UNIVERSITÀ DEGLI STUDI DI PALERMO

DOTTORATO DI RICERCA IN INGEGNERIA INFORMATICA

DIPARTIMENTO DI INGEGNERIA INFORMATICA

IMAGE AND FACE ANALYSIS FOR PERSONAL PHOTO ORGANIZATION

Tutor
Prof. Marco La Cascia

Candidato
Ing. Marco Morana

Coordinatore
Ch.mo Prof. Salvatore Gaglio

Tesi di Dottorato di Ricerca in Ingegneria Informatica – XXII Ciclo
SSD: ING-INF/05

Image and face analysis for personal photo organization

Ph.D. Thesis

-

Marco Morana

February 15, 2011

Abstract

In recent years, digital cameras are becoming very commonplace and users need tools to manage large personal photo collections. In a typical scenario, a user acquires a certain number of pictures and then transfers this new photo sequence to his PC. Thus, before being added to the whole *personal photo collection*, it would be desirable that this new photo sequence is processed and organized. For example, users may be interested in using (i.e., browsing, saving, printing and so on) a subset of stored data according to some particular picture properties. For these reasons, automatic techniques for content-based description of personal photos are needed. Tools enabling an incremental organization of the photo album should take advantage from the particular properties of digital library. Indeed, personal photo collections show peculiar characteristics as compared to generic image collections, namely, a relatively small number of different individuals can be detected across the whole collection and, generally, it is possible to group the photos based on specific attributes. In such a scenario, the user is mainly interested in *who* is in the picture and *where* and *when* the picture was shot. Considering *Who*, *where* and *when* as the fundamental aspects of photo information, in this thesis it will be given a detailed description of novel approaches for content-based image retrieval. Novel image analysis techniques will be presented, focusing in particular on face information. Then, two novel frameworks for personal photo organization will be shown.

To my family, for their help and support.

Thank you.

You could say I lost my faith in science and progress
You could say I lost my belief in the holy church
You could say I lost my sense of direction
You could say all of this and worse but

If I ever lose my faith in you
There'd be nothing left for me to do

(Gordon Matthew Thomas Sumner)

Acknowledgements

I would like to acknowledge everyone who supported my research either with their suggestions, or simply with their presence.

I wish to express my gratitude to my advisor, Prof. Marco La Cascia, and to my "putative advisor", Prof. Giuseppe Lo Re, for believing in me and giving me all that was in their power.

Special thanks to the friends who shared with me the ups and downs of the past three years. I am especially grateful to Orazio Farruggia, Antonino Fiannaca, Roberto Gallea and Massimo La Rosa for making my work easier.

Lastly, and most importantly, I wish to thank my family and my girlfriend, simply because they love me.

Contents

List of Figures

List of Tables

Glossary

FLD Fisher's Linear Discriminant; a class specific method that searches for those vectors in the feature space that best discriminate among classes.

LBP Local Binary Pattern; a type of feature originally used for texture classification.

LDA Linear Discriminant Analysis; a methods to find a linear combination of features which characterize or separate two or more classes of objects.

PCA Principal Component Analysis; a mathematical procedure that uses an orthogonal transformation to convert a set of observations of possibly correlated variables into a set of values of uncorrelated variables called principal components.

SVMs Support Vector Machines; a set of supervised learning methods that analyze data and recognize patterns, used for classification and regression analysis.

VJFD Viola and Jones Face Detector; the state-of-the-art face detection method.

1

Introduction

1.1 Motivations and goals

With the widespread diffusion of digital cameras, the cost of taking hundreds of digital pictures and storing them on personal computer is quickly approaching zero. People are then encouraged to take more and more pictures with the consequent risk that they end up with tens of thousands of pictures stored on their PCs that, without proper organization, become useless.

Currently, the main way to search digital photo libraries is by means keywords given by the user. This modality of access to the library is definitely unsatisfactory, moreover it requires users to manually associate keywords to pictures. This process has been observed to be inadequate since users add few keywords for large sets of images and, on the other side, keywords tend to be ambiguous. Time of shooting is a much more reliable cue and it is available for free since all the digital cameras attach a timestamp to the pictures they take. However its power in terms of searching capabilities is quite limited. An ideal system for image browsing should allow for automatic organization of pictures based on the semantics of photos. Personal photo libraries show peculiar characteristics as compared to generic image collection, namely the presence of people in most of the images and a relatively small number of different individuals across the whole library that allow to achieve reliable results with automatic approaches(1)(2).

Actually, managing this kind of collections requires the analysis of many aspects. For example, a user can be interested in browsing his collection based on

contextual information such as *where* the photo has been captured or considering a specific event or date, i.e. *when* the photo has been shot. Generally, user can ask for a particular person; in this case content information such as *who* is in the photo is the main cue that should be used for querying the whole collection. Of course, all this information can be combined in order to answer to more complex user's queries.

Organizing the collection based on "who" is in the photo generally requires much work from the user that, in the worst case, has to manually annotate all the photos in the collection. A few systems were proposed to assist the user in the tedious annotation task. Some systems help the user by automatically finding the faces in each collection(3). In some works, once the faces are detected, the user is presented with a list of likely tags that can be used for identifying each person; these tags are computed based on prior knowledge computed on previously uploaded and tagged photos(4).

Recently, new application such as iPhoto (5) and Picasa(6) were distributed for photo collection management. These applications provide tools for face detection and recognition that are used to suggest likely tag for each face detected within the collection. These applications seem to apply a sort of conservative strategy: faces are not considered or presented to the user when establishing matches becomes too uncertain so that precision is kept very high at the cost of several misses.

Many previously proposed methods (7) use clustering methods to group faces, each cluster representing an identity. These methods generally require to set suitable parameters to guarantee good performance as, for instance, the number of persons in the dataset (8). In general, improvements in the performance can be obtained by jointly considering information about face and clothing in the set of detected persons.

In practice, getting reliable person descriptions is not easy and a number of challenges must be considered. For instance, people detection is noisy, namely a person can not be detected at all and/or many false positives can be detected instead. In general, state of the art face detectors are able to detect almost-frontal faces but, in many cases, this condition does not hold. Faces need to be properly normalized to be compared: they should be registered in order to

set a common reference system and the effect of different illuminations should be reduced in some way. Moreover, persons' appearance changes over time, and very often persons occlude each others making challenging the segmentation of each person body. A reliable appearance representation should be able to cope with a great number of possible persons' poses and conditions in which the photos have been taken, but this is difficult to achieve in practice.

The main motivations of this thesis is the observation that personal photo organization can be considered as a collection of problems. *Who*, *where* and *when* are the fundamental aspects of photo information and input images can be intrinsically split in three domains of interest. For each domain, it is possible to focus on specific subtasks in order to provide a global solution for the main topic of this work, that is personal photo organization. Face processing techniques can provide a solution for representing the *who* aspect. The extraction of information from the image background (e.g., color and texture) allows for understanding *where* the picture was shot. Moreover, solutions for merging extracted information are required.

1.2 Contributions

The main contributions of this dissertation are:

- **Face processing techniques for personal photo organization**
  After more than 35 years of research, face processing is considered nowadays as one of the most important application of image analysis and understanding. Even though automatic recognition of faces has reached satisfactory results on well constrained conditions, it is still a challenging problem. Considering the scenario of personal collections, the task of detecting and recognizing a face is further made difficult by the fact that, in personal photos, persons are captured "in the wild", that is under totally unconstrained conditions, i.e., occlusions, different backgrounds, illumination conditions, poses and so on. For this reason a part of this work focused on the evaluation of state of the art methods applied to personal images. Face recognition

techniques have been evaluated on personal photo collections in order to select the best set of features and parameters to be used for face description. Moreover, a novel face normalization algorithm is presented for improving face recognition performances.

- **A mean-shift based approach for personal photo organization**
  Due to the diffusion of digital cameras, users are encouraged to take more and more pictures. The side effect of this action is that thousands of pictures stored on PCs become useless. For this reason novel approaches for the automatic representation of pictures achieving a more effective organization of personal photo albums need to be studied. We propose an automatic system, where each image in the collection is represented with features related to the presence of faces in the image and features characterizing background and time information. Faces, time, and background information of each image in the collection is automatically organized using a mean-shift clustering technique. Given the particular domain of personal photo libraries, where most of the pictures contain faces of a relatively small number of different individuals, clusters tend to be semantically significant besides containing visually similar data.

- **A data association framework for people re-identification in photo sequences**
  Organizing the collection based on *who* is in the photo generally requires much work from the user that, in the worst case, has to manually annotate all the photos in the collection. The main motivation of this contribution is the observation that clustering-based approach do not consider an important constraint: a person can not be present two times in the same photo and if a face is associated to an identity, the remaining faces in the same photo must be associated to other identities. Thus we formulate a data-association problem to select the most probable associations between depictions (features of the detected persons) and identities. Moreover, it has been noticed that people appearance heavily changes across the whole

photo collection, while it seems reasonable to consider these changes negligible when photos belong to the same event. For this reason, first the collection is organized in terms of temporal events, then person re-identification is performed within each event using appearance information, i.e., face and clothing information. Once identities have been found within each event, they are associated across events taking into account only face information and corresponding identities are merged. Experiments both on a public dataset, enabling future comparison, and on private collections confirm that this approach generally outperforms clustering methods.

- **Novel techniques for image representation and clustering on mobile devices**
  Considering the wide diffusion of mobile digital image acquisition devices, the need for managing a large number of digital images is quickly increasing. In fact, the storage capacity of such devices allow users to store hundreds or even thousands, of pictures. We addressed the scenario in which an user takes pictures in different sessions and different places, that is pictures belong to different *contexts*. In this case users may also be interested in using such devices to instantly manage (i.e., browse, save, print and so on) a subset of captured pictures according to some particular picture properties. The main motivation of this contribution is to consider the mobile multimedia device as a standalone device that allows an user to instantly manage its own pictures collection. Thus, all image representation and clustering steps have been performed simulating the device constraints, that is taking into account the cost of each operation while optimizing the whole system performance.

1.3 Dissertation outline

The remainder of the dissertation is organized as follows: an overview of face processing techniques will be given in Sect. 2 describing state of the art techniques and personal contributions to face detection, normalization and recognition challenges. Sect. 3 will provide a detailed description of a novel framework

for personal photo album management using a three-domain image representation and a mean-shift clustering approach. A two level architecture for personal photo organization will be described in Sect. 4. The problem will be modeled as the search for probable associations between faces detected in subsequent photos using face and clothing descriptions. For each proposed solution, related work will be analyzed and experimental results will be discussed. Sect. 5 will give an overview of a novel approach for automatic photo album management considering the scenario of mobile devices. Conclusions will follow in Sect. 6.

