomerative clustering. The general agglomerative clustering can be summarized by the following procedure.

1. Start with singleton clusters and calculate the distance matrix for the clusters.
2. Search the the nearest clusters pair and combine them into a new cluster
3. Update the matrix computing the distances between the new cluster and the other clusters.
4. Repeat steps (2) and (3) until all objects are in the same cluster.

Based on the different definitions for distance between two clusters, there are many agglomerative clustering algorithms. The simplest and most popular methods include single linkage [46], [113] and complete linkage technique [126]. For the single linkage method, the distance between two clusters is determined by the two closest objects in different clusters, so it is also called nearest neighbor method. On the contrary, the complete linkage method uses the farthest distance of a pair of objects to define inter-cluster distance. Both the single linkage and the complete linkage method can be generalized by the recurrence formula proposed by Lance and Williams [76]. Several more complicated agglomerative clustering algorithms, including group average linkage, median linkage, centroid linkage, and Wards method, can also be constructed [89]. Single linkage, complete linkage and average linkage consider all points of a pair of clusters, when calculating their inter-cluster distance,

and are also called graph methods. The others are called geometric methods since they use geometric centers to represent clusters and determine their distances. Remarks on important features and properties of these methods are summarized in [16]. More inter-cluster distance measures, especially the mean-based ones, were introduced by Yager, with further discussion on their possible effect to control the hierarchical clustering process [127].

2.3 Proposed Framework

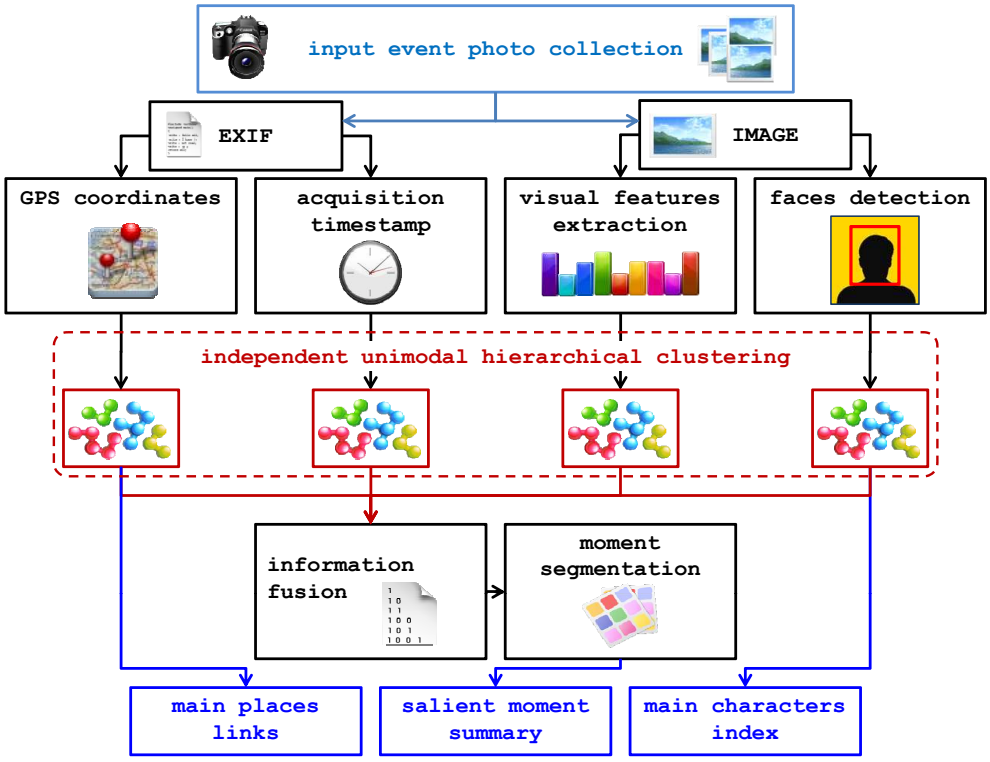


Figure 2.1: Summarization framework.

Figure 3.1 shows a block diagram of the proposed framework. The system implemented takes in input a photos gallery related to an event. From the EXIF file of each photos it extracts the acquisition time and the GPS coordinates if they are present. The content of the photos is analyzed in order to detect faces and extract regional and global descriptor of colors and textures. Each component of the photo collection is then treated separately exploiting particularly suited hierarchical clustering algorithms presented in the next section. All the clustering results are then fused together in order to build a correlation story histogram which is segmented in order to obtain a salient moment summary of the event analyzed. Furthermore from the GPS coordinates clustering it is possible to have a set of links to the main

geographical locations where the event has been taken place, while from the face clustering an index of the main characters is produced. Next section will describe into details the operation performed in each block of the framework of figure 3.1.

2.4 Photo Clustering

From each photo belonging of the considered collection, 4 different types of information are extracted:

1. global and local statistics of color and textures,
2. photo acquisition timestamp,
3. GPS coordinates,
4. detected faces.

The information are analyzed separately in order to find different type of similarities and grouping together photos according to different types of hierarchical algorithms.

2.4.1 Content-based hierarchical clustering

Given one photo collection, the system extracts from each image I_i, $i = 1, \ldots, N$, a set of 10 CEDD vectors [23], 9 of which are related to the 9 non-overlapping sub-images, (vectors $\overline{x}_i^r$, $r = 1, \ldots, 9$), and the last to the whole image (vector $\overline{x}_i^e$). Each vector is made of 144 features representing a set of color and texture visual statistics [119]. An examples of the features extraction process is depicted in figure 2.4.

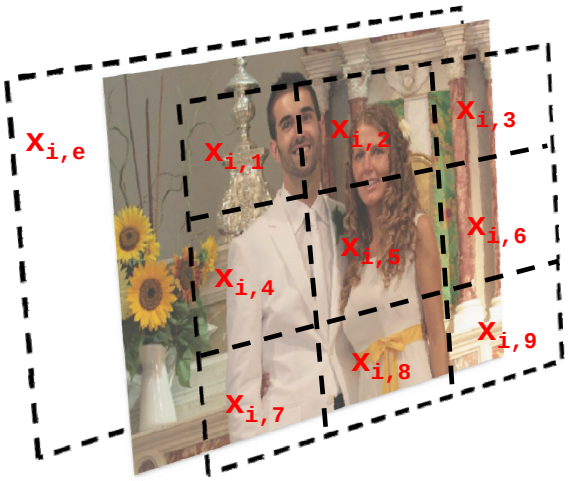


Figure 2.2: Summarization framework.

As far as the similarity metric is concerned, two distances $D_1(I_i, I_j)$ and $D_2(I_i, I_j)$ are defined using the non-binary Tanimoto coefficients (equation 2.1, 2.2) [42]. D_1 expresses the average distance of corresponding sub-images and stresses the local similarity among the two pictures; D_2 accounts for the global similarity.

$$D_1\left(I_i, I_j\right) = \frac{1}{9}\sum_{r=1}^{9}\frac{\overline{x}_i^{rT}\cdot\overline{x}_j^r}{\overline{x}_i^{rT}\cdot\overline{x}_i^r + \overline{x}_j^{rT}\cdot\overline{x}_j^r - \overline{x}_i^{rT}\cdot\overline{x}_j^r} \tag{2.1}$$

$$D_2\left(I_i, I_j\right) = \frac{\overline{x}_i^{eT}\cdot\overline{x}_j^e}{\overline{x}_i^{eT}\cdot\overline{x}_i^e + \overline{x}_j^{eT}\cdot\overline{x}_j^e - \overline{x}_i^{eT}\cdot\overline{x}_j^e} \tag{2.2}$$

Then, two $N \times N$ distance matrices M_1 and M_2 are built, whose entries are the D1 and D2 distances, respectively, calculated on each image pair in the input collection. M_1 and M_2 are clearly symmetric with null diagonal and are the basis for an unsupervised clustering process based on the single-linkage method (SLC) [46]. This method has two main advantages: first, it does not require a pre-defined number of clusters, and second, is a deterministic process that does not depend on the initial configuration or starting clustering point. The process starts with a cluster for each image in the collection, then we have an initial set of clusters C_k, $k = 1, \ldots, K$, with $K = N$. The merge process starts by using only M_1, thus linking together all picture pairs for which $D_1(I_i, I_j)$ is minimum. For each cluster C_k, the mean distances $\mu_1(C_k, I_j)$ are calculated by averaging, for each image I_j not belonging to the cluster, the D_1 distances from the images in the cluster, as in the following equation 2.3:

$$\mu_1\left(C_k, I_j\right) = \frac{1}{P_k}\sum_{p=1}^{P_k} D_1(I_p, I_j); I_p \in C_k \tag{2.3}$$

where P_k is the number of images in the k-th cluster. The nearest-neighbor to the cluster C_k is I_k^*, with:

$$I_k^* = arg\min_{I_j \notin C_k}\left\{\mu\left(C_k, I_j\right)\right\}; I_k^* \in C_h \tag{2.4}$$

The merge step is performed with the following rule:

$$if\left\{\begin{array}{c} I_k^* \in C_h \wedge \\ I_h^* \in C_k \wedge \\ \left[\mu\left(C_k, I_k^*\right) < th_d \vee \mu\left(C_h, I_h^*\right) < th_d\right] \end{array}\right\} \Rightarrow merge\left(C_h, C_k\right) \tag{2.5}$$

where th_d, $d = 1, 2$, is a threshold calculated from the initial distance matrices and in particular is the first quartile [51] of the distribution of the D_1 values (th_1) and the first quartile of the distribution of the D_2 values (th_2). An example of distances matrix and respective histograms are depicted in figure 1 and 2 where the two different thresholds are highlighted with the distance distribution boxplots.

The algorithm stops when there are no more mutually connected clusters with $\mu_1(C_k, I_k^*)$ less than th_1 or with $\mu_1(C_h, I_h^*)$ less than th_1. The use of the distance D_1 in this first clustering phase means that, till now, images that show a high similarity for

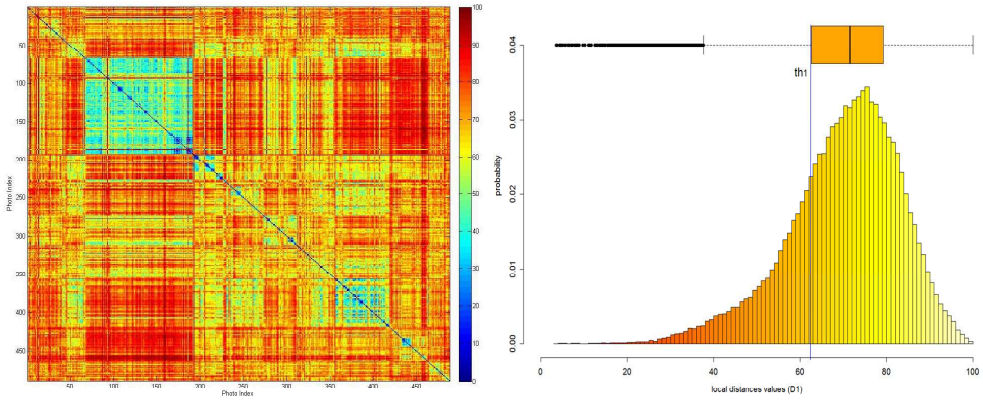


Figure 2.3: Pair-wise distance matrix using D_1 and distances value histogram wit the th_1 selected value.

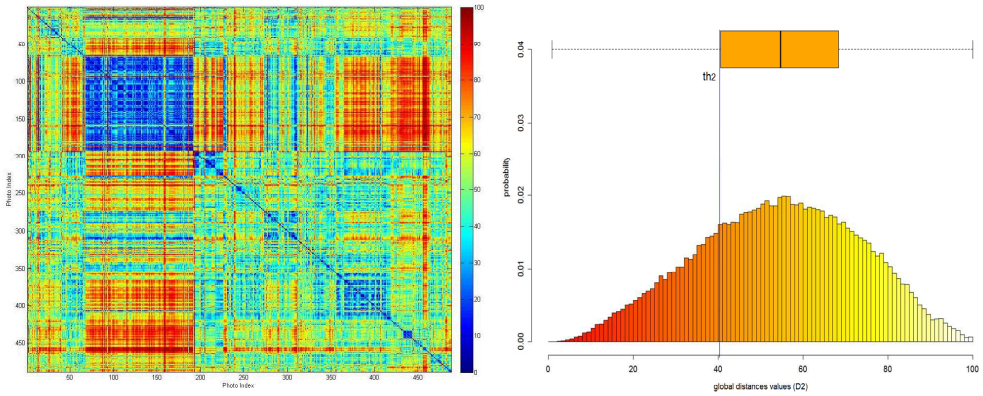


Figure 2.4: Pair-wise distance matrix using D_2 and distances value histogram wit the th_2 selected value.

all sub-images, have been merged. The result is then a large number of very similar (or near-duplicate) images with an early stagnation of the process. To achieve a higher level of diversity in the clustering we introduce a second merging phase based on the D2 distance. In order to do this, Equation 3 is modified by replacing D_1 with D_2, thus calculating the average distances $\mu_2(C_k, I_j)$ based on the global features. The linking process is then restarted with the rule in 2.5 by replacing th_1 with th_2. In this way, clusters of photos that are globally similar although locally different (e.g., mirrored images or images with similar contents in different positions) may be fused. Even in this case, the process stops itself when there are no more mutually similar clusters to merge with $\mu_2(C_k, I_k^*)$ less than th_2 or with $\mu_2(C_h, I_h^*)$ less than th_2. The first phase generates small sets of highly-uniform images, while the second phase progressively merges clusters with weaker mutual similarity. As soon as the algorithm proceeds in the merging, the entities to be clustered are not single images but increasingly larger sets of homogeneous pictures. Finally, after the termination

of the second merging phase, the two clustering final steps (with local and global distance) are stored using two binary matrices M_C^{lc} (local content clustering output matrix) and M_C^{gc} (global content clustering output matrix) where the values are 1 for couple of images belonging to the same cluster or 0 otherwise.

2.4.2 Timestamp-based hierarchical clustering

In order to obtain meaningful sub-event segmentation time information must be exploited thus importing the photo collection, the system automatically reorders the pictures according to the shoot timestamp. Even for this type of information two different clustering levels are created exploiting the previously explained algorithm: one level stress the importance of local shooting time and the other that considers the whole period of the photo collection. The hierarchical process starts from a matrix M_3 that contains pair-wise time interval D_3 between couple of photos.

$$D_3\left(I_i, I_j\right) = |t_i - t_j| \tag{2.6}$$

Even in this case the process starts with a cluster for each image in the collection, The merging operation uses values of M_3, linking together all picture pairs for which $D_3\left(I_i, I_j\right)$ is minimum and updating the mean distances $\mu_3(C_k, I_j)$. To obtain the local clustering level the threshold th_3 used is the first quartile of the distribution of the inter-photo interval $\Delta_i(t_i - t_{i-1})$ while to obtain the global final steps th_4 is the first quartile of the distribution of the D_3 values. An example of D_3 and Δ_i calculation is depicted in figure 2.5 The algorithm stops into two steps: first when there are no more mutually connected clusters with $\mu_3(C_k, I_k)$ less than th_3 and second when there are no more mutually connected clusters with $\mu_3(C_k, I_k)$ less than th_4. These two clustering levels are binarized obtaining two matrices M_C^{lt} (inter-photo time interval clustering output matrix) and M_C^{gt} (global time distance distribution clustering output matrix) where the values are 1 for couple of images belonging to the same cluster or 0 otherwise.

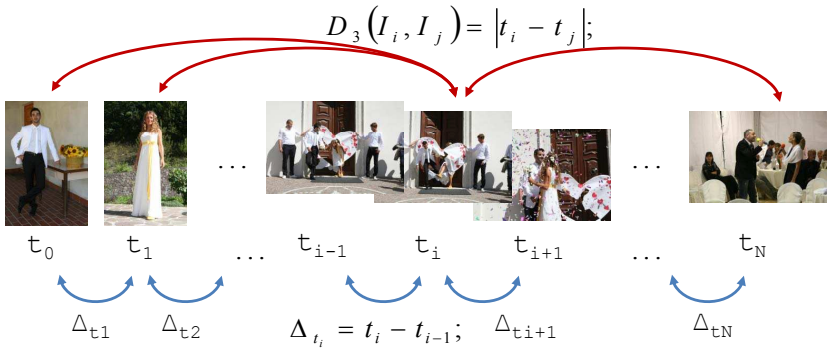


Figure 2.5: Example of D_3 and Δ_i calculation in a photos collection.

2.4.3 GPS-based hierarchical clustering

If GPS coordinates are present in the photos EXIF file, it is possible to summarize the event according to space. Location information is meaningful since it is possible to link events or moments to particular places connecting to web application such as google maps[4] adding, in this way, a different (from time-based of content-based) type of navigation and browsing and enriching the information of the event grabbing information of the location where the events have been taken places [81].

Even from GPS information two different clustering levels are created exploiting the previously explained algorithm: one level stress the importance of the local spatial movements according to the frequency shooting time and the other considers the whole period of the photo collection and the relative spatial movements. The hierarchical process starts from a matrix M_4 that contains pair-wise spatial distance D_4 between couple of photos assuming the earth spherical the distance is calculated as follows:

$$D_4\left(I_i, I_j\right) = 2 \cdot R \cdot \sin^{-1}\left(\frac{D_{GPS}\left(I_i, I_j\right)}{2 \cdot R}\right) \tag{2.7}$$

where R is the radius of the earth and $D_{GPS}\left(I_i, I_j\right)$ is calculated according to equation 2.8.

$$D_{GPS}\left(I_i, I_j\right) = \sqrt{\left(I_i(x_c) - I_i(x_c)\right)^2 + \left(I_i(y_c) - I_i(y_c)\right)^2 + \left(I_i(z_c) - I_i(z_c)\right)^2} \tag{2.8}$$

$I(\alpha)$ with $\alpha = x_c, y_c, z_c$ are the cartesian coordinates of the photo obtained from the conversion of the Longitude and Latitude information.

$$I(x_c) = 6367 \cdot \cos\left(\frac{2\pi L_{ong.}}{360}\right) \cdot \sin\left(\frac{2\pi L_{at.}}{360}\right) \tag{2.9}$$

$$I(y_c) = 6367 \cdot \sin\left(\frac{2\pi L_{ong.}}{360}\right) \cdot \sin\left(\frac{2\pi L_{at.}}{360}\right) \tag{2.10}$$

$$I(z_c) = 6367 \cdot \cos\left(\frac{2\pi L_{at.}}{360}\right) \tag{2.11}$$

Even in this case the process starts with a cluster for each image in the collection, The merging operation uses values of M_4, linking together all picture pairs for which $D_4\left(I_i, I_j\right)$ is minimum and updating the mean distances $\mu_4(C_k, I_j)$. To obtain the local clustering level the threshold th_5 used is the first quartile of the distribution of the inter-photo spatial distance calculated as:

$$\Theta_i = D_4\left(I_i, I_{i-1}\right) \tag{2.12}$$

between two time consecutive photos; while to obtain the global final steps th_6 is the first quartile of the distribution of the D_4 values. An example of D_4 and

[4] www.maps.google.com

Θ_i calculation is depicted in figure 2.6. The algorithm stops into two steps: first when there are no more mutually connected clusters with $\mu_4(C_k, I_k)$ less than th_5 and second when there are no more mutually connected clusters with $\mu_4(C_k, I_k)$ less than th_6. These two clustering levels are binarized obtaining two matrices M_C^{ls} (inter-photo spatial distance clustering output matrix) and M_C^{gs} (global photo spatial distance clustering output matrix) where the values are 1 for couple of images belonging to the same cluster or 0 otherwise.

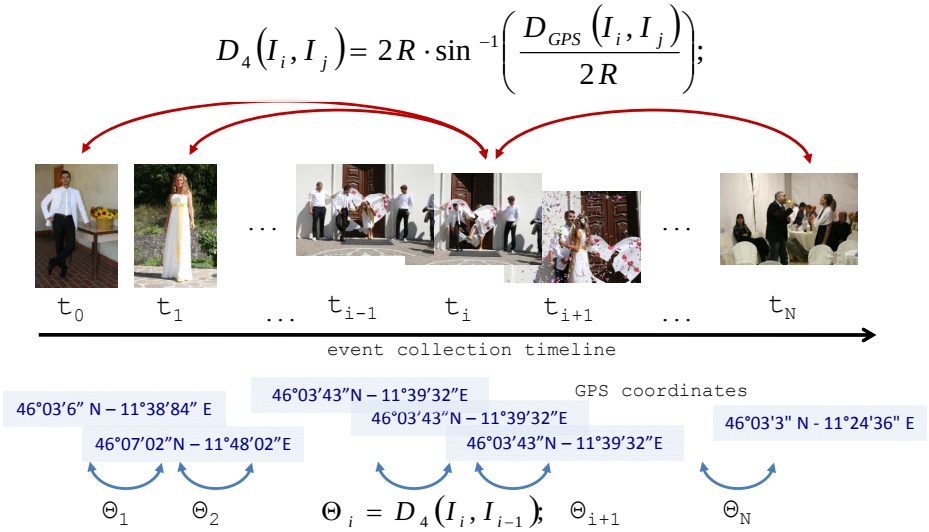


Figure 2.6: Example of D_4 and Θ_i calculation in a photos collection.

2.4.4 Face clustering

Automatic face annotation has received great attention in recent years and many systems have been developed exploiting hierarchical clustering [91], [92], [26], [114], [133], [25]. In our framework we introduce the face clustering as important information to detect different situations in an event because people more and more often organize personal photo collections according to the characters of the gallery [103].

The Viola-Jones face detection algorithm is used to find the faces regions inside the photos [120]. Each detected face region in a photo is represented as a Local Binary Pattern (LBP) [4] vector with 2124 bins ($LBP_{8,2}^{\mu 2}$ in 21×25-sized windows [3]). A second vector of CEDD color and textures features [23] is computed from region below the face, referred to here as the torso. This information helps in major characters clustering by matching low-variance clothing within a day-event, not faces alone. Ultimately, two similarity matrices are created: one with faces distances and one with the respective torso using D_2. After feature extraction and distance computation, there are two affinity matrices (face and torso) for each detected person. A pre-filtering process first removes outliers and spurious faces.

First, faces whose bounding boxes overlap and faces whose torso region is out of the photo boundaries are rejected. Second, an outlier analysis is performed using statistics of the affinity matrix. An integral over the distance matrix on each row is computed and the faces exceeding the threshold τ_o, derived in equation 2.13 below are discarded.

$$\tau_o = Q_{3rd} + 1.5 \cdot (Q_{3rd} - Q_{1th})\,; \tag{2.13}$$

where Q_{1th} and Q_{3rd} are the respective first and the third quartiles of the integral distance distribution. This thresholding process is repeated until no outliers are found. This approach follows a popular data mining algorithm discussed in detail in [51]. An example of the integral faces distance distribution, before and after the filtering process, is shown in figures 2.7 and 2.8.

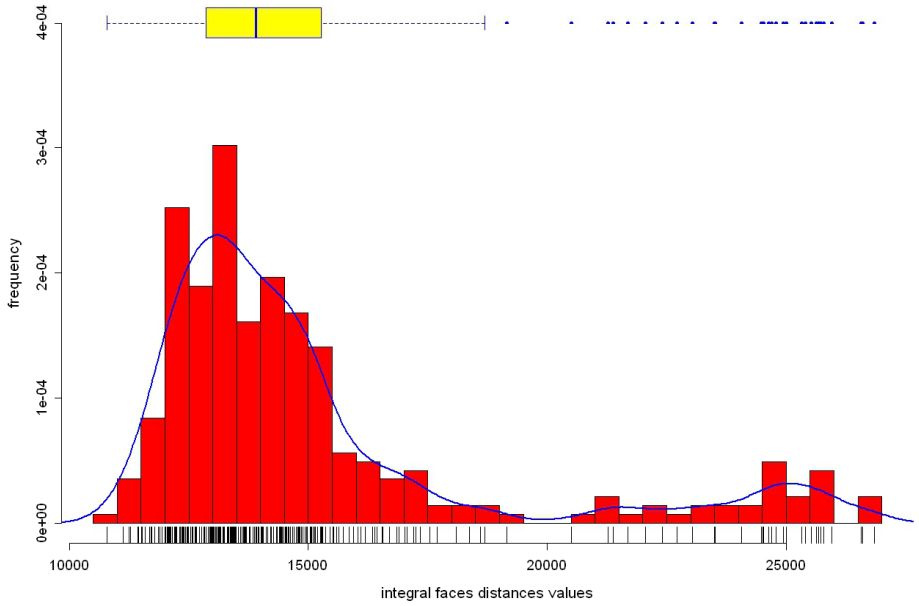


Figure 2.7: Distribution of integral faces distance before filtering.

Once the filtering process is completed, two different values are calculated for the resulting face and torso distance matrix according to the following equations where Q^f_{1rd} and Q^t_{1rd} are the first quartile of the face and torso distances.

$$\tau_f = \frac{Q^f_{1rd}}{2} \tag{2.14}$$

$$\tau_t = \frac{Q^t_{1rd}}{2} \tag{2.15}$$

The detailed clustering procedure is as follows.

1. Initially, each person (combination of face and torso) is a cluster on its own.

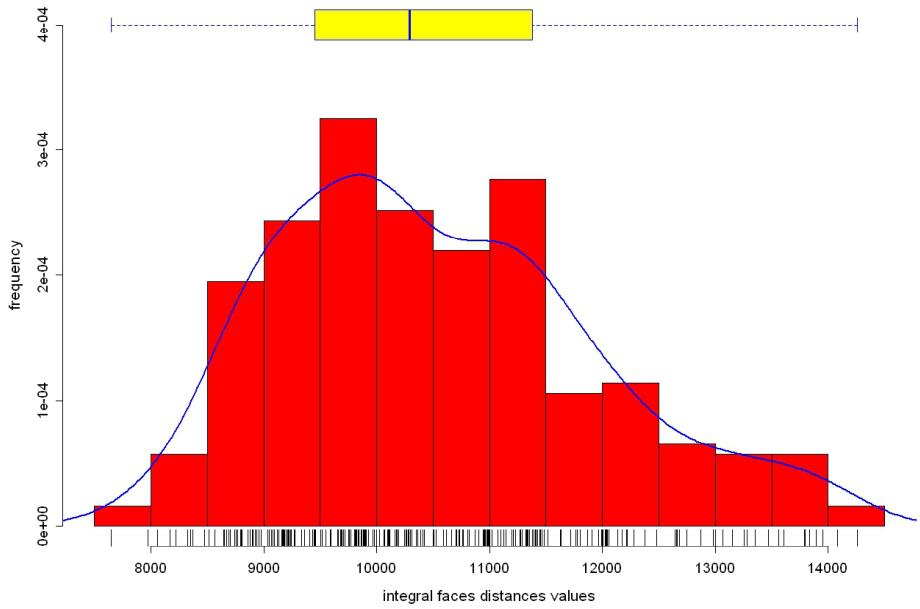


Figure 2.8: Distribution of integral faces distance after filtering.

2. At every iteration, the couple of people that have mutually minimumD1 distance both in the face matrix and in the torso one are clustered together if and only if their face and torso distances are smaller than τ_f and τ_t respectively.

3. The cluster created is represented by two average features vectors of the face and the torso, updated at every merge.

4. After the merging all the distances D_2 among clusters and the τ_f and τ_t values are recalculated. These steps are repeated until no mutually nearest clusters (either face or torso) are found with distances smaller than the two thresholds.

Once the clustering process is completed only the cluster with more than 3 faces are kept and a matrix M_C^{Fs} representing the relations between photos according to the face clustering is created. Each point of the matrix $M_C^{Fs}(i,j)$ represents the number of faces in common between a photos pairs i and j of the same event collection.

2.5 Information Fusion

The key point of a multimodal analysis is how to fuse together the different signals. In this work we propose an approach that has relapses in a possible user interaction. After all the independent unimodal clustering, a new matrix is built by a linear combination of the 7 output matrices created by the clustering. In particular, each value of the new matrix $M_{IF}(\cdot)$ will be a weighted sum of the 7 digits of the output clustering matrices $M_C(\cdot)$ calculated according to equation 2.16.

$$M_{IF}(i,j) = \sum_{\alpha} w^{\alpha} \cdot M_C^{\alpha}(i,j); \tag{2.16}$$

where i and j correspond to the matrix indexes (photos pair) while $\alpha = lt$, lc, ls, gt, gc, gs, Fs refers to the output clustering matrices: lt for inter-photo time interval, lc for local content clustering, ls for inter-photo spatial distance, gt for global time distance distribution, gc for global content clustering, gs for global photo spatial distance clustering and Fs for face clustering output matrix. To each clustering value we aggregate a weight w^{α} in order to stress the importance of the clustering output. This aggregation let the possibility of an easy interaction with the user that can stress/alleviate the importance of the time or content component, by changing the weights. This is a great advantage respect to previous approaches that only propose a fixed way of interpretation of the different signal components. This representation of many unimodal signals, is also really modular, letting the possibility to add as much as image relational information as possible, adding just other matrices that represent any other different type of clustering. For instance, it is possible to add values that represent the clusters according to any other information that could be extracted from EXIF or pixel of the photos.