1.4 Publications

Some of the work in this thesis has recently been submitted to the prestigious *Journal of Multimedia Tools and Application (MTAP)* and it is actually at second and final round of review. Other parts have already been published in important journals in the fields of computer vision, such as *The Journal of Electronic Imaging (JEI)* and *Journal of Mobile Multimedia (JMM)*, as well as in several referred conference proceedings:

- **International Journals**
  - Morana M., La Cascia M., Vella F. (2010) Automatic Image Representation and Clustering on Mobile Devices. In: Journal of Mobile Multimedia, vol. 6(2); pp. 158-169, ISSN: 1550-4646
  - Morana M., Ardizzone E., La Cascia M., Vella F. (2009). Clustering techniques for personal photo album management. In: Journal of Electronic Imaging, vol. 18(4); pp. 1-12, ISSN: 1017-9909
- **International Conferences**
  - Morana M., La Cascia M., Lo Presti L. (2010) A Data Association Algorithm for People Re-Identification in Photo Sequences. In Proceedings of the 12th IEEE International Symposium on Multimedia. ISM '10., pp.318-323, 13-15 Dec. 2010, DOI: 10.1109/ISM.2010.55

- Morana M., La Cascia M., Sorce S. (2010). Mobile interface for content based image management. In Proceedings of Complex, Intelligent and Software Intensive Systems (CISIS), 2010 International Conference on, pp. 718-723, ISBN: 978-1-4244-5917-9
- Morana M., Ardizzone E., La Cascia M., Vella F. (2010). Three-domain image representation for personal photo album management. In Multimedia Content Access: Algorithms and Systems IV, SPIE Vol. 7540. San Jose, USA, January 2010.
- Morana M., Ardizzone E., La Cascia M. (2009). Face Processing on Low-Power Devices. Proceedings of the fourth International Conference on Embedded and Multimedia Computing. EM-Com 2009, pp. 1-6
- Morana M., Ardizzone E., La Cascia, M. (2009). Probabilistic Corner Detection for Facial Feature Extraction. In Lecture Notes in Computer Science, vol. 5716; p. 461-470, ISSN: 0302-9743
- Morana M., Gallea R., La Cascia, M. (2009). A Combined Fuzzy and Probabilistic Data Descriptor for Distributed CBIR. In Lecture Notes in Artificial Intelligence (LNAI), 189-196, Vol. 5571, 2009, ISBN 978-3-642-02281-4
- Morana M., Gualdi, G., Prati, A., Cucchiara, R., Ardizzone, E., La Cascia, M., Lo Presti, L. (2008). Enabling technologies on hybrid camera networks for behavioral analysis of unattended indoor environments and their surroundings. In Proceedings of the 1st ACM workshop on Vision networks for behavior analysis. Vancouver, Canada, October, 2008, p. 1-8
- Morana M., Genco, A., Sorce, S., Ferrarotto, C., Gallea, R., Gentile, A., Impastato, S. (2008). A Java-based Wrapper for Wireless Communications. In Proceedings of the 2008 international Conference on Complex, intelligent and Software intensive Systems, pp. 769-774

2

Face Processing

After more than 35 years of research, face processing is considered nowadays as one of the most important application of image analysis and understanding. Even though automatic recognition of faces has reached satisfactory results on well constrained tasks, it is still a challenging problem.

Face processing can be considered as a collection of problems, i.e., *face detection*, *facial feature extraction*, *pose estimation*, *face validation*, *recognition*, *tracking*, *modelling* and so on, each of which can be treated separately.

Face recognition often represents the subsequent step of face detection and face normalization processes. Face detection aims to find the image position of a single face so it is usually the first step in any automated face processing system. Appearance-based approaches could then be used to compare detected faces against a database of known individuals in order to assign them an identity. Face normalization is required to support face recognition by normalizing a face for position so that the error due to face alignment is minimized.

2.1 Face Detection

As reported in (9): "given an arbitrary image, the goal of face detection is to determine whether or not there are any faces in the image and, if present, return the image location and extent of each face".

Up to the early '90s, most face detection algorithms were focused on images with single frontal face and simple backgrounds. A survey of these approaches

was written by Samal and Iyengar (10).

A number of challenges are associated with face detection due to several factors. For example, face appearance may heavily change according to the relative camera-face pose (e.g., frontal, 45 degree, profile, upside down). Moreover some facial features, such as beards and mustaches, or lighting variations may affect the appearance (i.e., shape and color) of different faces of the same individual.

In (9) face detection methods are classified into four categories:

1. **Knowledge-based methods:** human knowledge of what constitutes a typical face is encoded by defining rules that capture the relationships between facial features. Facial features are extracted from face images and a classification step is performed by comparing features against a set of coded rules. In (11) a hierarchical knowledge-based system using a multiresolution representation of the face image is proposed. The authors describe a coarse-to-fine strategy for detecting faces by means of three levels of rules. At the higher level image is searched for face candidates using simple rules, e.g. "the center part of the face has four cells with a basically uniform intensity". At Level 2, histogram equalization and edge detection is performed on face candidates. At Level 3 face candidates are finally examined to detect facial features such as eyes and mouth.

2. **Feature invariant approaches:** the goal of these methods is to find structural features that results invariant to pose, viewpoint, or lighting variations. In this category some methods (12, 13, 14) have been proposed to locate single facial features while trying to model their relationship in terms of position. Other works (15, 16) infer the presence of a face through the identification of face-like textures. Many methods have also been proposed to build a skin color model using different color spaces such as RGB (17), HSV (18), YCbCr (19). Color information has been proven to be an effective feature in many applications, however skin models are very sensitive to significant light variations.

3. **Template matching methods:** the whole face or some facial features are represented by means standard patterns. The correlations between an input

image and the stored patterns are computed for performing face detection. Advanced approaches (e.g., multiresolution (20) or deformable templates (21)) have subsequently been proposed to achieve scale and shape invariance.

4. **Appearance-based methods:** face models are learned from a set of training images which should capture the representative variability of facial appearance and then used for detection.

The framework proposed by Viola and Jones (22) is an appearance-based approach that represents the state of the art approach to face detection. Images are classified by evaluating the values of three simple *rectangular features*. *Two-rectangle*, *three-rectangle* and *four-rectangle features* are computed as the difference between the pixels within clear and shaded rectangles (Fig. 2.1).

Each feature is scaled and shifted across all possible combinations (e.g., considering a window of 24×24 pixel there are 160.000 possible features to be calculated), however the use of an image representation called *integral image* allows the features to be computed very quickly in just a few references.

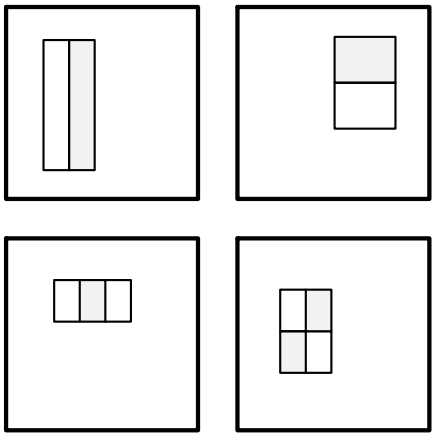


Figure 2.1: Example of two-rectangle (top row), three-rectangle and four-rectangle features used by Viola and Jones

A variant of AdaBoost (23) is then used both to select the best features from the huge feature space (e.g., 160.000 rectangle features associated with each image sub-window) and to combine them to train the classifier. Computation time is

further reduced by arranging the classifiers in a cascade, a decision tree, where a classifier at stage t is trained only on those examples which pass through all the previous stages. Thus, early stages of the cascade allow background regions of the image to be quickly discarded while spending more computation on promising regions.

Figure 2.2: Example of a set of faces detected by Viola and Jones face detector on a public face dataset.

2.2 Face Normalization

Automatic face processing for recognition (24) involves at least three different subtasks: *face detection*, *feature extraction*, *face recognition* and/or *verification*.

Face recognition has received more attention especially in the last 10 years. Recent works based on face appearance train the detection system using a large numbers of samples and perform really better than early template matching methods.

However, many face recognition systems need facial features location to normalize detected faces avoiding degradation in recognition performance.

2.2.1 State of the art

Early approaches focused on template matching to detect global features as eyes and mouth (25), while more recent models, i.e., ASM, AFM, AAM, offer more robustness and reliability working on local *feature point* position. Active Shape Model (ASM) (26) extends Active Contour Model (27) using a flexible statistical model to find feature point position in ways consistent with a training set. Active

Appearance Model (AAM) (28) combines shapes with gray-level appearance of faces.

The method proposed by Miller et al. (29) performs the co-alignment of a set of detected faces using bilinear interpolation to preserve the best quality of the face image. The alignment mechanism is based on two steps called "congealing" and "funneling". Congealing iteratively computes the distribution field, i.e., the distribution of all possible feature values at each pixel of a set of unlabeled images, then for each image computes a transformation that reduces the entropy of the set. Distributions from each iteration of congealing are saved and used during funneling to align a new image to the normalized set of images (Fig. 2.3). This method represent a useful solution to the face alignment problem, however a collection of face images are required as input of the algorithm, so this approach is unfeasible in many real scenarios.

Figure 2.3: Example of a set of photos processed by Miller et al.

Berg et. al (30) proposed a *rectification* procedure to move each face image to a canonical frame by identifying five facial feature points (corners of the left and right eyes, corners of the mouth, and the tip of the nose) and then applying an affine transformation. Geometric blur feature (31) is used as input of five SVMs and each point in the entire image is tested to identify features. This approach gives good results, however $M \times N$ points need to be tested, where $M \times N$ is the image size.

Some face detection (e.g., Viola-Jones face detector (32)) and face recognition (e.g. eigenfaces (33)) techniques have reached a certain level of maturity, however

feature extraction still represents the bottleneck of the entire process.

In the following a novel facial feature extraction method is presented to normalize Viola-Jones detected faces before performing face recognition.

2.2.2 Probabilistic Facial Feature Extraction

We started by analyzing the Viola and Jones face detector and we noticed that the use of rectangle features creates some structure on facial features distribution over the detected faces. Thus, all faces are extracted in similar way and each feature locates inside a specific region. For each feature a prior distribution is computed and used as *boost map* to filter the Harris corner detector response so that thresholding produces a finer corner detection on interest region while discarding other values. Each corner can then be tested using SVMs to detect the presence of a facial feature.

Several interest point detection techniques have been proposed and evaluated (34), however the Harris corner detector is still one of the most used due to low numerical complexity and invariance to image shift, rotation and lighting variation.