2.5.1 Story histogram creation

The values of the matrix points $M_{IF}(i,j)$ represent the correlation according to different type of clustering between two images i and j of the photo event. In order to exploit this information two different histograms are created from the matrix. $H^f(i)$ represents the correlation between the i-th image and all the following (in time); while $H^b(i)$ is the correlation between the i-th image and all the previously taken. In $H^f(\cdot)$ the smaller the value, the lower the correlation of a picture with the following ones and vice versa; while in $H^b(\cdot)$ the smaller the value, the lower the correlation of a picture with the previous ones and vice versa. These histograms are calculated as follows:

$$H^f(i) = \sum_{j=i}^{N} M_{IF}(i,j); \tag{2.17}$$

$$H^b(i) = \sum_{j=0}^{i} M_{IF}(i,j); \tag{2.18}$$

where N is the total number of images inside the collection and the images are time ordered. The final correlation story histogram is a combination of $H(\cdot)^f$ and $H(\cdot)^b$ according to equation 2.19.

$$H^c(i) = 2 \cdot \frac{H^f(i) \cdot H^b(i)}{H^f(i) + H^b(i)} \tag{2.19}$$

Each bin of this story histogram represents the degree of similarity of a single photo with the other photos of the collection. In this way, sequences of clustered

coherent pictures tend to form a peak, separated by local minima from neighboring image groups. As a matter of fact, correlogram minima are associated to correlation discontinuities in content, or time, or space or all of them. These points correspond to the salient moment separating points of the event.

2.6 Salient Moment Segmentation

Once the correlation histogram is built, the system adopts the non-parametric approach for histogram segmentation presented in [33] to obtain the final list of subevents filtering local minima. This method segments histogram without any a priori assumptions about the underlying density function and considers a rigorous definition of an admissible segmentation, avoiding over and under segmentation problems. Let $H^c(\cdot)$ be a correlation histogram on $\{1, \ldots, L\}$. A segmentation $S = \{s_0, \ldots, s_n\}$ of $H^c(\cdot)$ is admissible if it satisfies the following properties:

1. $H^c(\cdot)$ follows the unimodal hypothesis on each interval $[s_i, s_{i+1}]$;
2. there is no interval $[s_i, s_j]$ with $j > i + 1$, on which $H^c(\cdot)$ follows the unimodal hypothesis.

The first requirement avoids under segmentations, and the second one avoids over segmentations. Starting from the segmentation defined by all the local minima of $H^c(\cdot)$, the algorithm merges recursively the consecutive intervals until both properties are satisfied. The unimodal hypothesis of an interval $[a, b]$ of $H^c(\cdot)$ are satisfied if there are no meaningful rejections, comparing the relative entropy [22] of the original histogram $H^c(a, b)$ with the Grenander estimator $H^r(a, b)$ [48], [12] calculated on the same interval using the "Pool Adjacent Violators" algorithm [9]. The algorithm is made by the following steps: For each t in the interval $[s(i-1); s(i+1)]$, the increasing Grenander estimator $H_i^r(\cdot)$ of $H^c(\cdot)$ on the interval $[s(i-1), t]$ and the decreasing Grenander estimator $H_d^r(\cdot)$ of $H^c(\cdot)$ on the interval $[t, s(i+1)]$ are calculated where $L_c = t - s(i-1) + 1$ is the length of the interval $[s(i-1), t]$ and $N_c = H_i^r(s(i-1), t)$ its number of samples (respectively $L_d = s(i+1) - t + 1$ and $N_d = H_d^r(t, s(i+1))$ are the length and number of samples of $[t, s(i+1)]$). For each sub-interval $[a, b]$ of $[s(i-1); t]$ the "number of false alarms" are calculated using equation 2.20.

$$NFA_i([a,b]) = \begin{cases} K \cdot \mathcal{B}\left(N_c, H^c(a,b), \frac{H_i^r(a,b)}{N_c}\right) & if H^c(a,b) \geq H_i^r(a,b) \\ K \cdot \mathcal{B}\left(N_c, N_c - H^c(a,b), 1 - \frac{H_i^r(a,b)}{N_c}\right) & if H^c(a,b) < H_i^r(a,b) \end{cases} \tag{2.20}$$

where:

$$K = \frac{L_c(L_c+1)}{2} \tag{2.21}$$

and $\mathcal{B}(\alpha, \beta, \gamma)$ denotes the binomial tail:

$$\mathcal{B}(\alpha,\beta,\gamma)=\sum_{j=\beta}^{\alpha}\left(\begin{array}{c}\alpha\\ \beta\end{array}\right)\gamma^{j}(1-\gamma)^{\alpha-j} \tag{2.22}$$

An interval $[a,b]$ is said to be an ε-meaningful rejection for the increasing hypothesis on $[s(i-1);t]$ if

$$NFA_i\left([a,b]\right)\leq\varepsilon \tag{2.23}$$

In the same way the $NFA_i\left([a,b]\right)$ is calculated in order to verify the decreasing hypothesis on $[t;s(i+1)]$. Grompone and Jakubowicz have shown in [22] that the expectation of the ε-meaningful events could be approximated by $\varepsilon/100$. Once the segmentation process is completed, a new smoothed histogram is obtained where there are no local minima and the modes are clearly distinguished by minimum points that represent the salient moments separation points of the collection analyzed.

2.7 Experimental Results

In order to test the application fist several validating tests on the unimodal clustering algorithm have been performed. In particular content-based and face clustering procedures have been evaluated in order to obtain quantitative and comparable results. Second, tests on the segmentation procedures have been conducted to evaluate the event salient moment summary.

2.7.1 hierarchical content clustering

To evaluate the clustering algorithm, a ground-truth of the Wang database[5] was built and the clustering performance were evaluated in terms of *precision*, *recall* and F_1, as in the following equations:

$$precision\left(C_h\right)=\frac{N_{C_h}^{rel}}{N_{C_h}} \tag{2.24}$$

$$recall\left(C_h\right)=\frac{N_{C_h}^{rel}}{N_{C_k}^{gt}} \tag{2.25}$$

$$F_1\left(C_h\right)=2\cdot\frac{precision\left(C_h\right)\cdot recall\left(C_h\right)}{precision\left(C_h\right)+recall\left(C_h\right)} \tag{2.26}$$

where C_h is the selected h cluster coming from the summarization, N_{C_h} is the number of images in the h cluster obtained from the hierarchical agglomerative clustering, and $N_{C_h}^{rel}$ corresponds to relevant image number in the h cluster, which is the intersection value between the output cluster and the corresponding ground-truth cluster. The total number of images of the ground-truth cluster k is $N_{C_k}^{gt}$. It has to

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

be pointed out that, in order to obtain an objective performance result, the clusters compared to the ground-truth are those that the system creates at the second level of clustering. All the clusters coming from the algorithm of section 2.4.1 are compared with all the ground-truth clusters and the coupling with the highest precision and recall are taken into account for an average statistic presented in chart of figure 2.9. The red bins are the average precisions, the blue bins are the recalls and the green ones are the F_1 values obtained on the Wang database by four different agglomerative approaches. The first group of statistics ("local") is obtained by our agglomerative approach using only the matrix of distances D_1; the second group ("global") is obtained in the same way but using only the matrix of distances D_2, while the third group are the statistics obtained with the hierarchical approach. It is possible to note that the precision-recall obtained by the hierarchical approach takes advantage from the two types of distances and features used and the way they are combined. Using only the region features vectors, we obtain clusters made by near duplicated images, then the precision is very high despite of a very low recall. On the other hand, using just the global distribution of the content features it is possible to obtain a good recall with a significant decrease in the precision. The combination of the two distributions in a hierarchical way produces a stabilization of the performance, with similar percentages for precision (about 80%) and recall (about 70%). The last group of statistics is obtained by a different clustering algorithm, called Dominant Set Clustering [95], applied to the same images features. Tests show that the performance of the two hierarchical approaches are comparable.

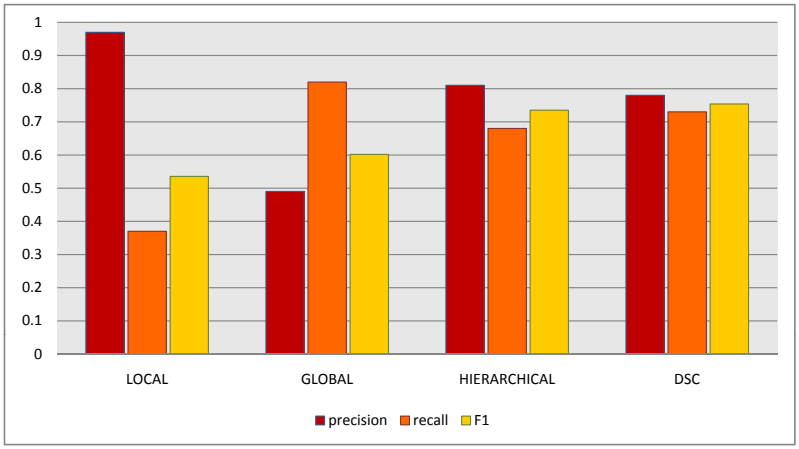


Figure 2.9: Quantitative evaluation and comparison of the hierarchical agglomerative clustering.

2.7.2 face clustering

To validate the face clustering algorithm and the generic setting for the thresholds τ_f and τ_f, we analyzed the performance of the proposed method on a large, wellknown

set of diverse news video content. From the TRECVID2005 dataset, 60 news programs (a total of 30 hours of content) were processed and the major cast members of each video was labeled [110]. This dataset is particularly challenging because of the diversity of cast members and production rules utilized in these programs from English, Arabic, and Chinese language channels. Figure 2.10 plots the precision, recall and F_1 performances of the face clustering algorithm for the five most frequent cast members. Comparing these results to similar literature using supervised techniques, our algorithm provided reasonable clustering results with an average precision of more than 85% and a recall of 71% (versus an SVM with 95% precision and 83% recall [2]).

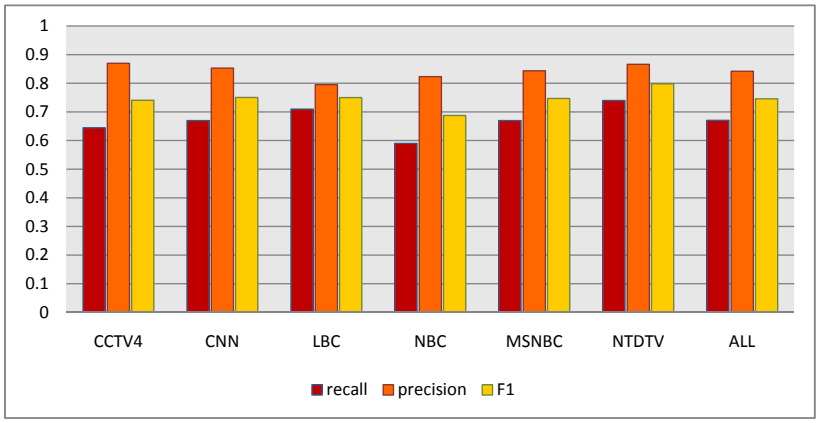


Figure 2.10: Precision, Recall and F_1 measure histograms of the face clustering algorithm of the 5 most present anchorpersons in the video.

2.7.3 event segmentation

Since there is no standard dataset with user generated content for research on personal photo collection, we built a user generated dataset with more than 6000 photos. The database is made by 10 different galleries each of them represents an event that could have duration of one day, of a week-end ore of an entire week of holiday. The ground truth is the result of the AND logic operation on the segmentation made by at least two people that have taken part at the specific event. In figure 2.11, the output histogram of a collection of 420 pictures about a mountain trip is presented. The blue histogram corresponds to the Grenander estimator of the input histogram of figure 2.14 at the end of the elimination of detected local minima. The remaining minima, which are separating the modes of the histogram, are highlighted with the red lines, while the event separation ground truth is marked by green peaks. Purple lines are the separating points using the method proposed in [30].

Table 2.1 summarizes our experiments computing precision and recall of the event separating points. In order to compare our approach with other algorithms

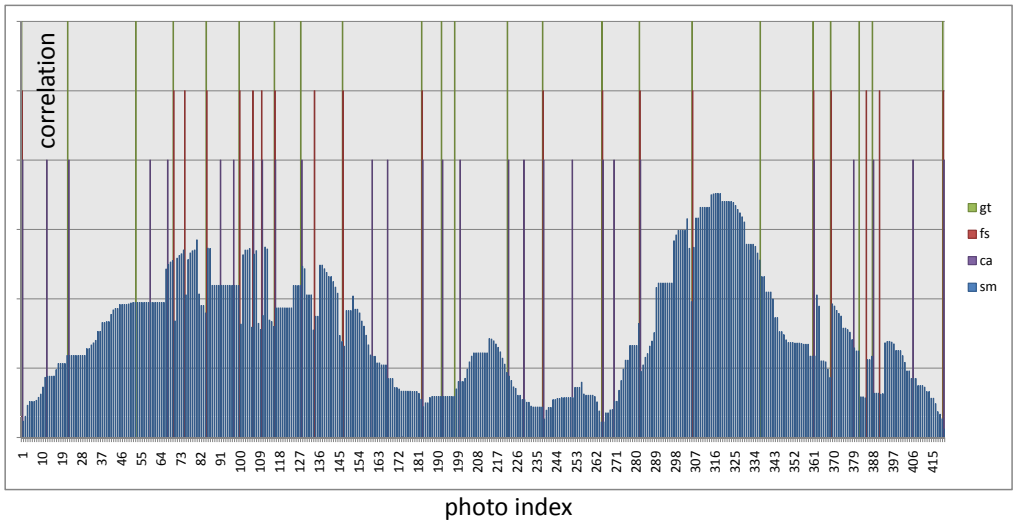


Figure 2.11: Output story histogram with event segmentation.

weights mode	***precision***	***recall***
equal	0.56	0.53
exponential mixed order	0.72	0.64
linear mixed order	0.73	0.69
linear time priority	0.77	0.59
linear content priority	0.65	0.71
linear mixed order *relaxed*	0.81	0.76
algorithm [30]	0.59	0.67
algorithm [30] *relaxed*	0.68	0.71

Table 2.1: Quantitative precision recall comparison of event segmentation algorithm.

in the presented test we exploit only time and content information excluding face and spatial clustering since there is no algorithm that takes into consideration all these information to obtain salient moments. Mixed order means that the output clustering matrices $M_C^{\alpha}(\cdot)$ are summed in the following order: local time *lt* local content *lc*, global time *gt* and global content *gc* output clustering results. First we compare three different ways of merging the clustering information: with *equal* weights $w^{\alpha} = 1$ for $\alpha = lt,\ lc,\ gt,\ gc$ (see equation 2.16 of section 2.5), with *linear* decreasing weights and with exponential weights. Weighting the information brings a performance improvement of 10% both in the precision and in the recall, with the *linear* method which outperforms the *exponential* one. Second, we change the order between content and time exchanging *lc* and *gt* order in the weighted sum. Stressing time importance, putting both clustering time results before content clustering output, the system finds more correct separating points but lacks in finding the scene

changes due to a strong discontinuity in visual features. On the contrary, putting first both the contents clustering values in the weighted sum, the system finds more separating points but the recall performances decrease. The mixed order of the clustering output values results the best combination that generally approaches better the user idea of the event. Our modular approach outperforms also the performances of the algorithm proposed in [30] that uses visual (DCT coefficients) and time information. It has to be pointed out that, if the ground truth segmentation points are relaxed of an interval of 1% of the entire time of the collection (e.g. on a 12h gallery is about 7 min), the performances increase sensitively.

2.7.4 event summarization example

A test case example is showed in this section. A mountain journey of two days, composed by 420 images, has been selected and a mosaic image of the event photos is reported 2.15. First the correlation histograms are showed: in figure the backward correlation $H^b(\cdot)$, in figure the forward correlation $H^f(\cdot)$, while in figure the final correlation histogram $H^c(\cdot)$. Each axis value refers to the gallery photo index temporally ordered.

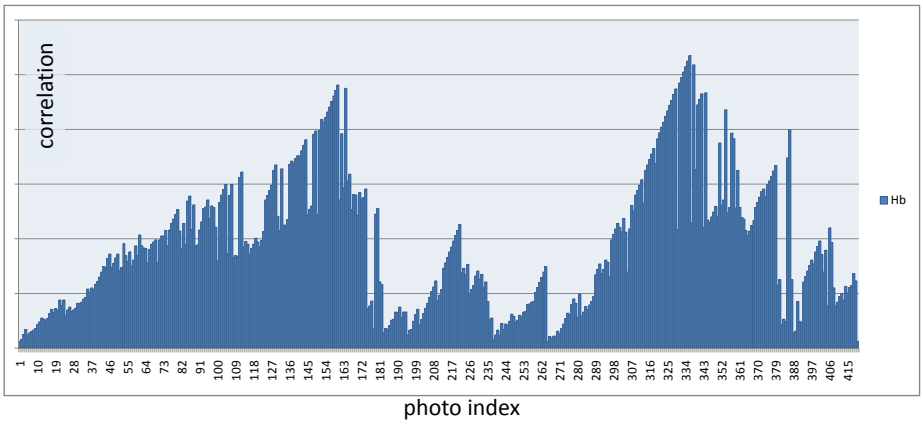


Figure 2.12: Backward correlation story histogram.

Figure 2.16 shows the face clustering results. It is possible to note that in figure a misclassified face is present, this is due to the fact of the glass presence and the similarity between the two persons erroneously combined. This clustering helps in the summarization of the event, ranking the people involved according to the number of photos related to the faces. Some detected faces but not classified in the right cluster are depicted in figure 2.17; it is possible to note that all these faces have some occlusions or the face pose is not exactly frontal, these characteristics reflect differently into the Local Binary Pattern features domain preventing a correct clustering. Figures 2.18, shows how the cluster is visualized on a map using the GPS information. Instead of positioning each single photo in the correspondent GPS coordinates point, a Longitude and Latitude centroid is associated to each sub-

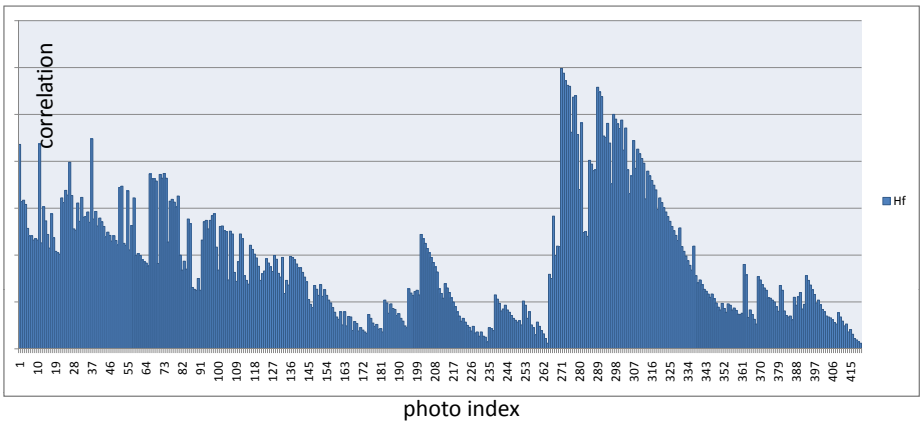


Figure 2.13: Forward correlation story histogram.

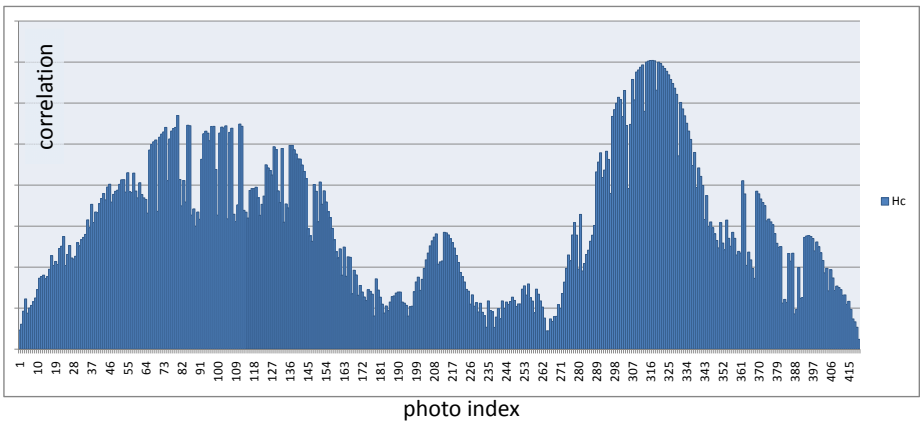


Figure 2.14: Combined correlations story histogram.

event detected. This kind of summarization helps user in the location annotation task. The sub-event mosaic images are also attached. It is possible to note that sub-event 1 (figure 2.19) can be further subdivided into 2 different moments (as the ground truth) but in this case the histogram segmentation algorithm fails filtering the minimum that separates the two modes (see input histogram 2.14 and output resulting histogram 2.11). There are also some semantic over-segmentations (figures 2.23, 2.24 and 2.25), indeed the algorithm splits into different sub events images belonging to the same picture topic, but clusters created result consistent in most of the cases both in time and in content (except for the above mentioned sub event 1). Sub-event 2.31 starts with some images strictly related to the images of the previous sub event of figure 2.31, even if this could be considered a "visual" error, it has to be pointed out that the two sub-events belong to completely different moments, in fact one situation is captured during night and the second is settled in the morning.

Figure 2.15: Mosaic image of the event photos.

(a) Face clustering result of the most present person in the gallery.

(b) Face clustering result of the second most present person in the gallery.

(c) Face clustering result of the third most present person in the gallery (error inside).

Figure 2.16: Face clustering result of the selected event.

Figure 2.17: Detected faces but not associated to any clusters.

Figure 2.18: Sub events visualization on the map.

Figure 2.19: Mosaic image of the sub event 1.

Figure 2.20: Mosaic image of the sub event 2.

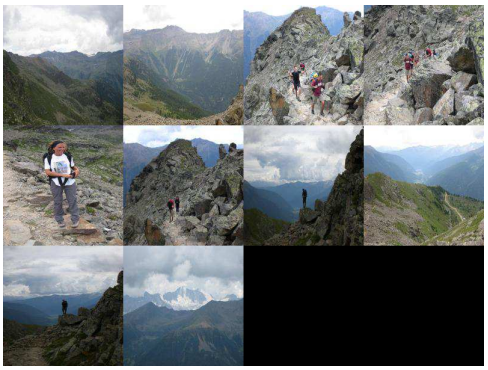


Figure 2.21: Mosaic image of the sub event 3.

Figure 2.22: Mosaic image of the sub event 4.

Figure 2.23: Mosaic image of the sub event 5.

Figure 2.24: Mosaic image of the sub event 6.

Figure 2.25: Mosaic image of the sub event 7.

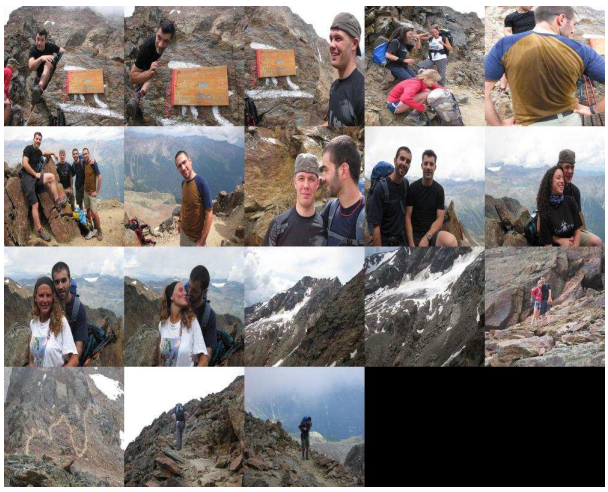


Figure 2.26: Mosaic image of the sub event 8.

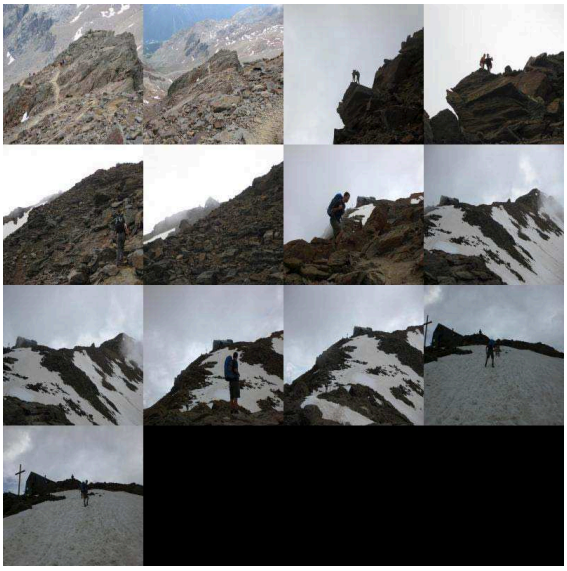


Figure 2.27: Mosaic image of the sub event 9.

Figure 2.28: Mosaic image of the sub event 10.

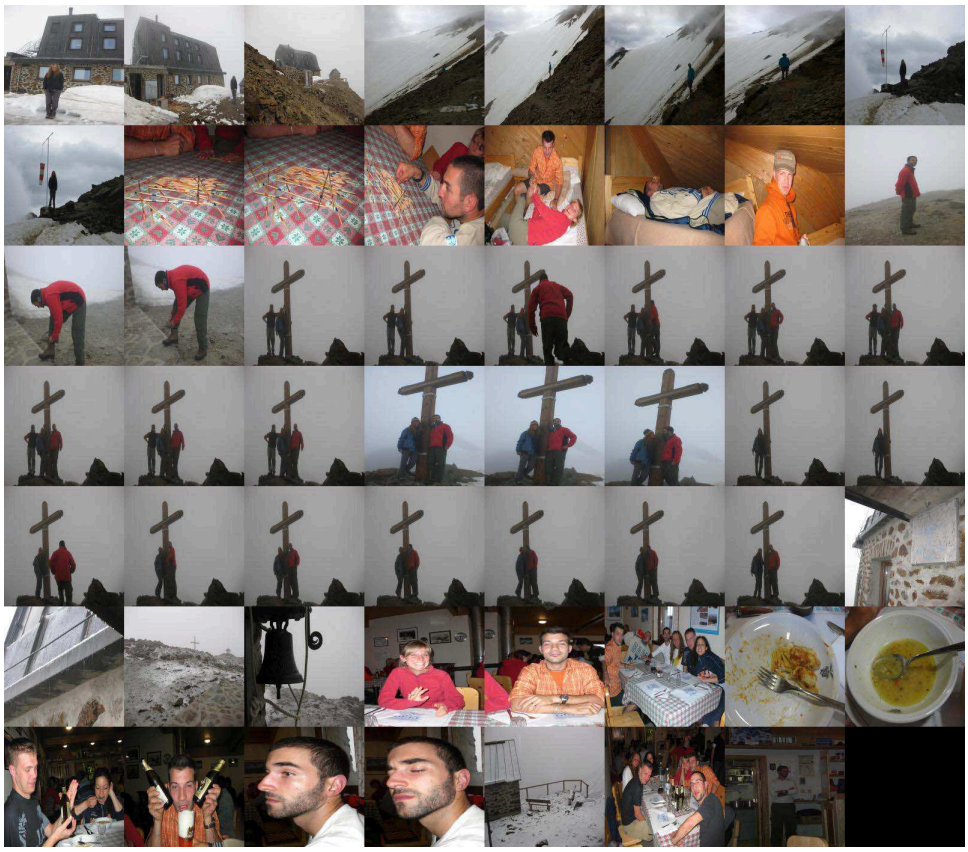


Figure 2.29: Mosaic image of the sub event 11.

Figure 2.30: Mosaic image of the sub event 12.

Figure 2.31: Mosaic image of the sub event 13.

Figure 2.32: Mosaic image of the sub event 14.