Harris approach relies on the fact that at some image points, *corners*, the image intensity changes largely in multiple directions. Corners are captured by considering the changes of intensity due to shifts in a local window.

Let I is a gray-scale image; consider taking a window W and shifting it by $(\Delta x, \Delta y)$, the auto-correlation function(35) E is defined as,

$$E(x,y) = \sum_{W} \left(I(x_i, y_i) - I(x_i + \Delta x, y_i + \Delta y)\right)^2 \tag{2.1}$$

where (x_i, y_i) are the points in the gaussian window centered on (x, y).

Approximating $I(x_i + \Delta x, y_i + \Delta y)$ by Taylor expansion,

$$I(x_i + \Delta x, y_i + \Delta y) \approx I(x_i, y_i) + \left(I_x(x_i, y_i)\, I_y(x_i, y_i)\right) \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} \tag{2.2}$$

where $I_x = \frac{\partial I}{\partial x}$ and $I_y = \frac{\partial I}{\partial y}$ denote partial differentiation in x and y, we obtain

$$E(x,y) = \begin{bmatrix} \Delta x & \Delta y \end{bmatrix} M(x,y) \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} \tag{2.3}$$

The matrix $M(x,y)$ captures the local intensity structure of the image and angle brackets denote summation over W.

Corner detection can be done by analyzing the eigenvalues of M for each point in the image, however this computation is computationally expensive. Harris suggested a measure based on the determinant and trace of M to find corners avoiding eigenvalue decomposition of M

$$\begin{aligned} R_H &= \alpha\beta - k(\alpha+\beta)^2 \\ &= Det(M) - kTr^2(M) \end{aligned} \tag{2.4}$$

where α and β are the eigenvalues of M.

A point $c(x,y)$ is then detected as corner if $R_H(x,y)$ is an 8-way local maximum.

2.2.2.1 Feature-based corner detection

Harris corner detector performs well on different types of images, however it is not sufficient for facial feature extraction.

We want to obtain a set of points C_P that contains, among others, the true facial features so that each point in C_P can be tested using SVMs to detect facial features.

The analysis of Viola and Jones face detector showed that the use of rectangle features creates some structure on facial features distribution over the detected faces. The reason for this is that each Viola-Jones face region is selected using the rectangle features (32) response to facial features (i.e. eyes, nose, mouth) position. Thus, all faces are extracted in similar way and each feature locates inside a well defined region, as shown in Fig. 2.4.

In order to reduce the computational cost and increase the rate of success of feature classification using SVMs, all points to be tested should represent true feature candidates. However Harris output is a "general-purpose" set of points,

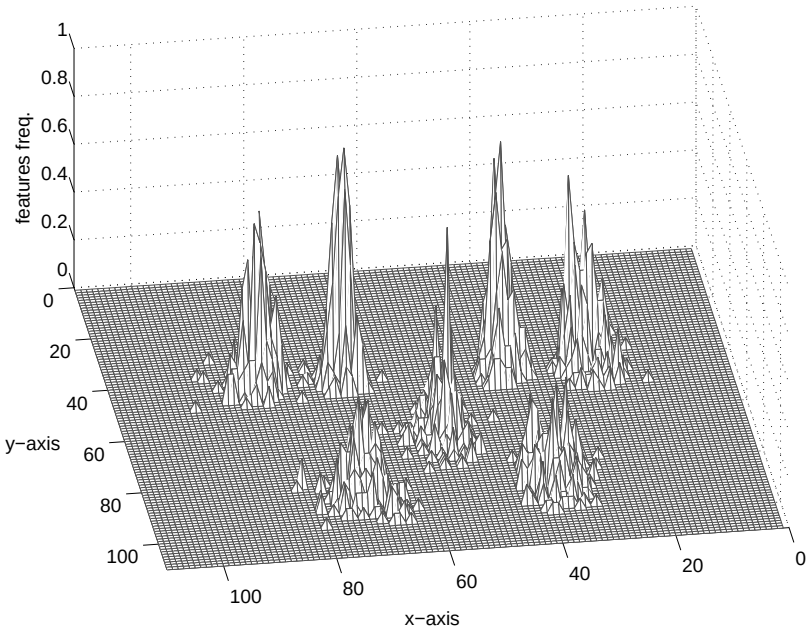


Figure 2.4: 3-D distribution of 7 facial feature points over 400 Viola-Jones size-normalized faces (110x110 pixels).

therefore many useless corners are detected and necessary ones are frequently missed. The proposed method is based on feature points distribution over size-normalized Viola-Jones detected faces. For each feature j a prior distribution B_j is used as *boost map* to filter Harris response:

1. reducing the number of corners outside the region of interested
2. increasing the number of corners inside the region of interested

Considering a training set of N face images of size $W \times L$ detected by VJFD, for each feature j the boost map B_j is given by:

$$B_j(x,y) = \frac{1}{N}\sum_{i=1}^{N} b_{ij}(x,y) \tag{2.5}$$

where $1 \leq x \leq W$, $1 \leq y \leq L$ and

$$b_{ij}(x,y) = \begin{cases} 1 & if \;\; (X_{ij} = x)\, and\, (Y_{ij} = y) \\ 0 & otherwise \end{cases} \tag{2.6}$$

Each point $B_j(x, y)$ represents the frequency with which observed features of coordinates (X_{ij}, Y_{ij}) fall in (x, y).
To reduce the dependence from training data, each B_j is approximated by a Thin Plate Spline (TPS) function (36). Harris response R_H is then filtered using the corresponding *boost map* to obtain the feature-based corner detection.

$$R_j = B_j(x, y)\, R_H \tag{2.7}$$

A feature point candidate $c_j(x, y)$ is finally detected as corner if $R_j(x, y)$ is an 8-way local maximum.

(a) **(b)** **(c)** **(d)**

Figure 2.5: Harris corner detection (a) and (c) compared with boost map-based facial feature detection (b) and (d). Detected corners are marked with x while a circle denotes the true feature position.

We refers at B_j as *boost map* since it boost the values in R_H according to the observed distribution of feature j. The values in B_j perturb the structure of R_H so that Harris thresholding produce a finer corner detection on interest region while discarding other values. TPS approximation allows to generalize training data while preserving the characteristics of the observed distributions. For this reason experimental results are very promising even using different training and test datasets.

2.2.2.2 Evaluation and Comparison

To enable detailed testing and *boost map* building we used two datasets manually annotated by Tim Cootes' staff. In order to detect feature position for face normalization, 7 feature points have been selected from the 22 facial features

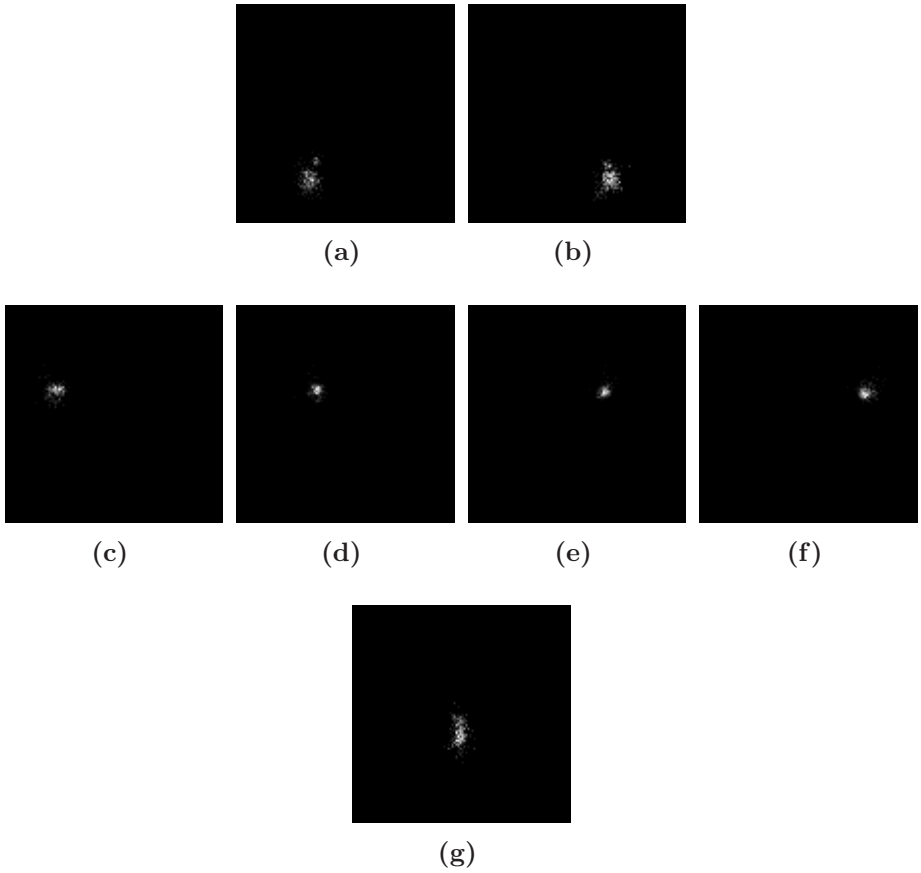


Figure 2.6: Boost maps for right (a) and left (b) corners of the mouth, external (c-f) and internal (d-e) corners of the left and right eye, tip of the nose (g).

available from the AR and BioID face database annotation. Face detection on both AR (359 labeled images) and BioID (1412 labeled images) datasets has been performed and 7 *boost maps* (Fig. 2.6) have been computed observing the position of the corners of the left and right eyes, corners of the mouth, and the tip of the nose on a 400 images subset.

To validate our method, several tests were conducted on the 1771 faces detected by VJFD. Our goal is to obtain feature candidates as close as possible to true feature point, so that for each feature we computed the corresponding *boost map*-filtered Harris response. We then compared the known position of each feature with the nearest corner detected by Harris and proposed detector. Three test sessions have been run using different couples of training and test data:

- Test A: 359 images from AR database to build the *boost map* and 1412 BioID images as test set,
- Test B: 400 images from BioID database to build the *boost map* and the remaining 1012 images as test set,
- Test C: 400 images from BioID database to build the *boost map* and 359 AR images as test set.

Results of Test A, B and C are shown in Table 2.1, Table 2.2 and Table 2.3 respectively. Each column contains the number of images in which detected corners falls at distance m from the true feature position using Harris (H_m) and proposed *boost map*-based detector (B_m). We tested for distance $m = 0$, $0 < m \leq 1$, $1 < m \leq 2$, $2 < m \leq 3$, $m > 3$, evaluating the ratio B/H for each m.

Each row contains results for the right (R) and left (L) corners of the mouth (*mR,mL*), external (E) and internal (I) corners of the left and right eyes(*eER, eIR, eLR, eIR*) and tip of the nose (*n*).

Positive ratio ($B/H > 1$) is obtained for $0 < m \leq 2$, that is the proposed approach performs better than Harris detector finding more corners in the radius of 2 pixels from the considered feature point.

Table 2.1: Test A - AR training (359 images) and BioID testing (1412 images). Results for Harris (H) and proposed approach (B) using 7 feature points.

	H_0	B_0	B/H	H_1	B_1	B/H	H_2	B_2	B/H	H_3	B_3	B/H
mR	62	168	2,71	813	804	0,99	466	421	0,90	60	18	0,30
mL	157	144	0,92	716	783	1,09	479	482	1,01	46	3	0,07
eER	29	127	4,38	77	286	3,71	413	755	1,83	468	216	0,46
eIR	47	83	1,77	109	228	2,09	493	707	1,43	382	364	0,95
eIL	18	117	6,50	63	210	3,33	366	673	1,84	482	261	0,54
eEL	27	92	3,41	130	269	2,07	641	719	1,12	390	268	0,69
n	9	71	7,89	143	259	1,81	374	711	1,90	349	230	0,66

Previous tests indicate system performance referring to the number of images in which a corner is found at distance m from the true feature position, while Fig. 2.7 shows previous values normalized to the number of corners detected by Harris (N_H) and proposed (N_B) method for each test set.

Table 2.2: Test B - BioID training (400 images) and BioID testing (1012 images). Results for Harris (H) and proposed approach (B) using 7 feature points.

	H_0	B_0	B/H	H_1	B_1	B/H	H_2	B_2	B/H	H_3	B_3	B/H
mR	88	194	2,20	618	550	0,89	274	262	0,96	28	6	0,21
mL	141	201	1,42	535	632	1,18	307	177	0,58	26	2	0,09
eER	37	129	3,50	82	199	2,43	338	500	1,48	323	150	0,47
eIR	49	103	2,11	120	183	1,52	369	481	1,30	243	198	0,82
eIL	31	115	3,67	83	172	2,08	274	509	1,86	328	190	0,58
eEL	16	134	8,38	115	177	1,54	421	553	1,31	287	139	0,48
n	2	106	45,29	94	228	2,42	300	574	1,92	174	90	0,52

Table 2.3: Test C - BioID training (400 images) and AR testing (359 images). Results for Harris (H) and proposed approach (B) using 7 feature points.

	H_0	B_0	B/H	H_1	B_1	B/H	H_2	B_2	B/H	H_3	B_3	B/H
mR	21	33	1,57	201	207	1,03	121	100	0,83	12	7	0,58
mL	20	37	1,85	206	188	0,91	122	129	1,06	7	4	0,57
eER	23	28	1,22	27	65	2,41	78	204	2,62	57	48	0,84
eIR	4	13	3,25	8	26	3,25	23	109	4,74	84	165	1,96
eIL	2	6	3,00	12	26	2,17	31	108	3,48	99	136	1,37
eEL	18	37	2,06	34	72	2,12	140	184	1,31	51	53	1,04
n	22	39	1,77	89	96	1,08	170	198	1,16	54	26	0,48

Experimental results showed that our *boost map*-based facial feature detector performs generally better than general purpose Harris corner detector. Even using different training and test data results are stable showing that each *boost map* attains an adequate level of generalization apart from used training data. Harris method detects more corners than proposed approach, moreover Harris corners are distributed over full image area while we detect corners just in features region as shown in Fig. 2.5. Thus, boost maps improve the SVMs point classification both reducing the number of points to be tested and increasing the quality/importance of those points.

Each face image is fully processed (i.e., detected, analyzed, recognized) in about 3 seconds using Matlab on a conventional 2,4 GHz Intel Pentium 4, however it does not represent a limit to the efficiency of Viola-Jones approach, since *face detection* is conceptually distinct from other face processing steps. Face detection

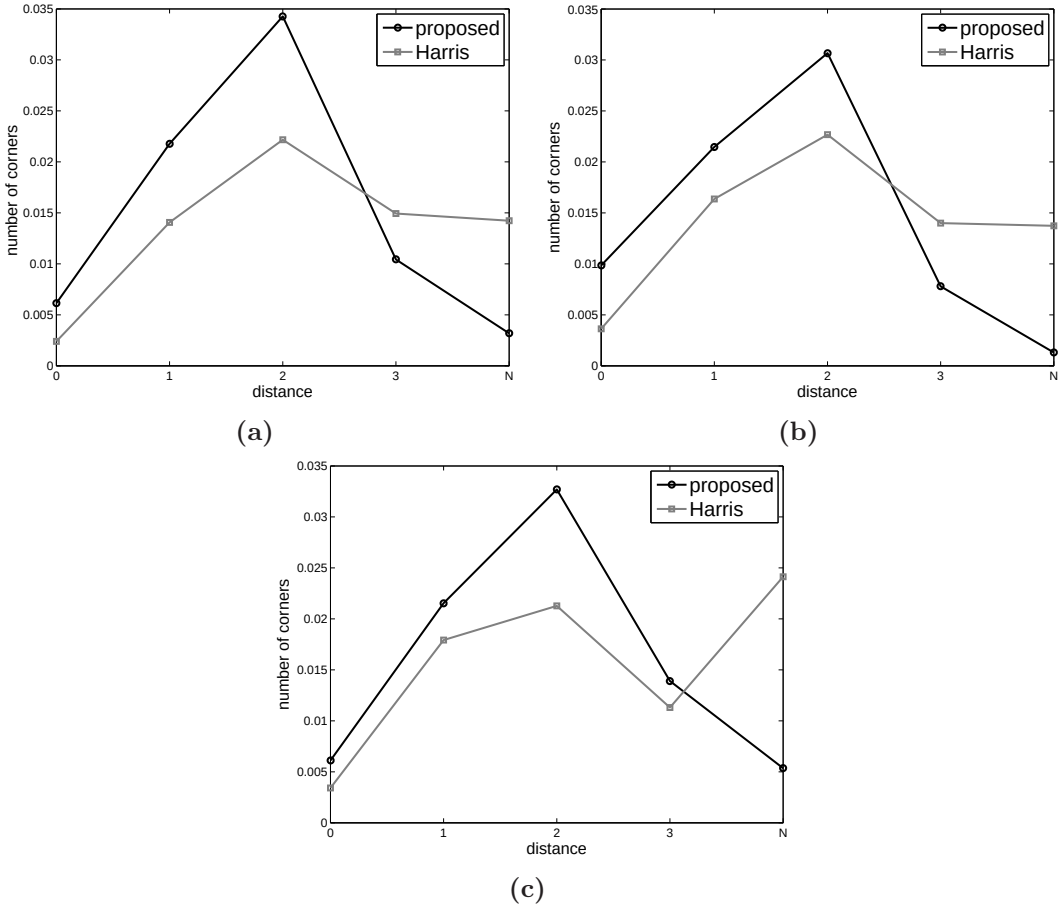


Figure 2.7: Results for Harris and proposed approach normalizing to average number of corners detected for each image.(a) Test A: $N_H = 145796$, $N_B = 130375$. (b) Test B: $N_H = 100636$, $N_B = 99655$. (c) Test C: $N_H = 32194$, $N_B = 31563$.

aims to find the image position of a single face so it is anyway the first, necessary, step in any automated face processing system. The efficiency of Viola-Jones technique is required to quickly detect a face while discarding other regions, however detected faces need to be processed again to perform subsequent tasks, e.g., face recognition, face tracking, face modelling and so on. Thus, even if the proposed technique is not suitable for a real-time system, the computational cost is not prohibitive for online image analysis.

2.3 Face Recognition

Many face recognition techniques report good classification rates over lighting variation, while performance usually drops dramatically with orientation and size changes. For this reason, once we brought a face to canonical position by means of a normalization step, it is possible to proceed to the face recognition step.

As defined by Zhao and Chellappa (37), *a general statement of the problem of machine recognition of faces can be formulated as follows: given still or video images of a scene, identify or verify one or more persons in the scene using a stored database of faces.*

Several face recognition techniques have been proposed during the past 50 years. Face recognition can be view as a multidisciplinary challenge, thus it has attracted researchers with different backgrounds: from psychology to computer science.

Many methods have been successfully applied to the task of face recognition, however nowadays face recognition is still an open issue.

In this Section a comparison of three state of the art face recognition techniques will be given.

2.3.1 Eigenfaces

The *eigenfaces* is an information theory approach, based on Principal Component Analysis (PCA), to code and decode the information content of face images.

A two-dimensional N by N image can be considered as a point in a N^2-dimensional space, called image space. A set of different images then maps to a collection of points in this space but face images will occupy a relatively low dimensional subspace. PCA provides a method to find the vectors that best represent the distribution of face images within the whole image space.

Let a training set X of M face images x_i of size $1 \times N^2$, the average face of X is defined by the vector $\bar{x}$:

$$\bar{x} = \frac{1}{M} \sum_{i=1}^{M} x_i \tag{2.8}$$

and the difference between each face image x_i and $\bar{x}$ is stored in the matrix A:

$$\begin{aligned} A &= \left[\begin{array}{cccc} (x_1 - \bar{x}) & (x_2 - \bar{x}) & \cdots & (x_M - \bar{x}) \end{array} \right] \\ &= \left[\begin{array}{cccc} \Phi_1 & \Phi_2 & \cdots & \Phi_M \end{array} \right]. \end{aligned} \tag{2.9}$$

PCA then seeks the $M-1$ orthogonal vectors which best describe the distribution of the input data, that is to find the eigenvectors of the covariance matrix C defined by:

$$\begin{aligned} C &= \frac{1}{M} \sum_{i=1}^{M} \Phi_i \Phi_i^T \\ &= AA^T \end{aligned} \tag{2.10}$$

The matrix C is $N^2 \times N^2$, thus eigenvectors and eigenvalues calculation is impractical. However there are only $M-1$ meaningful (nonzero) eigenvectors v_k and they can be computed considering the matrix $L = A^T A$ of size $M \times M$ instead of the matrix C.

The linear combination of the M input face images forms the eigenface u_k:

$$u_k = \sum_{i=1}^{M} v_{ki} \Phi_i \qquad k = 1, \ldots, M \tag{2.11}$$

In practice, while a lot of eigenfaces are required for accurate reconstruction of the image it has been observed that a smaller number of eigenfaces is sufficient for identification. Thus, the M' eigenvectors ($M' < M$) of the L matrix are chosen as those with the largest associated eigenvalues.