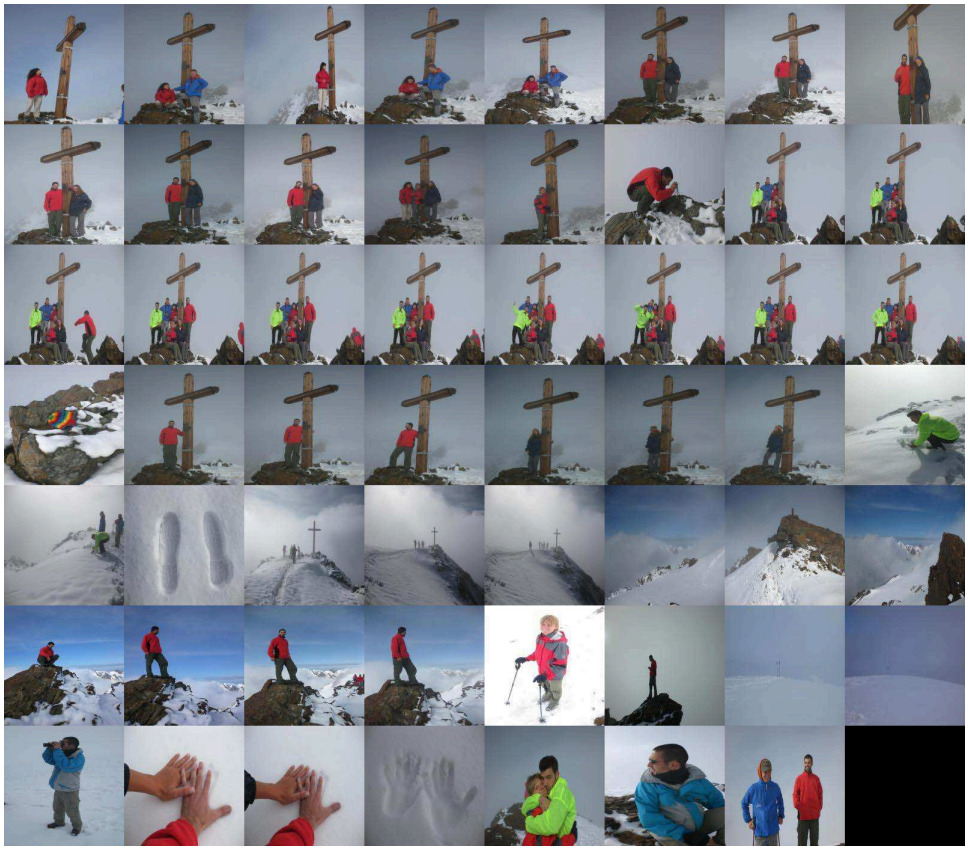


Figure 2.33: Mosaic image of the sub event 15.

Figure 2.34: Mosaic image of the sub event 16.

Figure 2.35: Mosaic image of the sub event 17.

Figure 2.36: Mosaic image of the sub event 18.

Figure 2.37: Mosaic image of the sub event 19.

Chapter 3

Multiple Photos Galleries Synchronization

The large diffusion of photo cameras makes quite common that an event is acquired from different devices, conveying different subjects and perspectives of the same happening. Often, these photo collections are shared among different users through social networks and networked communities. Automatic tools are more and more used to support the users in organizing such archives, and it is largely accepted that time/space information is fundamental to this purpose. Unfortunately, both data are often unreliable, and in particular, timestamps may be affected by erroneous or imprecise setting of the camera clock. In this chapter a synchronization algorithm is presented that uses the content of pictures to estimate the mutual delays among different cameras, thus achieving an a-posteriori synchronization of various photo collections referring to the same event. Experimental results show that, for sufficiently large archives, a notable accuracy can be achieved in the estimation of the synchronization information.

3.1 Motivations

Life is made by events and taking pictures is the most popular way to maintain memories of what is happening [132] [112]. Modern digital cameras made it easier and cheaper to collect large photo galleries of daily life. Different tools are available to organize and share all those contents, (e.g. Picasa[1] , iLife[2] or and Windows Media Center[3]); such tools provide basic functionalities to ease the user in image cataloguing, including face recognition, geo-referencing, time ordering. Nevertheless, an issue that is becoming more and more relevant among users is concerned with the reliability of contextual information stored with the picture. In particular,

[1] www.picasa.google.it
[2] www.apple.com
[3] http://windows.microsoft.com

since the timestamp is one of the most valuable data to order and catalog photos [47], its accuracy is of great importance. In particular, this problem becomes critical when several independent users want to share the pictures acquired with their own devices at the same event. This is more and more common both in large-scale events (e.g., sports, music, etc.), where networked communities of users share their contents about some theme of common interest, and in personal life, where relatives or friends want to bring together their photo collections to create a unique chronologic storyboard of a joint event. Often, however, the timestamp stored in pictures is affected by a wrong setting in the camera, thus introducing a de-synchronization among different datasets, and consequently significant errors in the following temporal analysis [67]. Annotation [17], [75] summarization [109], [79] event cataloguing [30], [18] and automatic album creation [100], [85], [98] are deeply connected to the timestamp of the photos. All these applications work well on a single camera, but suffer from lack of synchronization. For instance, a bad synchronization among cameras makes impossible to define and understand salient moments of an event, grouping correctly pictures related in time and content, create summaries and storyboards. Manual recovery of the synchronization is a boring task, and the result may be imprecise if no precise triggering instant can be found. Let us consider the following scenario. Several people went to a wedding and, after the party, the guest of honor wants to collect the photos taken by every other guests. Probably many photographers have shot pictures at key moments, such as, for instance, ring exchange, spouses kiss, or cutting of the wedding cake. If all these pictures could be collected in a single chronological sequence, summarization algorithms can easily select most significant shots and build a summary. On the contrary, non synchronized pictures will interlace each other, making very difficult to assemble them without a complex manual work. We try to solve this problem by automatically estimating the delay between photos coming from different cameras, based on the analysis of their content. The only a-priori assumption is that each camera has a coherent clock within the whole sequence. The method tries to detect the most significant associations between similar pictures in different galleries, to calculate a set of delay estimates, which are then combined through a statistical procedure. To the best of our knowledge, this is the first attempt to solve this problem exploiting the visual content only.

3.2 Proposed Approach

Figure 3.1 outlines the proposed algorithm which is made up of three main phases:

1. region color and textures matching
2. salient points matching
3. estimation of the delay

The main idea of the algorithm consists in finding the maximum possible number of pairs of similar pictures among different galleries. Such photos should probably

refer to the same episodes taken from different photographers, and therefore reveal to some extent the delay among the time settings of the two devices. Since however the duration of every single episode may vary (is not instantaneous) to achieve a sufficiently accurate estimation one needs to find an adequate set of correct photos pairs that can confirm the same time delay. For this reason we split the matching process into two steps. In the first step the algorithm matches two different galleries according to the features that describe the scene. This matching process selects a few candidates from the entire set of images, which could have been taken at the same time instant. The objective of this first step is to limit as much as possible false positives. The second step takes as input the candidates found in step 1, and further filters the relevant photos pairs by matching their SURF salient points [11]. Finally, the delay is estimated on the remaining data.

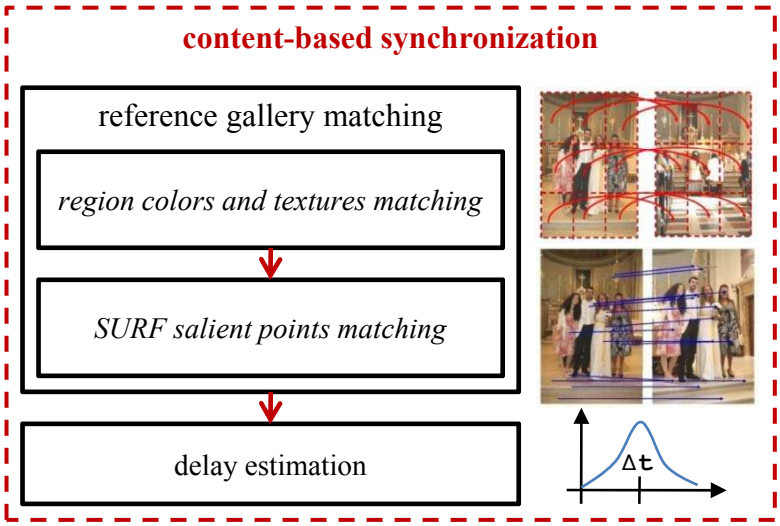


Figure 3.1: Content based synchronization algorithm.

3.2.1 Region color and texture matching

Let C be a collection of photos albums $\{C^j\}, j = 1, \ldots, J;$ taken by J different cameras. Let's call the i-th image of the j-th album as $c_i^j, i = 1, , I^j;$ I^j being the number of images of the j-th album. As a first step, the system extracts from each image c_i^j a set of 9 CEDD vectors [23] $\overline{x}_{ir}^j$, $r = 1, , 9;$ related to 9 non-overlapping sub-images. Each vector is made of 144 features representing a set of color and texture statistics [24]. After that, a reference gallery C^{j*} is selected, and the average region similarity is calculated between images in C^{j*} and all other photos, according to equation 3.1:

$$D\left(c_i^{j*}, \widehat{c}_k^j\right) = \frac{1}{9}\sum_{r=1}^{9} \frac{\overline{x}_{ir}^{j*T} \cdot \overline{x}_{kr}^j}{\overline{x}_{ir}^{j*T} \cdot \overline{x}_{ir}^{j*} + \overline{x}_{kr}^{j\ T} \cdot \overline{x}_{kr}^j - \overline{x}_{ir}^{j*T} \cdot \overline{x}_{kr}^j} \tag{3.1}$$

where $\widehat{c}_k^j \in \{C/C^{j*}\}$ is the whole set of photos excluding the reference gallery C^{j*}. $D(\cdot)$ expresses the average Tanimoto coefficient [42] of the corresponding sub-images and expresses the global similarity among the two pictures. In order to reduce the false positives while keeping enough samples for delay estimation, the photo pairs are further filtered by discarding the image pairs whose coefficient D is lower than a given threshold t_h. This value is calculated from the empirical distribution function (EDF)[51] (equation 3.2:

$$EDF_n(t_h) = \frac{1}{N}\sum_{n=1}^{N} K_{[D(n)<t_h]} = 0,2 \tag{3.2}$$

where N is the total number of image pairs calculated at the previous step, $D(n)$ are the relevant D values, and K is the so-called indicator random variable, which is 1 when the property $[D(n) < t_h]$ holds, and 0 otherwise. In other words, among all the admissible photos pairs, only the 20% with lower D are kept for further analysis. Figure 3.2 shows an example of EDF (blue line) and the relevant histogram of the D values calculated on an event made of more than 800 images. The red dot highlights the selected t_h value.

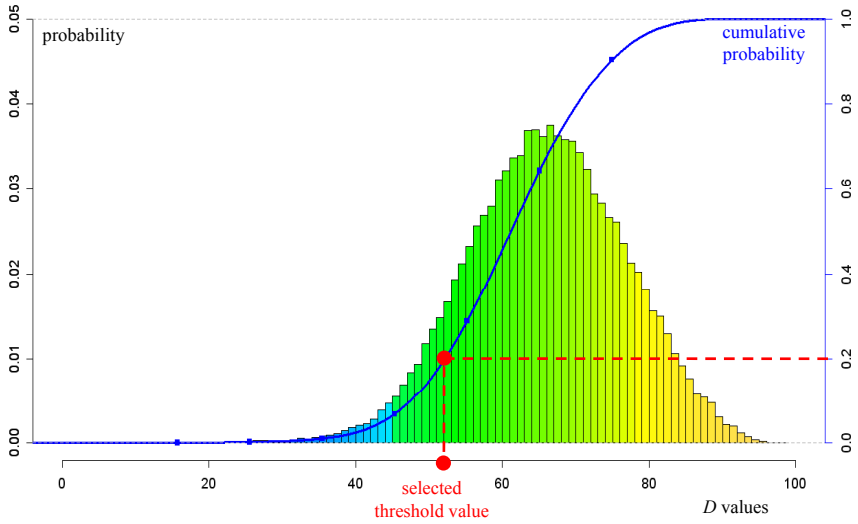


Figure 3.2: Cumulative distribution function and the relevant histogram calculated on a set of test images.

3.2.2 SURF salient points matching

Once color and texture matching is completed, local points content descriptors are extracted from the candidate photo pairs. SURF descriptors were chosen for their compact representation (64 features for each key-point) and fast computation. The matching procedure is applied to the previously detected photos pairs $\{c_i^{j*}, \widehat{c}_k^j\} | D\left(c_i^{j*}, \widehat{c}_k^j\right) < t_h$ and is based on the method proposed by Lowe [86]. Here,

the nearest neighbor of a feature descriptor is calculated, and the second closest neighbor is checked to see if its distance is higher than a pre-defined threshold. The nearest neighbor computation is based on the Euclidean distance between the descriptors. To complete the matching, three other filters are applied to the matching points of the selected photos pairs:

- all the matches that are not unique are rejected;
- the matching points whose scale and rotation do not agree with the majority's scale and rotation are eliminated;
- photos pairs with less than a given number of matching points are discarded.

3.2.3 Delay estimation

The last step is the estimation of the delay. The timestamp delay between each photos pair coming from the previous process is calculated using equation 3.3:

$$\Delta\left(c_i^{j*}, \widehat{c}_k^{j}\right) = t_{c_i^{j*}} - t_{\widehat{c}_k^{j}}; \quad \widehat{c}_k^{j} \in \{C/C^{j*}\} \tag{3.3}$$

where $t_{c_i^{j*}}$ and $t_{\widehat{c}_k^{j}}$ are the timestamps of the reference and current photos, respectively, extracted from the $EXIF$. The calculated delays are then split according to the j-th photo galleries and the most frequent delay $\Delta_m(C^{j*}, C^j)$ is calculated. Then, $\Delta_m(C^{j*}, C^j)$, includes all the image pairs whose delay is estimated within the relevant 1 minute window. Finally, the delays of the photo pairs in the 1 minute window found are averaged to find the output estimated delay $\Delta_o(C^{j*}, C^j)$ using equation 3.4 where $f_{\Delta_m}(C^{j*}, C^j)$ is the number of photos pairs of the j-th gallery inside $\Delta_m(C^{j*}, C^j)$ 1 minute window.

$$\Delta_o\left(C^{j*}, C^j\right) = \frac{1}{f_{\Delta_m}(C^{j*}, C^j)} \sum \Delta\left(c_i^{j*}, \widehat{c}_k^{j}\right) \in \Delta_m\left(C^{j*}, C^j\right) \tag{3.4}$$

$\Delta_o(C^{j*}, C^j)$ represents the estimated delay in terms of years, days, hours, minutes and seconds between the reference gallery C^{j*} and the j-th gallery of the same event. An example of the time delay histograms (with 1 minute temporal quantization) is depicted in figure 3.3 where, for each gallery, the most frequent delays $\Delta_m(C^{j*}, C^j)$ is highlighted. Since the accuracy of the estimation may be limited by the size of the galleries (when few images are available it may be difficult to find a sufficient number of reliable photo pairs) the overall synchronization accuracy can be further increased by adding to the reference the photos of the new galleries just synchronized. To this purpose, a precision coefficient $p_{\{C^j\}}$ of the estimated time delay is calculated for each synchronized gallery according to equation 3.5:

$$p_{C^j} = \frac{f_{\Delta_m}\left(C^{j*}, C^j\right)}{\sigma_{\Delta\left(c_i^{j*}, \widehat{c}_k^{j}\right)}}; \quad j = 1, \ldots, J; \quad j \neq j^* \tag{3.5}$$

where $f_{\Delta_m}(C^{j*}, C^j)$ is the frequency of the photos pairs in $\Delta_m(C^{j*}, C^j)$ and $\sigma_{\Delta\left(c_i^{j*}, \widehat{c}_k^{j}\right)}$ is the variance of all the acquisition delays found with respect to $\Delta_o(C^{j*}, C^j)$.

The gallery with the highest precision coefficient $p_{\{C^j\}}$ is synchronized, adding or subtracting $\Delta_o\left(C^{j*}, C^j\right)$ (years, days, hours, minutes, seconds) and used as a reference gallery for the remaining sets of photos (with reference to the example in figure 3.3, gallery C^1 is synchronized and merged with C^{j*}). Since more photos are included in the reference collection C^{j*}, the following collections could benefit of an increased number of matches, thus gaining a higher accuracy.

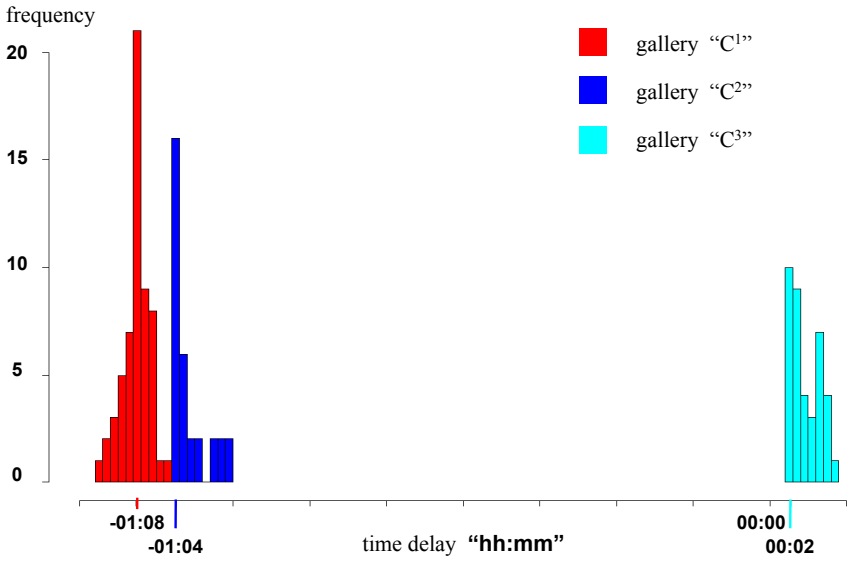


Figure 3.3: Time delay histogram of a set of photo pairs belonging to three different galleries.

3.3 Experimental Results

A user-generated dataset with more than 6.000 photos was built. The database is made of 10 different collections, each of which representing an event with different possible durations (a day, a week-end, a full week). Each collection is made of photos coming from at least 3 different cameras, for a total of 40 galleries. The desynchronization of cameras was simulated by inserting random delays in the galleries, modifying year, day, hour, minute and seconds. As far as the SURF salient point matching is concerned, we stress the importance of reducing false positives to obtain a set of highly reliable photo pairs. For this reason, we calibrated the parameters as follows: for the matching we search the second nearest neighbor till 20 leaves of the $k-d$ tree; the distance different ratio for which one match is considered unique is set to 0.4. The difference in scale for neighborhood bins is set to 1.5 (which means that matched features in bin $b+1$ are scaled of a factor 1.5 with respect to features in bin b) while the number of bins for rotation is fixed to 20 (which means that each

bin covers 18 degrees). Finally, we keep only the photo pairs with at least 10 correct matches.

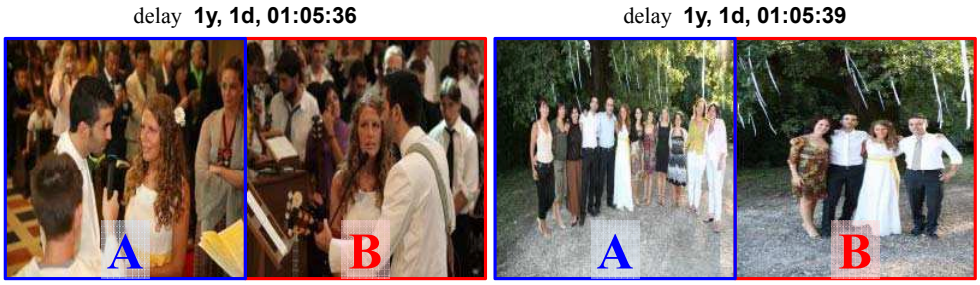


Figure 3.4: Examples of false positives using only SURF matching between gallery A (reference) and B

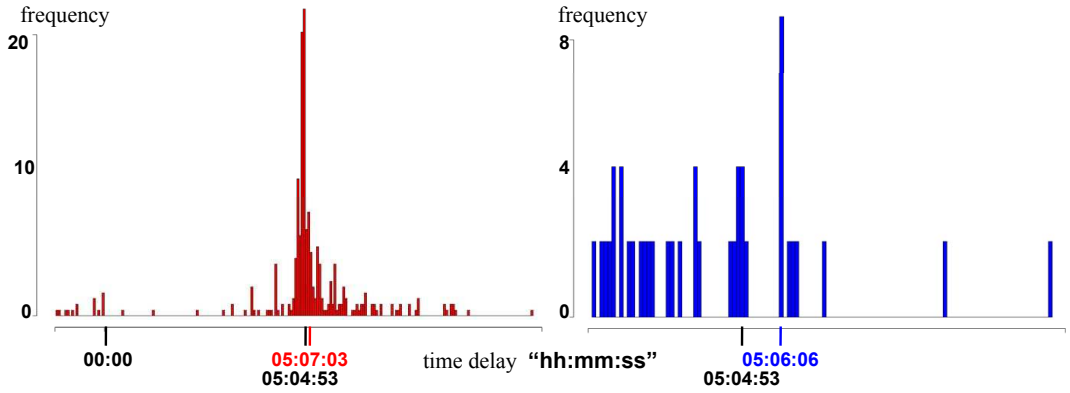


Figure 3.5: Photo pairs delay histograms using only SURF matching (red) and SURF+CEDD matching (blue).

Table 3.1 shows the result of the synchronization algorithm: the error estimation columns represent the difference between the real and estimated delays, averaged among different galleries of the same event. Two different experiments are reported: "SURF matching error" column is the error obtained without applying the first step of global features matching, while "CEDD+SURF matching" column reports the error results using the 20% filtering on the D distances. It is possible to observe that in the second case, the accuracy of the estimated delay increases considerably, due to the initial filtering of false positives, however the algorithm fails in 6 photo sets since in those cases no valid photo pairs survived after the two steps of matching. The average delay estimation error for the other galleries is around 2 minutes. On the other hand, the use of SURF alone allows synchronizing all the galleries, but results is a lower accuracy due to false positives (e.g., due to the presence of the same objects or persons in different context, as in the example of figure 3.4). The average delay estimation error in this case is around 16 minutes, which is anyway

acceptable for events with duration of several hours or days.

Figure 3.6: Examples of true positives photos pairs between gallery A (reference) and B with the corresponding delay.

An example of gallery synchronization is presented in figure 3.5: 5 hours 4 minutes and 53 seconds is the inserted de-synchronization, the red histogram corresponds to the estimation using only the SURF matching, while the blue histogram shows the estimated delay using the proposed approach. In this case, the error decreases from 0:02:10 to 0:01:13, gaining about 1 minute in accuracy. Figure 3.6 presents four true positives photos pairs with a delay within the $\Delta_m(A, B)$ one minute time window.

event type	**gall.**	**photos**	**duration**	**SURF matching est. error *h:mm:ss***	**gall. failed**	**CEDD+SURF matching est. error *h:mm:ss***	**gall. failed**
wedding	7	1173	2 days	0:40:32	0	0:02:47	1
wedding	7	937	1 day	0:09:10	0	0:03:56	1
wedding	4	644	1 day	0:23:03	0	0:01:25	1
trip	4	659	4 days	0:05:41	0	0:01:52	1
trip	3	526	3 days	0:06:33	0	0:02:26	0
trip	3	1148	1 week	0:04:13	0	0:05:34	0
graduation	3	307	1 day	0:04:09	0	0:02:03	0
graduation	3	211	1 day	0:58:17	0	0:01:49	1
journey	3	271	1 day	0:01:19	0	0:01:28	0
journey	3	262	1 day	0:07:10	0	0:01:13	1
total	**40**	**6138**	**average**	**0:16:06**	**0%**	**0:02:27**	**20%**

Table 3.1: Average delays estimation errors.

Conclusions

In order to improve existing solutions for personal photo albums management, in this dissertation three new content-based tools, for help the user in the retrieval and annotation task, have been presented.

The possibility of embedding the Relevance Feedback (RF) process into a stochastic optimization engine, namely Particle Swarm Optimization (PSO), has been investigated in chapter 1. PSO algorithm resulted to be able to avoid the stagnation in local minima during the retrieval process. Extensive simulations showed that the proposed technique outperforms traditional deterministic RF approaches of the same class, thanks to its stochastic nature, which allows a better exploration of complex, non-linear and highly-dimensional solution spaces. Further developments will try to insert into the loop explicit semantic knowledge (in the form of annotation) together with an unsupervised clustering of the database. Furthermore, it is worth mentioning that standardized methodologies to allow measuring the user satisfaction based on subjective criteria are still missing. Studies are being carried on to place the proposed RF methodologies in a framework for human-oriented testing and assessment.

In chapter 2 an multi-modal event segmentation method has been proposed, the system developed subdivides the considered photo gallery in salient moments exploiting unsupervised algorithms of clustering and histograms segmentation. Leveraging different types of content and context information the uni-modal correlation of each signal is analyzed and fused together in a second compositional mining phase. The algorithm analyzes the consistency in terms of time, spatial and visual content across the gallery to detect the major points of discontinuity, which may identify the transition to a different episode in the event description. The created summary reflects very good quantitative results compared with user judge. Further work will include more subjective tests and will explore different types of weighting the clustering output adding also location and face information.

A content-based synchronization algorithm has been presented in chapter 3 with the aim of providing an estimation of the time delay between photo galleries of the same event coming from different cameras. The method is the first attempt to solve this problem based on picture content, and is based on the hypothesis that photographers involved in the same event often take photos of the same subevents. Performed tests show that the proposed algorithm was able to correctly

synchronize about 80% of the considered galleries, with an average delay error of about 2 minutes. The achieved estimation can be used as an interactive support for users in synchronizing different photo archives describing the same event, or as an automatic tool to enable the automatic creation of digital storyboards from multiple galleries. Future work includes the extension of the algorithm to videos coming from different camcorders and the correct temporal link between videos and photos of the same event.

The unsupervised approaches adopted brought promising results both in the retrieval and in the event segmentation avoiding curse of dimensionality and any sophisticated training phase. All the methods proposed exploit image content similarity without direct machine decision rules, letting the user to be the leading actor of the relations among data. This brought a roundabout involvement of the user context improving the usefulness of the proposed applications.

Bibliography

[1] A survey of methods for image annotation. *Journal of Visual Languages & Computing*, 19(5):617 – 627, 2008. 28

[2] W. Hsu A. Yanagawa and S.-F. Chang. Anchor shot detection in trecvid-2005 broadcast news videos. Technical report, Columbia University, 2005. 44

[3] Timo Ahonen, Abdenour Hadid, and Matti Pietikinen. Face recognition with local binary patterns. In Toms Pajdla and Jir Matas, editors, *Computer Vision - ECCV 2004*, volume 3021 of *Lecture Notes in Computer Science*, pages 469–481. Springer Berlin, 2004. 37

[4] Timo Ahonen, Abdenour Hadid, and Matti Pietik?inen. Face description with local binary patterns: Application to face recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28:2037–2041, 2006. 37

[5] Selim Aksoy and Robert M. Haralick. Feature normalization and likelihood-based similarity measures for image retrieval. *Pattern Recognition Letters*, 22(5):563 – 582, 2001. 17

[6] E. Ardizzone, M. La Cascia, and F. Vella. Mean shift clustering for personal photo album organization. In *Image Processing, 2008. ICIP 2008. 15th IEEE International Conference on*, pages 85 –88, 2008. 29

[7] Edoardo Ardizzone, Marco La Cascia, and Filippo Vella. Unsupervised clustering in personal photo collections. In Marcin Detyniecki, Ulrich Leiner, and Andreas Nrnberger, editors, *Adaptive Multimedia Retrieval. Identifying, Summarizing, and Recommending Image and Music*, volume 5811 of *Lecture Notes in Computer Science*, pages 140–154. Springer Berlin / Heidelberg, 2010. 29

[8] La Cascia M. Ardizzone E. and Vella F. A novel approach to personal photo album representation and management. In *20th Annual IS&T/SPIE Symposium on Electronic Imaging*, 2008. 29

[9] Miriam Ayer, H. D. Brunk, G. M. Ewing, W. T. Reid, and Edward Silverman. An empirical distribution function for sampling with incomplete information. *The Annals of Mathematical Statistics*, 26(4):pp. 641–647, 1955. 41

[10] Ricardo A. Baeza-Yates and Berthier Ribeiro-Neto. *Modern Information Retrieval*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1999. 6, 18

[11] H. Bay and A. L. Tuytelaars, T.and Van Gool. SURF: Speeded Up Robust Features. In *9th European Conference on Computer Vision*, May 2006. 63

[12] Lucien Birge. The grenader estimator: A nonasymptotic approach. *The Annals of Statistics*, 17(4):pp. 1532–1549, 1989. 41

[13] Gloria Bordogna and Gabriella Pasi. A user-adaptive neural network supporting a rule-based relevance feedback. *Fuzzy Sets and Systems*, 82(2):201 – 211, 1996. Connectionist and Hybrid Connectionist Systems for Approximate Reasoning. 7

[14] D. Bratton and J. Kennedy. Defining a standard for particle swarm optimization. In *Swarm Intelligence Symposium, 2007. SIS 2007. IEEE*, pages 120 –127, 2007. 20

[15] M. Broilo, P. Rocca, and F.G.B. De Natale. Content-based image retrieval by a semi-supervised particle swarm optimization. In *Multimedia Signal Processing, 2008 IEEE 10th Workshop on*, pages 666–671, 2008. 15, 18

[16] S. Landau B.S. Everitt and M. Leese. *Cluster Analysis*. Arnold, 2001. 30, 31

[17] L. Cao, J. Luo, H. Kautz, and T. S. Huang. Image annotation within the context of personal photo collections using hierarchical event and scene models. *Transaction on Multimedia*, 11:208–219, February 2009. 62

[18] Liangliang Cao, Jiebo Luo, H. Kautz, and T.S. Huang. Image annotation within the context of personal photo collections using hierarchical event and scene models. *Multimedia, IEEE Transactions on*, 11(2):208 –219, 2009. 62