A new face image x is projected into *face space* producing a vector of weights $\Omega^T = \left[\begin{array}{cccc} \omega_1 & \omega_2 & \cdots & \omega_M \end{array} \right]$ that describes the contribution of each eigenface in representing the input face image, where

$$\omega_k = u_k^T (x - \bar{x}) \tag{2.12}$$

Thus, the face descriptor is given by the vector Ω^T and a face is classified as belonging to the individual k when the euclidean distance ε_k is below a threshold θ_ε, where $\varepsilon_k^2 = \|(\Omega - \Omega_k)\|^2$ and Ω_k is the vector describing the kth individual.

2.3.2 Fisherfaces

As discussed above, PCA represents a powerful way to represent the data because it ensures the data variance is maintained while eliminating unnecessary existing correlations among the original features (dimensions) in the sample vectors.

As analyzed in (38) PCA yields projection directions that maximize the total scatter across all face images, thus while the PCA projections are optimal for reconstruction from a low dimensional basis, they may be not optimal from a discriminant standpoint.

A well-known extension of the Eigenfaces method is based on the FLD, Fisher's Linear Discriminant (39). FLD is a class specific method that searches for those vectors in the underlying space that best discriminate among classes, rather than those that best describe the data. Fisherfaces (38) uses information from a labeled learning set for reducing the dimensionality of the feature space according to two measures:

1. within-class scatter matrix
   $S_W = \sum_{j=1}^{c} \sum_{i=1}^{N_j} \left(x_i^j - \mu_j\right) \left(x_i^j - \mu_j\right)^T$
2. between-class scatter matrix
   $S_B = \sum_{i=1}^{c} N_i \left(\mu_i - \mu\right) \left(\mu_i - \mu\right)^T$

where μ represents the mean of all classes, x_i^j is the ith sample of class j, μ_j is the mean of class j, c is the number of classes and N_j the number of samples in class j.

The goal is to maximize the between-class measure while minimizing the within-class measure.

This method has been proposed as an improvement of eigenfaces, however, since the exact number of classes is required it can not be applied in some real scenarios. Moreover, as discussed by Martinez et al. (40), LDA-based algorithms are not always superior to those based on PCA, especially if just a few training data are used. This is a typical situation on personal photo albums since you usually have a relatively small number of different individuals across the whole library. Thus, a PCA approach should be preferred.

2.3.3 Local Binary Patterns

Local Binary Pattern (LBP) (41) is an operator invariant to monotonic gray level and computationally efficient. This operator is a non-parametric kernel which summarizes the local spacial structure of an image. At a given pixel position, LBP is defined as an ordered set of binary comparisons of pixel intensities between the centre pixel and its eight surrounding pixels. LBP was extended to a circular neighborhood of different radius size.

The LBP operator performs the labeling of image pixels by thresholding the $M \times N$ neighborhood of each pixel with the center pixel value $p(x, y)$. Pixel labels are obtained by considering the result of the thresholding as a binary number (i.e., 1 if $p(m, n) > p(x, y)$, 0 otherwise). In (42), LBP was used for face recognition. The histogram of the extracted labels are concatenated to form a global descriptor of a generic image. In order to preserve spatial relations between facial features, the authors define *spatially enhanced histograms* for encoding both the appearance and the spatial relations of facial regions.

2.3.4 Comparison

Face recognition techniques are usually evaluated on public datasets. Many face databases have been proposed (e.g., FERET, Color FERET, PIE, Yale, AR, ORL, BioID) for testing purposes, and ground truth (i.e., face identities, location of facial features) is usually provided. This allows for easy comparison of some state of the art approaches, however such collections of faces look very different (Fig. 2.2) than those obtained from a face detection algorithm applied on a personal photo album (Fig. 2.8).

For this reason, in the following it is reported a comparison of results obtained by applying on a personal photo collection some of the most used face recognition techniques.

We considered a test set of $N = 850$ face images detected in a personal photo collection. Tests have been performed running K-means clustering with $K = 32$, being 32 the identities given in the ground truth, on three different face descriptors.

Figure 2.8: Face recognition challenge on personal photos.

Fig. 2.9 shows the three tables obtained by testing PCA, LDA and LBP descriptors of aligned faces (columns a), and both aligned and normalized faces (columns b). For each cell we report the accuracy values (eq. 2.13) obtained by repeating the clustering process three times.

$$Accuracy(\%) = \frac{\sum_i^k \sharp\,(predominant\ ID\ in\ cluster\ i)}{N} \tag{2.13}$$

	PCA (a)	PCA (b)	LDA (a)	LDA (b)	LBP (a)	LBP (b)	
ALL	35,78	**54,22**	42,89	**45,18**	66,27	61,08	18x21
	35,9	**52,41**	41,08	**46,02**	65,54	63,98	
	35,66	**55,66**	42,25	**46,14**	64,34	62,41	
1:3 = [ ]	46,75	51,45			69,88	**70,48**	10x10
	46,51	53,49			68,67	**69,28**	
	47,23	53,13			69,16	**71,33**	
21:end = [ ]	36,39	48,92					
	36,39	50,72					
	37,23	46,99					

Figure 2.9: Comparison of face recognition results using PCA, LDA, LBP after face alignment (a) and both face alignment and light normalization (b).

For each table, each row corresponds to a different set of parameters.

As reported in Moses et al. (43), much of the difference between two face images is usually due to illumination changes. This may affect PCA since the points

in the projected space will not be well clustered and images of the same individual may be incorrectly classified. Depending on this consideration, it has been suggested that by discarding the three most significant principal components, the variation due to lighting is reduced. However, it is unlikely that the first principal components correspond solely to illumination changes. This point has been verified performing the experiment reported the second row of PCA table (Fig. 2.9). Results show that, in the case of a personal photo library, discarding the first three component of PCA makes worse the accuracy of recognition since useful information for face discrimination is lost too.

On the other hand, some works proposed to discard the last X less important components (third row of PCA table in Fig. 2.9). However, as shown in the third row of PCA table, we obtained better accuracy while considering the whole PCA vector, in particular when face recognition is performed after face alignment and light normalization steps (highlighted cell).

In the second set of experiments we evaluated the performances of fisherfaces. Results, reported in LDA table, show that LDA descriptors are less discriminative than PCA-based one giving an accuracy of about 42% and 45%, after face alignment and light normalization respectively.

The last descriptor we tested is LBP. In (41), the authors suggest to use LBP in 18×21 pixel windows for images of 130×150 pixel; in our experiments, we tested the operator on 61×61 pixel and we noticed that best representations were obtained using windows of 10×10 pixel.

We discovered that LBP-based descriptors is more accurate than PCA-based one, giving an accuracy of about 70% on aligned and normalized faces, against the 55% obtained by using PCA.

3

Data Clustering Approach

A novel approach is presented here for the indexing of personal photo albums. The key point is the representation of each image by means of multiple descriptors in a form suitable for clustering. An image can be represented in several spaces allowing to capture different aspects of input data (1). In the proposed system, each image in the collection is represented with features related to the presence of faces in the image and features characterizing background and time information. A data-oriented clustering allows to generate aggregation structures driven by statistical regularities in the represented data. The proposed process of image representation is shown in Fig. 3.1.

Faces are extracted from images and are referred to a person identity; the remaining part of the image is considered as image context. Known techniques(44) are used to detect and rectify faces from the data set allowing to project all the samples in a common low dimensional *face space*. Low-level features, based on color and texture, are used to identify different contexts (*where*) by analyzing the information stored in the image background. Also for this aspect, as the typical user is interested to a limited number of different contexts, the link between low-level features and context semantic content can be reasonably established. The *when* aspect is bound to when the picture was captured and is typically referred to temporal ranges (e.g. *winter '06, summer '07*) or particular user events (e.g. *birthdays, weddings, parties*).

To automatically organize image data based on faces, background and time descriptors we use a mean-shift based approach. Organization of data does not

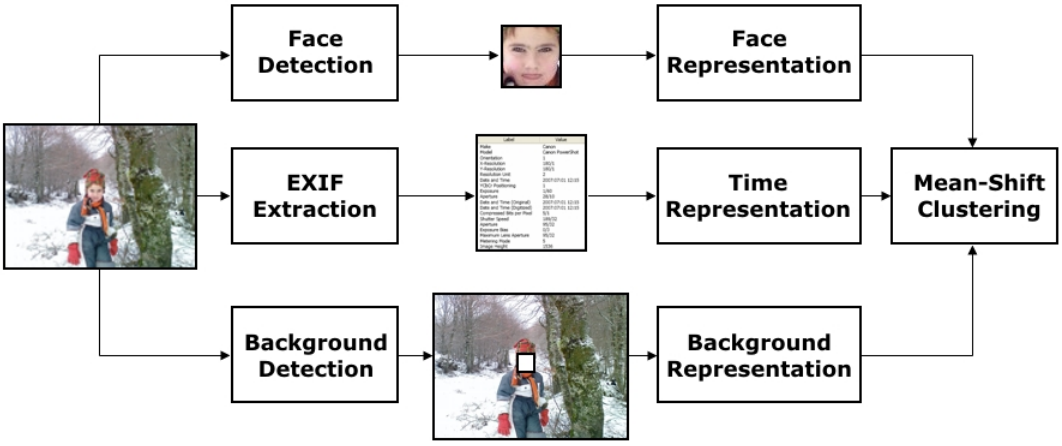


Figure 3.1: Image representation for personal photo collections.

need any human intervention as image features are automatically extracted and parameters of the clustering method are automatically determined according to a proposed entropy based figure of merit.

Note that instead of using a single common vector to represent data, the three representations allow to manage information with its own dimensionality. Furthermore the curse of dimensionality does not encourage to have larger dimension as the system may result prone to overfitting. The solution we adopted is to firstly manage data in reduced dimension spaces and then aggregate them (after clustering).

In the following sections the processing of visual information in the three chosen representation spaces is described. Faces are preprocessed to reduce the variation in appearance and are mapped in an auto emerging space employing eigenfaces. Information from the background is managed in a vector space representing low-level features. Time information is normalized according to a reference date.