[19] K. Chandramouli. Particle swarm optimisation and self organising maps based image classifier. In *Semantic Media Adaptation and Personalization, Second International Workshop on*, pages 225 –228, 2007. 7

[20] K. Chandramouli and E. Izquierdo. Image classification using chaotic particle swarm optimization. In *Image Processing, 2006 IEEE International Conference on*, pages 3001 –3004, 2006. 7

[21] K. Chandramouli, T. Kliegr, J. Nemrava, V. Svatek, and E. Izquierdo. Query refinement and user relevance feedback for contextualized image retrieval. In *Visual Information Engineering, 2008. VIE 2008. 5th International Conference on*, 29 2008. 7

[22] Chein-I Chang, Kebo Chen, Jianwei Wang, and Mark L.G. Althouse. A relative entropy-based approach to image thresholding. *Pattern Recognition*, 27(9):1275 – 1289, 1994. 41, 42

[23] S. A. Chatzichristofis and Y. S. Boutalis. Cedd: color and edge directivity descriptor: a compact descriptor for image indexing and retrieval. In *Proceedings of the 6th international conference on Computer vision systems*, pages 312–322, 2008. 32, 37, 63

[24] S. A. Chatzichristofis, Y. S. Boutalis, and M. Lux. Selection of the proper compact composite descriptor for improving content based image retrieval. In *International Conference on Signal Processing, Pattern Recognition and Applications*, 2009. 63

[25] Jae Young Choi, W. De Neve, Y.M. Ro, and K.N. Plataniotis. Automatic face annotation in personal photo collections using context-based unsupervised clustering and face information fusion. *Circuits and Systems for Video Technology, IEEE Transactions on*, 20(10):1292 –1309, 2010. 37

[26] Jae Young Choi, Seungji Yang, Yong Man Ro, and Konstantinos N. Plataniotis. Face annotation for personal photos using context-assisted face recognition. In *Proceeding of the 1st ACM international conference on Multimedia information retrieval*, MIR '08, pages 44–51, New York, NY, USA, 2008. ACM. 37

[27] Wei-Ta Chu and Chia-Hung Lin. Automatic selection of representative photo and smart thumbnailing using near-duplicate detection. In *Proceeding of the 16th ACM international conference on Multimedia*, MM '08, pages 829–832, New York, NY, USA, 2008. ACM. 29

[28] Wei-Ta Chu and Chia-Hung Lin. Automatic summarization of travel photos using near-duplication detection and feature filtering. In *Proceedings of the seventeen ACM international conference on Multimedia*, MM '09, pages 1129–1130, New York, NY, USA, 2009. ACM. 29

[29] M. Clerc and J. Kennedy. The particle swarm - explosion, stability, and convergence in a multidimensional complex space. *Evolutionary Computation, IEEE Transactions on*, 6(1):58 –73, February 2002. 19

[30] M. Cooper, J. Foote, A. Girgensohn, and L. Wilcox. Temporal event clustering for digital photo collections. *ACM Transaction Multimedia Compututing Communication Application*, 1:269–288, August 2005. 29, 44, 45, 46, 62

[31] David J. Crandall, Lars Backstrom, Daniel Huttenlocher, and Jon Kleinberg. Mapping the world's photos. In *Proceedings of the 18th international conference on World wide web*, WWW '09, pages 761–770, New York, NY, USA, 2009. ACM. 29

[32] Ritendra Datta, Dhiraj Joshi, Jia Li, and James Z. Wang. Image retrieval: Ideas, influences, and trends of the new age. *ACM Comput. Surv.*, 40:5:1–5:60, May 2008. 1, 5

[33] J. Delon, A. Desolneux, J.-L. Lisani, and A.B. Petro. A nonparametric approach for histogram segmentation. *Image Processing, IEEE Transactions on*, 16(1):253 –261, 2007. 41

[34] Thomas Deselaers, Daniel Keysers, and Hermann Ney. Features for image retrieval: an experimental comparison. *Information Retrieval*, 11:77–107, 2008. 10.1007/s10791-007-9039-3. 1, 17

[35] Thomas Deserno, Sameer Antani, and Rodney Long. Ontology of gaps in content-based image retrieval. *Journal of Digital Imaging*, 22:202–215, 2009. 1

[36] D. Djordjevic and E. Izquierdo. An object- and user-driven system for semantic-based image annotation and retrieval. *Circuits and Systems for Video Technology, IEEE Transactions on*, 17(3):313 –323, 2007. 7

[37] Ramprasath Dorairaj and K.R. Namuduri. Compact combination of mpeg-7 color and texture descriptors for image retrieval. In *Signals, Systems and Computers, 2004. Conference Record of the Thirty-Eighth Asilomar Conference on*, volume 1, pages 387 – 391 Vol.1, 2004. 17

[38] Nikolaos Doulamis and Anastasios Doulamis. Evaluation of relevance feedback schemes in content-based in retrieval systems. *Signal Processing: Image Communication*, 21(4):334 – 357, 2006. 7

[39] R. O. Duda and P. E. Hart. *Pattern Classification and Scene Analysis*. Wiley, 1973. 2

[40] Eberhart and Yuhui Shi. Particle swarm optimization: developments, applications and resources. In *Evolutionary Computation, 2001. Proceedings of the 2001 Congress on*, 2001. 7

[41] Marco Morana Edoardo Ardizzone, Marco La Cascia and Filippo Vella. Clustering techniques for personal photo album management. *Journal of Electronic Imaging*, 18, December 2009. 29

[42] M. Fligner, J. Verducci, J. Bjoraker, and P. Blower. A new association coefficient for molecular dissimilarity. In *Second joint Sheffield Conference on Chemoinformatics*, 2001. 33, 64

[43] Giorgio Giacinto, Fabio Roli, and Giorgio Fumera. Adaptive query shifting for content-based image retrieval. In *Proceedings of the Second International Workshop on Machine Learning and Data Mining in Pattern Recognition*, MLDM '01, pages 337–346, London, UK, 2001. Springer-Verlag. 7

[44] D. Gies and Y. Rahmat-Samii. Reconfigurable array design using parallel particle swarm optimization. In *Antennas and Propagation Society International Symposium, 2003. IEEE*, volume 1, pages 177 – 180 vol.1, 2003. 7

[45] Andreas Girgensohn, John Adcock, Matthew Cooper, Jonathan Foote, and Lynn Wilcox. Simplifying the management of large photo collections. In *In Proc. of INTERACT03, IOS*, pages 196–203. Press, 2003. 29

[46] J. C. Gower and G. J. S. Ross. Minimum Spanning Trees and Single Linkage Cluster Analysis. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 18(1), 1969. 30, 33

[47] A. Graham, H. Garcia-Molina, A. Paepcke, and T. Winograd. Time as essence for photo browsing through personal digital libraries. In *Proceedings of the 2nd ACM/IEEE-CS joint conference on Digital libraries*, pages 326–335, 2002. 28, 62

[48] U. Grenander. *Abstract inference*. Wiley, 1981. 41

[49] G. Griffin, A. Holub, and P. Perona. Caltech-256 Object Category Dataset. Technical Report 7694, California Institute of Technology, 2007. 16

[50] A. Grigorova, F.G.B. De Natale, C. Dagli, and T.S. Huang. Content-based image retrieval by feature adaptation and relevance feedback. *Multimedia, IEEE Transactions on*, 9(6):1183 –1192, 2007. 6

[51] J. Han and M. Kamber. *Data Mining: Concepts and Techniques (The Morgan Kaufmann Series in Data Management Systems)*. 1st edition, September 2000. 33, 38, 64

[52] A. Hanjalic, R. Lienhart, W.-Y. Ma, and J. R. Smith. The holy grail of multimedia information retrieval: So close or yet so far away? *Proceedings of the IEEE*, 96(4):541 –547, 2008. 1

[53] Pierre Hansen and Brigitte Jaumard. Cluster analysis and mathematical programming. *Mathematical Programming*, 79:191–215, 1997. 10.1007/BF02614317. 29

[54] J. Hartigan and M. Wang. A k-means clustering algorithm. *Applied Statistics*, 28:100108, 1979. 30

[55] John A. Hartigan. *Clustering Algorithms*. John Wiley & Sons, Inc., New York, NY, USA, 99th edition, 1975. 29

[56] M.G. Hinchey, R. Sterritt, and C. Rouff. Swarms and swarm intelligence. *Computer*, 40(4):111 –113, 2007. 14

[57] Kyoji Hirata and Toshikazu Kato. Query by visual example - content based image retrieval. In *Proceedings of the 3rd International Conference on Extending Database Technology: Advances in Database Technology*, EDBT '92, pages 56–71, London, UK, 1992. Springer-Verlag. 1

[58] S.C.H. Hoi, M.R. Lyu, and R. Jin. A unified log-based relevance feedback scheme for image retrieval. *Knowledge and Data Engineering, IEEE Transactions on*, 18(4):509 – 524, 2006. 16

[59] T.S. Huang, C.K. Dagli, S. Rajaram, E.Y. Chang, M.I. Mandel, G.E. Poliner, and D.P.W. Ellis. Active learning for interactive multimedia retrieval. *Proceedings of the IEEE*, 96(4):648 –667, 2008. 6

[60] Mark J. Huiskes and Michael S. Lew. Performance evaluation of relevance feedback methods. In *Proceedings of the 2008 international conference on Content-based image and video retrieval*, CIVR '08, pages 239–248, New York, NY, USA, 2008. ACM. 7

[61] Ivan Ivanov, Peter Vajda, Jong-Seok Lee, and Touradj Ebrahimi. Epitome - A Social Game for Photo Album Summarization. In *Proceedings of the ACM SIGMM International Conference on Multimedia, the First ACM International Workshop on Connected Multimedia*, pages 33–38, 2010. 29

[62] K. Li J. Deng, A. Berg and booktitle = European Conference of Computer Vision (ECCV) year = 2010 L. Fei-Fei, title = What does classifying more than 10,000 image categories tell us? 2

[63] A. K. Jain, M. N. Murty, and P. J. Flynn. Data clustering: a review. *ACM Comput. Surv.*, 31:264–323, September 1999. 29

[64] Anil K. Jain. Data clustering: 50 years beyond k-means. *Pattern Recognition Letters*, 31(8):651 – 666, 2010. Award winning papers from the 19th International Conference on Pattern Recognition (ICPR), 19th International Conference in Pattern Recognition (ICPR). 30

[65] Anil K. Jain and Richard C. Dubes. *Algorithms for clustering data*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1988. 29

[66] Ramesh Jain and Pinaki Sinha. Content without context is meaningless. In *Proceedings of the international conference on Multimedia*, MM '10, pages 1259–1268, New York, NY, USA, 2010. ACM. 2

[67] C. Jang, T. Yoon, and H.-G. Cho. A smart clustering algorithm for photo set obtained from multiple digital cameras. In *Proceedings of ACM symposium on Applied Computing*, pages 1784–1791, 2009. 62

[68] Stephen Johnson. Hierarchical clustering schemes. *Psychometrika*, 32:241–254, 1967. 30

[69] Hyunmo Kang and B. Shneiderman. Visualization methods for personal photo collections: browsing and searching in the photofinder. In *Multimedia and Expo, 2000. ICME 2000. 2000 IEEE International Conference on*, 2000. 28

[70] J. Kennedy and R. Eberhart. Particle swarm optimization. In *Neural Networks, 1995. Proceedings., IEEE International Conference on*, volume 4, pages 1942 –1948 vol.4, 1995. 7

[71] James Kennedy. Swarm intelligence. In Albert Zomaya, editor, *Handbook of Nature Inspired and Innovative Computing*, pages 187–219. Springer US, 2006. 6

[72] M.L. Kherfi and D. Ziou. Image retrieval based on feature weighting and relevance feedback. In *Image Processing, 2004. ICIP '04. 2004 International Conference on*, volume 1, pages 689 – 692 Vol. 1, 2004. 6

[73] Atanas Kiryakov, Borislav Popov, Ivan Terziev, Dimitar Manov, and Damyan Ognyanoff. Semantic annotation, indexing, and retrieval. *Web Semantics: Science, Services and Agents on the World Wide Web*, 2(1):49 – 79, 2004. 28

[74] Markus Koskela, Jorma Laaksonen, and Erkki Oja. Use of image subset features in image retrieval with self-organizing maps. In Peter Enser, Yiannis Kompatsiaris, Noel E. OConnor, Alan F. Smeaton, and Arnold W. M. Smeulders, editors, *Image and Video Retrieval*, volume 3115 of *Lecture Notes in Computer Science*, pages 634–634. Springer Berlin / Heidelberg, 2004. 7

[75] Y. F. Sun H. J. Zhang M. P. Czerwinski B. Field L. Wenyin, S. T. Dumais. Semi-automatic image annotation. In *Eighth IFIP TC.13 Conference on Human Computer Interaction*, July 2001. 62

[76] G. N. Lance and W. T. Williams. A General Theory of Classificatory Sorting Strategies. *The Computer Journal*, 9(4):373–380, 1967. 30

[77] Cheng-Hung Li, Chih-Yi Chiu, Chun-Rong Huang, Chu-Song Chen, and Lee-Feng Chien. Image content clustering and summarization for photo collections. In *Multimedia and Expo, 2006 IEEE International Conference on*, pages 1033 –1036, 2006. 29

[78] Jun Li, Joo Hwee Lim, and Qi Tian. Automatic summarization for personal digital photos. In *Information, Communications and Signal Processing, 2003 and the Fourth Pacific Rim Conference on Multimedia. Proceedings of the 2003 Joint Conference of the Fourth International Conference on*, volume 3, pages 1536 – 1540 vol.3, 2003. 29

[79] Joo-Hwee Lim, Jun Li, P. Mulhem, and Qi Tan. Content-based summarization for personal image library. In *Digital Libraries, 2003. Proceedings. 2003 Joint Conference on*, page 393, May 2003. 62

[80] Joo-Hwee Lim, Qi Tian, and P. Mulhem. Home photo content modeling for personalized event-based retrieval. *Multimedia, IEEE*, 10(4):28 – 37, 2003. 29

[81] Dahua Lin, Ashish Kapoor, Gang Hua, and Simon Baker. Joint people, event, and location recognition in personal photo collections using cross-domain context. In Kostas Daniilidis, Petros Maragos, and Nikos Paragios, editors, *Computer Vision ECCV 2010*, volume 6311 of *Lecture Notes in Computer Science*, pages 243–256. Springer Berlin Heidelberg, 2010. 36

[82] Hong-Bo Liu, Yi-Yuan Tang, Jun Meng, and Ye Ji. Neural networks learning using vbest model particle swarm optimisation. In *Machine Learning and Cybernetics, 2004. Proceedings of 2004 International Conference on*, volume 5, pages 3157 – 3159 vol.5, 2004. 7

[83] A.C. Loui and A. Savakis. Automated event clustering and quality screening of consumer pictures for digital albuming. *Multimedia, IEEE Transactions on*, 5(3):390 – 402, 2003. 29

[84] A.C. Loui and A.E. Savakis. Automatic image event segmentation and quality screening for albuming applications. In *Multimedia and Expo. ICME 2000.IEEE International Conference on*, 2000. 28

[85] Alexander C. Loui and Mark D. Wood. A software system for automatic albuming of consumer pictures. In *Proceedings of the seventh ACM international conference on Multimedia (Part 2)*, MULTIMEDIA '99, pages 159–162, New York, NY, USA, 1999. ACM. 62

[86] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal Computer Vision*, 60:91–110, November 2004. 1, 64

[87] B.S. Manjunath, J.-R. Ohm, V.V. Vasudevan, and A. Yamada. Color and texture descriptors. *Circuits and Systems for Video Technology, IEEE Transactions on*, 11(6):703 –715, June 2001. 17

[88] Philippe Mulhem and Joo-Hwee Lim. Home photo retrieval: Time matters. In Erwin Bakker, Michael Lew, Thomas Huang, Nicu Sebe, and Xiang Zhou, editors, *Image and Video Retrieval*, volume 2728 of *Lecture Notes in Computer Science*, pages 321–330. Springer Berlin / Heidelberg, 2003. 28

[89] F. Murtagh. A Survey of Recent Advances in Hierarchical Clustering Algorithms. *The Computer Journal*, 26(4):354–359, 1983. 30

[90] M. Naaman, Y.J. Song, A. Paepcke, and H. Garcia-Molina. Automatic organization for digital photographs with geographic coordinates. In *Digital Libraries, 2004. Proceedings of the 2004 Joint ACM/IEEE Conference on*, pages 53 – 62, 2004. 29

[91] M. Naaman, R.B. Yeh, H. Garcia-Molina, and A. Paepcke. Leveraging context to resolve identity in photo albums. In *Digital Libraries, 2005. JCDL '05. Proceedings of the 5th ACM/IEEE-CS Joint Conference on*, pages 178 –187, 2005. 37

[92] N. O'Hare and A.F. Smeaton. Context-aware person identification in personal photo collections. *Multimedia, IEEE Transactions on*, 11(2):220 –228, 2009. 37

[93] Mayuko Okayama, Nozomi Oka, and Keisuke Kameyama. Relevance optimization in image database using feature space preference mapping and particle swarm optimization. In Masumi Ishikawa, Kenji Doya, Hiroyuki Miyamoto, and Takeshi Yamakawa, editors, *Neural Information Processing*, volume 4985 of *Lecture Notes in Computer Science*, pages 608–617. Springer Berlin Heidelberg, 2008. 7

[94] K.E. Parsopoulos and M.N. Vrahatis. Recent approaches to global optimization problems through particle swarm optimization. *Natural Computing*, 1:235–306, 2002. 10.1023/A:1016568309421. 7

[95] M. Pavan and M. Pelillo. Dominant sets and pairwise clustering. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 29(1):167 –172, 2007. 43

[96] Tomas Piatrik, Krishna Chandramouli, and Ebroul Izquierdo. Image classification using biologically inspired systems. In *Proceedings of the 2nd international conference on Mobile multimedia communications*, MobiMedia '06, pages 28:1–28:5, New York, NY, USA, 2006. ACM. 7

[97] J.C. Platt, M. Czerwinski, and B.A. Field. Phototoc: automatic clustering for browsing personal photographs. In *Information, Communications and Signal Processing, 2003 and the Fourth Pacific Rim Conference on Multimedia. Proceedings of the 2003 Joint Conference of the Fourth International Conference on*, volume 1, pages 6 – 10 Vol.1, 2003. 28

[98] John C. Platt. Autoalbum: Clustering digital photographs using probabilistic model merging. *Content-Based Access of Image and Video Libraries, IEEE Workshop on*, 0:96, 2000. 62

[99] Riccardo Poli, James Kennedy, and Tim Blackwell. Particle swarm optimization. *Swarm Intelligence*, 1:33–57, 2007. 10.1007/s11721-007-0002-0. 12

[100] P. Rabbath, M.and Sandhaus and S. Boll. Automatic creation of photo books from stories in social media. In *Proceedings of second ACM SIGMM workshop on Social media*, pages 15–20, 2010. 62

[101] J. Robinson and Y. Rahmat-Samii. Particle swarm optimization in electromagnetics. *Antennas and Propagation, IEEE Transactions on*, 52(2):397 – 407, 2004. 14

[102] J. Rocchio. *Relevance Feedback in Information Retrieval*, pages 313–323. 1971. 8

[103] Kerry Rodden and Kenneth R. Wood. How do people manage their digital photographs? In *Proceedings of the SIGCHI conference on Human factors in computing systems*, CHI '03, pages 409–416, New York, NY, USA, 2003. ACM. 28, 37

[104] Y. Rui, T. S. Hunag, and S.-F. Chang. Image Retrieval: Current Techniques, Promising Directions, and Open Issues. *Journal of Visual Communication and Image Representation*, 10(1):39–62, March 1999. 5

[105] Yong Rui, T.S. Huang, M. Ortega, and S. Mehrotra. Relevance feedback: a power tool for interactive content-based image retrieval. *Circuits and Systems for Video Technology, IEEE Transactions on*, 8(5):644–655, 1998. 6

[106] Olga Russakovsky and Li Fei-Fei. Attribute learning in large-scale datasets. In *European Conference of Computer Vision (ECCV), International Workshop on Parts and Attributes*, Crete, Greece, September 2010. 2

[107] Yuhui Shi and Russell Eberhart. Parameter selection in particle swarm optimization. In V. Porto, N. Saravanan, D. Waagen, and A. Eiben, editors, *Evolutionary Programming VII*, volume 1447 of *Lecture Notes in Computer Science*, pages 591–600. Springer Berlin / Heidelberg, 1998. 10.1007/BFb0040810. 19

[108] I. Simon, N. Snavely, and S.M. Seitz. Scene summarization for online image collections. In *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, pages 1 –8, 2007. 27

[109] P. Sinha, H. Pirsiavash, and R. Jain. Personal photo album summarization. In *Proceedings of the seventeen ACM international conference on Multimedia*, pages 1131–1132, 2009. 29, 62

[110] Alan F. Smeaton, Paul Over, and Wessel Kraaij. Evaluation campaigns and trecvid. In *Proceedings of the 8th ACM international workshop on Multimedia information retrieval*, MIR '06, pages 321–330, New York, NY, USA, 2006. ACM. 44

[111] 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 Transaction Pattern Analysis Machine Intelligence*, 22:1349–1380, December 2000. 1, 5

[112] N. K. Speer, J. M. Zacks, and J. R. Reynolds. Human Brain Activity Time-Locked to Narrative Event Boundaries. *Psychological Science*, 18(5):449–455, May 2007. 61

[113] Werner Stuetzle and Rebecca Nugent. A generalized single linkage method for estimating the cluster tree of a density. *Journal of Computational and Graphical Statistics*, 19(2):397–418, 2010. 30

[114] Bongwon Suh and Benjamin B. Bederson. Semi-automatic photo annotation strategies using event based clustering and clothing based person recognition. *Interacting with Computers*, 19(4):524 – 544, 2007. 37

[115] Yuichiro Takeuchi and Masanori Sugimoto. User-adaptive home video summarization using personal photo libraries. In *Proceedings of the 6th ACM international conference on Image and video retrieval*, CIVR '07, pages 472–479, New York, NY, USA, 2007. ACM. 29

[116] Dacheng Tao, Xiaoou Tang, Xuelong Li, and Yong Rui. Direct kernel biased discriminant analysis: a new content-based image retrieval relevance feedback algorithm. *Multimedia, IEEE Transactions on*, 8(4):716 –727, 2006. 16

[117] Q. Tian, P. Hong, and T.S. Huang. Update relevant image weights for content-based image retrieval using support vector machines. In *Multimedia and Expo, 2000. ICME 2000. 2000 IEEE International Conference on*, 2000. 7

[118] Ioan Cristian Trelea. The particle swarm optimization algorithm: convergence analysis and parameter selection. *Information Processing Letters*, 85(6):317 – 325, 2003. 19

[119] Mutlu Uysal and Fatos Yarman-Vural. Selection of the best representative feature and membership assignment for content-based fuzzy image database. In Erwin Bakker, Michael Lew, Thomas Huang, Nicu Sebe, and Xiang Zhou, editors, *Image and Video Retrieval*, volume 2728 of *Lecture Notes in Computer Science*, pages 625–630. Springer Berlin Heidelberg, 2003. 17, 32

[120] Paul Viola and Michael J. Jones. Robust real-time face detection. *International Journal of Computer Vision*, 57:137–154, 2004. 37

[121] Liwei Wang, Yan Zhang, and Jufu Feng. On the euclidean distance of images. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 27(8):1334 –1339, 2005. 9

[122] Liu Wenyin, Yanfeng Sun, and Hongjiang Zhang. Mialbum - a system for home photo managemet using the semi-automatic image annotation approach. In *Proceedings of the eighth ACM international conference on Multimedia*, MULTIMEDIA '00, pages 479–480, New York, NY, USA, 2000. ACM. 29

[123] Edward Wilson. What is sociobiology? *Society*, 15:10–14, 1978. 10.1007/BF02697770. 7

[124] R.C.F. Wong and C.H.C. Leung. Automatic semantic annotation of real-world web images. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 30(11):1933 –1944, 2008. 28

[125] Yimin Wu and Aidong Zhang. A feature re-weighting approach for relevance feedback in image retrieval. In *Image Processing. 2002. Proceedings. 2002 International Conference on*, 2002. 11, 18

[126] Rui Xu and II Wunsch, D. Survey of clustering algorithms. *Neural Networks, IEEE Transactions on*, 16(3):645 –678, May 2005. 30

[127] R.R. Yager. Intelligent control of the hierarchical agglomerative clustering process. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, 30(6):835 – 845, December 2000. 31

[128] Keiji Yanai, Nikhil V. Shirahatti, Prasad Gabbur, and Kobus Barnard. Evaluation strategies for image understanding and retrieval. In *Proceedings of the 7th ACM SIGMM international workshop on Multimedia information retrieval*, MIR '05, pages 217–226, New York, NY, USA, 2005. ACM. 1

[129] Seungji Yang, Sang-Kyun Kim, and Yong Man Ro. Semantic home photo categorization. *Circuits and Systems for Video Technology, IEEE Transactions on*, 17(3):324 –335, 2007. 29

[130] K.-H. Yap and K. Wu. Fuzzy relevance feedback in content-based image retrieval systems using radial basis function network. In *Multimedia and Expo, 2005. ICME 2005. IEEE International Conference on*, page 4 pp., 2005. 7, 29

[131] Ping Yuan, Chunlin Ji, Yangyang Zhang, and Yue Wang. Optimal multicast routing in wireless ad hoc sensor networks. In *Networking, Sensing and Control, 2004 IEEE International Conference on*, volume 1, pages 367 – 371 Vol.1, 2004. 7

[132] J. M. Zacks, T. S. Braver, M. A. Sheridan, D. I. Donaldson, A. Z. Snyder, J. M. Ollinger, R. L. Buckner, and M. E. Raichle. Human brain activity time-locked to perceptual event boundaries. *Natural Neuroscience*, 4(6):651–655, June 2001. 61

[133] Ming Zhao, Yong Teo, Siliang Liu, Tat-Seng Chua, and Ramesh Jain. Automatic person annotation of family photo album. In Hari Sundaram, Milind Naphade, John Smith, and Yong Rui, editors, *Image and Video Retrieval*, volume 4071 of *Lecture Notes in Computer Science*, pages 163–172. Springer Berlin Heidelberg, 2006. 37

[134] Yan-Tao Zheng, Ming Zhao, Yang Song, H. Adam, U. Buddemeier, A. Bissacco, F. Brucher, Tat-Seng Chua, and H. Neven. Tour the world: Building a web-scale landmark recognition engine. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 1085 –1092, 2009. 28

[135] Xiang Sean Zhou and Thomas S. Huang. Relevance feedback in image retrieval: A comprehensive review. *Multimedia Systems*, 8:536–544, 2003. 7

Publications

Journals

J1 M. Broilo, F.G.B. De Natale, "A Stochastic Approach to Image Retrieval using Relevance Feedback and Particle Swarm Optimization", in *Multimedia IEEE Transactions on*, vol. 12, issue 4, pp. 267–277, 2010.

J2 M. Broilo, N. Piotto, G. Boato, N. Conci, F.G.B. De Natale, "Object Trajectory Analysis in Video Indexing and Retrieval Applications" in *Video Search and Mining, Springer-Verlag*, pp. 3-32, ISBN 978-3-642-12899-8, vol. 287, 2010.

International Conferences and Workshops

C1 M. Broilo, and F.G.B. De Natale, "Personal Photo Album Summarization for Global and Local Annotation", in Proceeding of *SPIE Electronic Imaging*, 2011.

C2 M. Broilo, E. Zavesky, A.Basso, and F.G.B. De Natale, "Unsupervised Event Segmentation of News Content with Multimodal Cues", in Proceedings of *ACM Workshop on Automated Information Extraction in Media Production*, 2010.

C3 M. Broilo, F.G.B. De Natale, "Evolutionary Image Retrieval", in Proceedings of *IEEE International Conference on Image Processing*, 2009.

C4 M. Broilo, P. Rocca, and F.G.B. De Natale, "Content-Based Image Retrieval by a Semi-Supervised Particle Swarm Optimization", in Proceedings of *IEEE International Workshop on Multimedia Signal Processig*, 2008.

C5 M. Broilo, and F.G.B. De Natale, "Content-Based Synchronization for Multiple Photos Galleries", submitted to *IEEE International Conference on Image Processing*, 2011.

C6 M. Broilo, F.G.B. De Natale, "Unsupervised Event Segmentation of Digital Photos Galleries", submitted to *IEEE International Conference on Image Processing*, 2011.

C7 R. Mattivi, M. Broilo, and F.G.B. De Natale, "An Event-based Self-organizing Framework for Personal Photo Album Management", submitted to *ACM International Conference on Multimedia Retrieval*, 2011.

C8 M. Broilo, A.Basso, and F.G.B. De Natale, "Unsupervised Anchorpersons Differentiation in News Video", submitted to *IEEE International Workshop on Content-Based Multimedia Indexing*, 2011.