3.1 State of the art

One of the first personal photo collection browser has been reported by Kang and Shneiderman(45). The goal of this system was to enable non-technical users of personal photo collection to browse and search efficiently for particular images. The authors proposed a very powerful user interface but implemented very limited Content Based Image Retrieval (CBIR) capabilities. Moreover the search was heavily based on manual annotation of the data. As in personal photos the objects of interest are often people, Zhang et al. (46) addressed the problem of automated annotation of human faces in family album. CBIR techniques and face recognition are integrated in a probabilistic framework. Based on initial training data, models of each person are built and faces in images are often recognized correctly even in presence of some occlusions. User interaction during the annotation process is also possible to reinforce the classifier. Experimental results on a family album of a few thousands photos showed the effectiveness of the approach. In a subsequent work (47) some of the authors developed a system where the user is allowed to select multiple images and assign them personal names. Then the system tries to propagate names from photograph level to face level exploiting face recognition and CBIR techniques. Abdel-Mottaleb and Chen (48) also studied the use of face arrangement in photo album browsing and retrieval. In particular they defined a similarity measure based on face arrangement that can be computed automatically and is used to define clusters of photos and finally to browse the collection. A photo management application leveraging face recognition technology has also been proposed by Girgensohn et al.(49). The authors implemented a user interface that greatly helps the users in face labeling. Other semi-automatic annotation techniques for personal photo libraries have also been proposed recently(50, 51, 52).

Other researcher address the problem of personal photo album management in an image clustering framework. For example hierarchical clustering enable the users to navigate up and down the levels to find images. Navigating the collection is also useful in query-by-example systems to find the initial image. In any case the cluster prototypes are a compact representation of classes of similar images and then can be used in browsing or searching the library. The efficacy

of the clustering approach, as well as any CBIR system, is obviously affected by the goodness of the image features used to describe the images and the similarity metrics defined over these features. As similarity metrics may not reflect semantic similarity between images, sometimes clusters are not semantically homogeneous. Many techniques have been proposed to refine the automatic clustering approach with human intervention to make cluster semantically homogeneous. Several techniques have been proposed for the clustering of images. Some authors (53) use color histogram and histogram intersection distance measure to perform hierarchical clustering. Similarly, Chen et al. (54) used global color, texture and edge histogram and the L_1 distance to define an hierarchical browsing environment. In Deng et al. (55) a self-organizing map is used to let the structure of the data emerge and then to browse the collection.

Recently, Goldberger et al. (56) proposed a generalized version of the information bottleneck principle where images are clustered to maximally preserve the mutual information between the clusters and image contents. In other cases the presence of faces in an attempt to bridge the gap between visual and semantic content is exploited. For example in Berg et al. (44) face detection is performed on captioned images and clustering is used to associate automatically extracted names to the faces. Li et al. (57) detect faces and describe clothes and nearby regions with color histogram. A similarity matrix of a photo collection is then generated according to temporal and content features and hierarchical clustering is performed based on this matrix. Song and Leung (58) aim at clustering the dataset such that each cluster contains images of a particular individual. They use face and clothing descriptors to construct an affinity matrix over the identities of individuals and perform clustering using a normalized-cut approach.

In Cui et al.(59) a semi automatic photo annotation system based on enhanced spectral clustering is proposed. They use time, global color correlogram for location/event clustering and local facial features and color correlogram from human body area for face clustering. As automatic techniques cannot guarantee that all the faces in a cluster are related to the same individuals or that an individual is not spread across several clusters, the final validation of the clustering is done by hand.

3.2 Three-domain image representation

The representation of images is composed by concatenating vectors to form a composite vector. For each face in the personal album the global representation is given by:

$$\mathbf{x} = [\mathbf{x}^f, \mathbf{x}^b, \mathbf{x}^t] \tag{3.1}$$

where $\mathbf{x}^f \in \mathbf{R}^M$ is the representation of face in the eigenspace of rectified faces, $\mathbf{x}^b \in \mathbf{R}^P$ is the background representation for the corresponding image, and $\mathbf{x}^t \in \mathbf{R}$ is the time of capture.

3.2.1 Face Representation

As already discussed, finding faces in general images is a very challenging task due to variations in pose and illumination. Berg et al.(44) analyzed hundreds of thousands of images taken from the Internet to detect faces *in the wild*. In a similar way in our approach each image to be archived in the system is searched for faces.

Detected faces are validated and rectified to a canonical pose and size. The face detector we adopted(60) is usually successful in detecting faces in a quite large range of pose, expression and illumination conditions. We used the method described in Sect. 2.2.2 to detect five feature point per face *left eye external corner*, *right eye ex tern corner*, *tip of the nose*, *mouth left corner*, and *mouth right corner* through SVM detectors. Fiducial points detectors have been trained with hundreds of positive and negative examples and Radial Basis Function kernels have been employed in SVM training. Tests on feature detection lead to about 90% of true positive with a low number of missed features.

Each face image is processed for finding the set of characteristic points. If detection is successful we estimate an affine transformation to rescale and align the face to canonical position, otherwise the face is discarded. Transformed images are cropped to images of 100×100 pixels producing a face referred to a common reference system. If the detection process produces uncertain features there is no way to use them as input for the rectification step. Our point is that only

the best faces should be processed to preserve the system robustness even if false negatives may occur.

An example of face detection, evaluation of fiducial points and rectification is shown in Fig. 3.2. A few rectified faces are reported in Fig. 3.3. Note that,

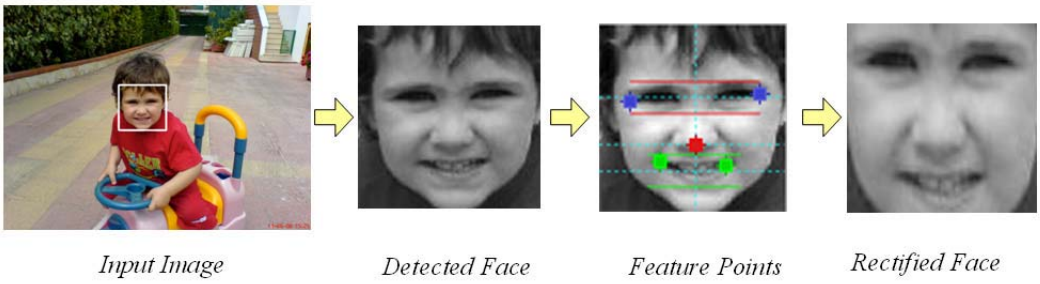


Figure 3.2: Example of detected face and corresponding rectified image

even though faces are heavily distorted, the identity of depicted people is still evident and faces are reasonably aligned to allow for appearance-based similarity search(61).

Figure 3.3: Examples of rectified faces.

As it has been shown in Sect. 2.3, several appearance-based approaches could be used for face representation, however PCA is one of the most mature and investigated method and it performs well while normalizing faces with respect to scale, translation and rotation.

Given the set of all rectified face in the Personal Album, the mean face Ψ and the eigenvectors $\mathbf{e}$ are calculated. Each image in the data set is represented by subtracting the mean image Ψ and by projecting the face vector in the eigenspace. If Φ indicates the difference between the face and the average face, the representation in the eigenspace is $w_i = \mathbf{e}_i^T \Phi$. For the data set considered in the experimental setup, the average face Ψ and the 16 eigenfaces $\mathbf{e}_i$ associated with the largest eigenvalues are shown in Fig. 3.4.

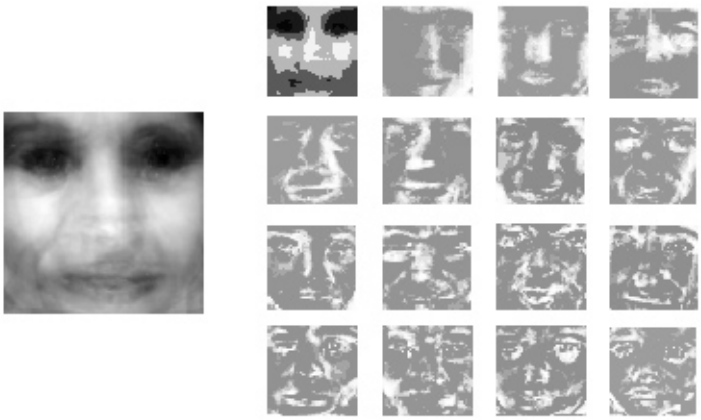


Figure 3.4: Average face and eigenfaces associated to the 16 largest eigenvalues shown in decreasing order left-to-right then top-to-bottom.

The main goal of the proposed face rectification step is to remove useless information from detected faces while maintaining facial appearance. We applied an affine transformation to preserve the differences between individuals without introducing distortion. The mean face reflects this choice since most information is stored around the features. This is the reason why even if the alignment is correct, some regions are not well defined.

The face space, as well as the average face, is learned off-line on a significant subset of the image collection and it is not updated. At any time, if most of the faces present in the image collection differ significantly from the training set, it is possible to build a new face space and recompute the projection of each detected, rectified and cropped face in the new face space.

In practice, some processed (i.e., detected and rectified) faces have been manually labeled considering a set of known, recurrent subjects. Other faces are automatically classified as belonging to a known class or to the "unknown" category. The user can also add, at any time, a new class by providing some examples.

3.2.2 Background Representation

The largest part of semantic information in personal photo is conveyed by areas where faces appear, while the remaining part of the image information is related to the context of the scene. As described above, each picture is processed with the face detector (60) by selecting areas containing faces. These areas are approximated with bounding boxes and are dealt with as explained in the previous section. The remaining part, not representing a face, conveys context for people identified in the picture. The segmentation of images into multiple areas, with different semantic value, can be extended by using additional detectors extracting further objects of interest and operating different figure/background separations. For example a detector for the entire body can be easily integrated in the system. Background information can be represented with a composition of color and texture features. In the chosen representation features are globally evaluated and a single vector for each image is produced. Color information is captured through histograms in the RGB color space. The 60-dimensional global descriptor is computed as the concatenation of the 20-bin histograms of the R, G and B channels. Texture is evaluated through Gabor filters (62) with 6 different filters, taking into account 3 orientations and 2 scales. For each filter the energy value is evaluated and represented as a 15-bin histogram. The texture feature is then composed of a total of 15x6 components and the total image feature has 150 components considering both color and texture information. The number of bins for color and texture features are empirically set.
Considering that features are distributed according to a gaussian density function, values close to the mean value are very common and - for this reason - less discriminative. To stretch values towards lower or higher values they are processed through a sigmoid.

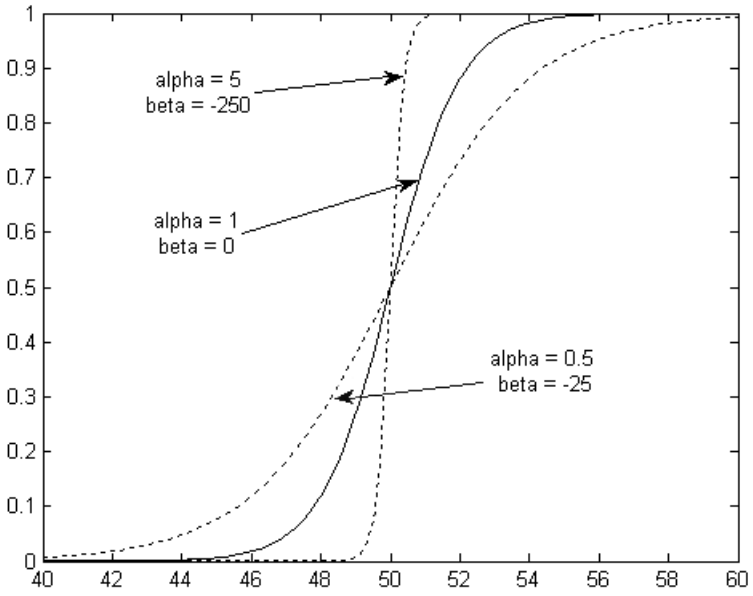


Figure 3.5: Sigmoid plotting with multiple values of α and β

Sigmoids are characterized by the parameters α and β as given in Equation (3.2)

$$f(\mathbf{x}) = \frac{1}{1 + e^{-(\alpha x - \beta)}} \tag{3.2}$$

The value of α is chosen to modulate the mapping of feature and get a softer or stronger stretching. The value of β is chosen to translate the sigmoid across the mean and is set to $\beta = -\mu\alpha$ where μ is the mean value. In Fig. 3.5 a sigmoid is shown considering a feature with $\mu = 50$.

3.2.3 Time Representation

Temporal data are available for free in image collections through the extraction of EXIF (Exchangeable image file format) data. This metadata, attached when the picture is captured, stores information about camera (manufacturer, model of camera), exposure parameters (exposure time and F number), date and hour of the image shot. Liu et al. (63) use this piece of information to classify image into

indoor or outdoor classes. Leaving out information referred to camera sensor and image exposure, here only the time of capture is considered. The value stored in the time field as date and hour of capture is converted into an integer number counting seconds from a set date (i.e., Jan 1, 1970). Images are placed in the time line and organized according to time similarity. A parameter q is introduced to choose the temporal granularity and represent time scattering of samples in a coarser or finer representation.

$$t_q = \left\lfloor \frac{t}{q} \right\rfloor \tag{3.3}$$

The larger is q the more events will be mapped in the same t_q, the smaller is q the more the event temporal description is detailed.

3.3 Image Clustering

3.3.1 The mean shift algorithm

Mean-shift is a technique applying gradient climbing to probability distribution to estimate kernel density (64). Given n data points $\mathbf{x}_i, i = 1, 2, ..., n$ in the d-dimensional space R^d, a multivariate kernel density estimator $\hat{f}(\mathbf{x})$ is calculated as:

$$\hat{f}(\mathbf{x}) = \frac{1}{nh^d} \sum_{i=1}^{n} K(\frac{\mathbf{x} - \mathbf{x_i}}{h}) \tag{3.4}$$

where h is the bandwidth and the kernel $K(.)$ is the Epanechnikov kernel defined as:

$$K(x) = \begin{cases} \frac{1}{2V_d}(d+2)(1 - \|\mathbf{x}\|^2) & \text{if } \|\mathbf{x}\|^2 < 1 \\ 0 & \text{otherwise} \end{cases} \tag{3.5}$$

with V_d representing a volume of a unit d-dimensional sphere.
Using a differentiable kernel, the estimate of the gradient density can be written as the gradient of the kernel density estimate (Equation (3.4)):

$$\hat{\nabla} f(\mathbf{x}) \equiv \nabla \hat{f}(\mathbf{x}) = \frac{1}{nh^d} \sum_{i=1}^{n} \nabla K(\frac{\mathbf{x} - \mathbf{x}_i}{h}) \tag{3.6}$$

For the Epanechnikov kernel, shown in Equation (3.5), the density gradient estimate is:

$$\hat{\nabla} f(\mathbf{x}) = \frac{n_c}{nV_d} \frac{d+2}{h^d} \left(\frac{1}{n_c} \sum_{\mathbf{x} \in S(x)} (\mathbf{x}_c - \mathbf{x}) \right) \tag{3.7}$$

where $S(\mathbf{x})$ is the hyper-sphere of radius h, having volume $h^d V_d$, centered in $\mathbf{x}$ and containing n_c data points. The quantity $M_h(\mathbf{x})$ defined as

$$M_h(\mathbf{x}) \equiv \frac{1}{n_c} \sum_{\mathbf{x} \in S(\mathbf{x})} (\mathbf{x}_c - \mathbf{x}) \tag{3.8}$$

is called Mean-Shift Vector that can be expressed, using Equation (3.7) as:

$$M_h(\mathbf{x}) = \frac{h^d}{d+2} \frac{\hat{\nabla} f(\mathbf{x})}{\hat{f}(\mathbf{x})} \tag{3.9}$$

The Mean-Shift Vector at location $\mathbf{x}$ is aligned with the local density gradient estimate and is oriented towards the direction of maximum increase in density. For each point in the vector space, the Mean Shift Vector defines a path leading from the given point to a stationary point where gradient of estimated density is equal to zero.

3.3.2 Mean Shift Clustering for Personal Album

Each picture is represented as a generic point in the feature space composed by representation in time, faces and backgrounds spaces. The Mean Shift Vector shown in Equation (3.8) describes a trajectory in the density space converging to points where the density is maximum. The set of all points converging to a local maximum is the *basin of attraction* for the found maximum density point. The procedure for the detection of modes in the data distribution is composed of two steps:

- Run mean shift to find stationary points for $\hat{f}(\mathbf{x})$
- Prune the found points retaining only the local maximum points

Clusters are refined through a merging procedure unifying adjacent clusters. Clusters are merged if:

$$\left\|\mathbf{y}_i - \mathbf{y}_j\right\| < \frac{h}{2} \tag{3.10}$$

where $\mathbf{y}_i$ and $\mathbf{y}_j$ are two local maximum points, $i \neq j$, and h is the bandwidth used to estimate the distribution density.

3.3.3 Entropy based Clustering Measure

The clustering process is driven by a set of parameters and although the number of clusters is not fixed, the best bandwidth must be selected. To evaluate the best clustering parameters, a number of evaluation indexes have been proposed, from the older Partition Coefficient and Partition Entropy(65) to the more recent ones, such as partition based on exponential separation (66). All of them tend to capture the quality of the separation proposed by clustering. Typically these methods are oriented to fuzzy clustering more than to hard (crisp) clustering and they use an estimate of the density to evaluate the clustering performance (e.g. Parzen Windows). Since we adopted a density estimation in the mean-shift procedure, to avoid a biased clustering measure, we choose to evaluate clustering as function of scattering of hand assigned identifiers in the clusters. These identifiers are related to the image content and are the names of people in a picture - for faces domain - , the identified context - for background domain - and event for event domain. These identifiers are usually referred to labels, indicating the ground truth for the given images. We define two indexes; the *Intra-Cluster Entropy* is defined as:

$$E_c = -\frac{1}{log(N_C) * log(N_L)} \sum_{i=1}^{N_L} \sum_{j=1}^{N_C} \frac{u_{ij}}{T_j} log \frac{u_{ij}}{T_j} \tag{3.11}$$

where N_C is the number of clusters, N_L is the number of *labels*, u_{ij} is the number of times the i-th label is present in the j-th cluster and T_j is the number of samples in the j-th cluster. This index gives a measure of the entropy inside clusters. If many different labels are present in a cluster, the value of ratio u_{ij}/T_j is close to the average and the value of *Intra-Cluster Entropy* is high. If a label

is concentrated in few clusters and is absent in all the others the ratio u_{ij}/T_j is close to 1 or to 0 and the entropy has a low value. This index measures the uncertainty of labels inside clusters.

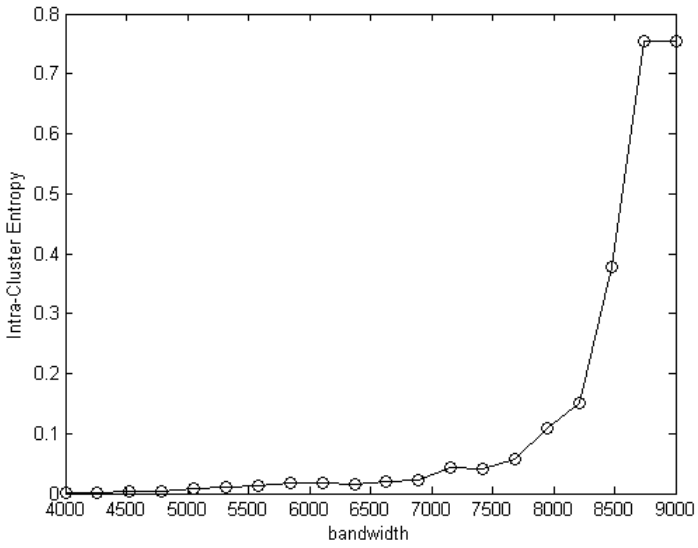


Figure 3.6: Plot of *Intra-Cluster Entropy* evaluated for clustering of faces for bandwidth values between 4000 and 8000.

In Fig. 3.6 is shown the values of *Intra-Cluster Entropy* for the clustering with mean-shift of a set of eigenfaces-faces when the value of bandwidth is between 4000 and 8000. For lower value of the bandwidth, the kernel covers a reduced volume and the number of modes is over-estimated. In this case the number of samples inside each cluster is reduced and the disorder is limited. With a larger bandwidth, the number of clusters decreases until all the samples are merged into a single cluster. In this case the *Intra-Cluster Entropy* reaches a maximum and will remain constant for higher values of the bandwidth.
The number of clusters according the value of bandwidth is shown in Fig. 3.7

The second index, the *Intra-Label Entropy* is defined as:

$$E_l = -\frac{1}{log(N_L) * log(N_C)} \sum_{i=1}^{N_L} \sum_{j=1}^{N_C} \frac{u_{ij}}{S_i} log \frac{u_{ij}}{S_i} \tag{3.12}$$

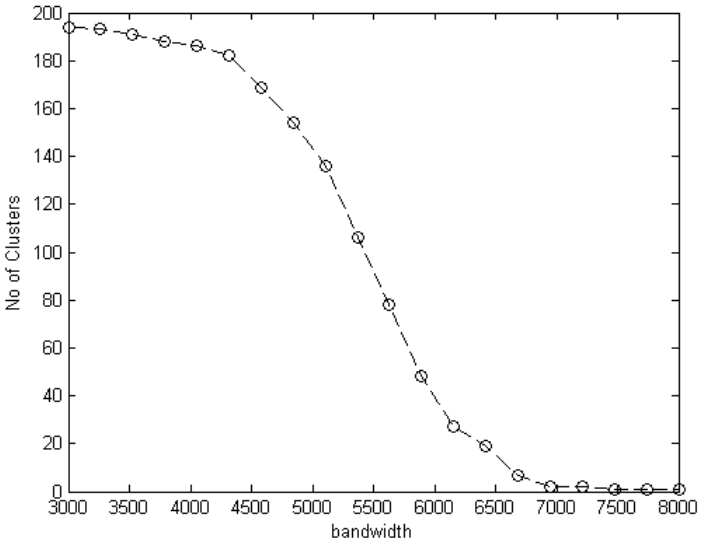


Figure 3.7: Number of clusters determined with Mean Shift procedure on eigenfaces with values of bandwidth among 4000 to 8000.

where N_C is the number of clusters, N_L is the number of labels, u_{ij} is the number of times the i-th label is present in the j-th cluster and S_i is the number of occurrence of the i-th label. This function provides a measure of the distribution of a label across clusters. If a label is always present in a cluster, or conversely always absent, the ratio u_{ij}/S_i is near 1, or near 0, and the entropy has a low value. On the other side if a label is generally present in many clusters, the more the value u_{ij}/S_i is near the average, the higher is the entropy.

For low values of bandwidth, the number of clusters is overestimated and each label (i.e. identifier) is present in many clusters. In this case, an identifier is not gathered in few clusters but distributed among many of them. At the opposite, when values of bandwidth are larger, clusters are few and many labels are found in each cluster. It is more likely, in this case, that a label is found in a single cluster and the uncertainty where a label can be found is low.
Ideally, each label describing a set of samples, should be referred to a single cluster containing all uniform samples. In real cases this distribution is very rare due to the intrinsic variability of data and errors affecting sampling. Usually a

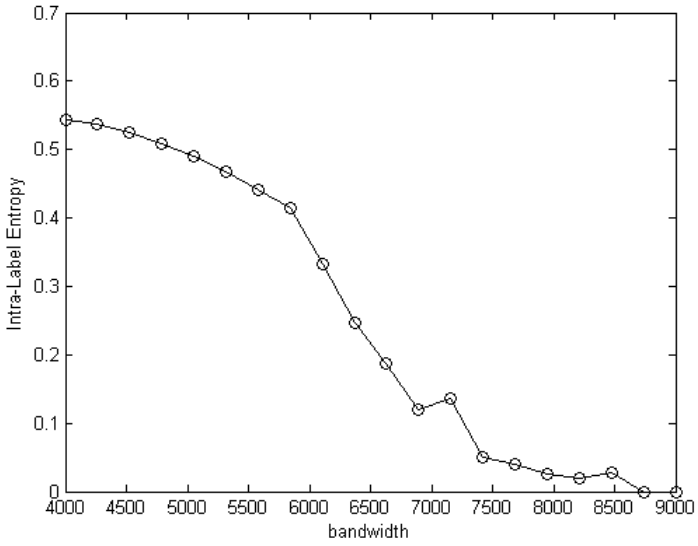


Figure 3.8: Plot of *Intra-Label Entropy* for values of bandwidth from 4000 to 8000 for the Mean Shift clustering of eigenfaces.

tradeoff in the number of cluster must be fixed.
According to the above-defined indices, in order to reduce the *Intra-Cluster Entropy* a lower bandwidth should be preferred, while in order to reduce *Intra-Label Entropy* a higher bandwidth should be chosen. To modulate this tradeoff, a measure depending on *Intra-Cluster Entropy* and *Intra-Label Entropy* is defined and is called *Global Clustering Entropy*:

$$E_G = \zeta \cdot \ E_c + (1 - \zeta) \cdot E_l \tag{3.13}$$

The value of the parameter ζ allows to modulate the weights of *Intra-Cluster Entropy* and *Intra-Label Entropy* in the final clustering. The measure of *Global Clustering Entropy* referred to Fig. 3.6 and Fig. 3.8, with needed scaling, is shown in Fig. 3.9 considering ζ equal to 0.5.

3.3.4 Mean Shift Clustering for Composite Data

The clusterization of data through the mode seeking assumes the possibility to estimate distribution density with a single kernel being the data characterized by

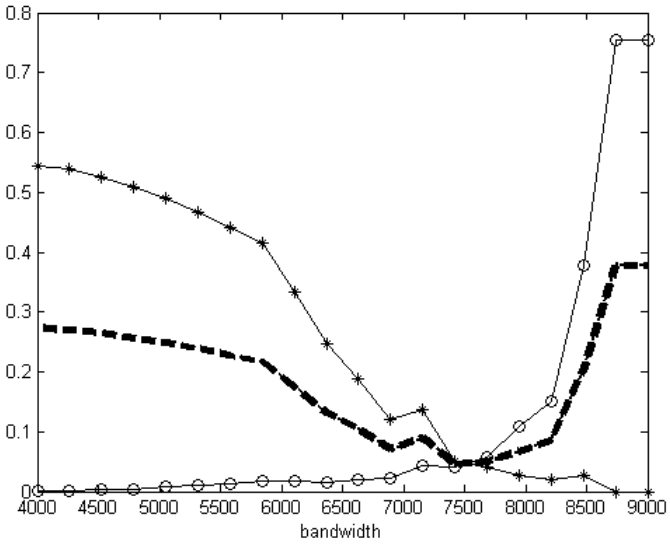


Figure 3.9: Plot of the *Global Clustering Entropy*.

the same density distribution in all the vector space. In the case considered here, the samples in personal photo album can be split into multiple representations carrying orthogonal information composed together in a single data vector.

To cluster data represented in multiple domains (Sect. 3.2), the mean-shift algorithm is applied in a similar way to what is done in image segmentation by Comaniciu et al.(64). Data are organized in homogeneous clusters with processes driven by data distribution in each domain.

Assuming that domains adopted to describe items of personal photo album allow the Euclidean norm as metric, a multivariate kernel is defined as the product of three radially symmetric kernels:

$$K_{h_f,h_b,h_t}(\mathbf{x}) = \frac{C}{h_f^M h_b^P h_t} k\left(\left\|\frac{\mathbf{x}^f}{h_f}\right\|^2\right) k\left(\left\|\frac{\mathbf{x}^b}{h_b}\right\|^2\right) k\left(\left\|\frac{\mathbf{x}^t}{h_t}\right\|^2\right) \tag{3.14}$$

where $\mathbf{x}^f$ is the data represented in the first domain, $\mathbf{x}^b$ is the data referred to the second domain, $\mathbf{x}^t$ is the representation in the third domain, h_f,h_b and h_t are the corresponding kernel bandwidths, C is the normalization constant.

For personal photo album, information is described (as shown in Sect. 3.2) as a

composition of face representation, time of capture and background representation. Faces information has a dimensionality f corresponding to the dimension of the eigenspace adopted. Background information has a dimensionality equal to b that is the sum of the dimensions of the chosen features (Sect. 3.2.2). Time information is represented with a scalar value. To cluster this composite information, a multivariate kernel is applied with mean shift procedure. Since data are intrinsically composed by three domain independent parts, a composition of three kernels is applied. The adopted kernels are three Epanechnikov kernel (Equation (3.5)) each with its own bandwidth.
Instead of empirically evaluating the performance of multiple values for the bandwidth, the *Global Clustering Entropy* introduced in Sect. 3.3.3 is used as performance measure. Driven by clustering results, the bandwidth value is automatically chosen. The process is run for the three domains, and ideally can be applied to the whole the set of orthogonal features representing input samples. The merging of clusters among multiple domains is performed as in Comaniciu et al.(64):

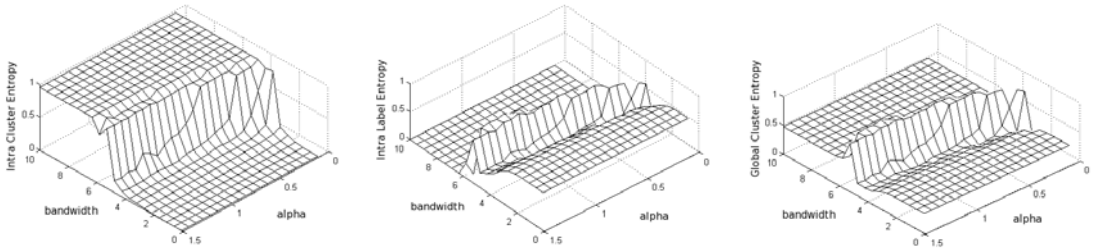
- Run the Mean Shift procedure for the chosen domains fixing a value of bandwidth for each of them. The information about the convergence points is stored.
- Identify in the joint domain the clusters by grouping the convergence points that are closer than the value of bandwidth in the corresponding domain. That is the basins of attraction of the corresponding convergence points are concatenated.
- Assign each point in the space to a cluster in the joint domain.
- Optional:Eliminate spatial regions containing less than a fixed amount of elements.

3.4 Results

To evaluate the performances of the proposed system we ran a set of experiments on a real photo collection. The digital album used is a subset of a real personal collection of 1000 images taken in the last three years. Each image has been manually labeled to store information on the presence of faces, background characteristics and time of shooting. We chose five known people so that each face is defined by an ID. The presented process for face detection and rectification brought to the extraction of 384 images of rectified faces. The experiments were aimed to evaluating the retrieval capability of the proposed system in terms of faces, background and temporal labeling and an entropy-based analysis of the clustering process was also performed to get a deepest understanding of the process itself.

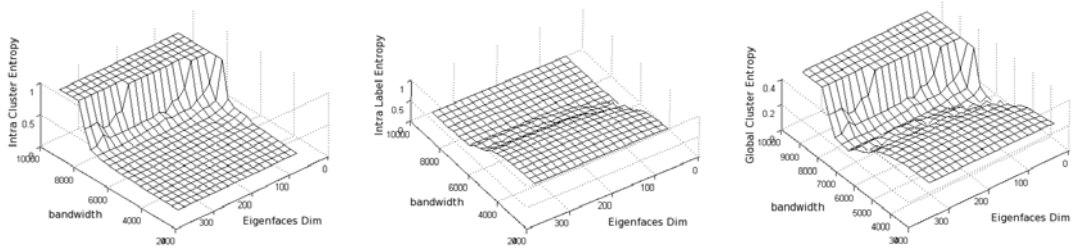
Faces	
ID	*id1, id2, id3, id4, id5, unknown*
Background	
Type	*indoor, urban, green, pool & sea, beach, snow*
Time	
Type	*kate birth, summer 06, summer 07, wedding, winter 06/07, winter 07/08, older photos*

Table 3.1: Faces and background labels



(a) Backg.d IntraCluster **(b)** Backg.d IntraLabel **(c)** Backg.d Global Entropy

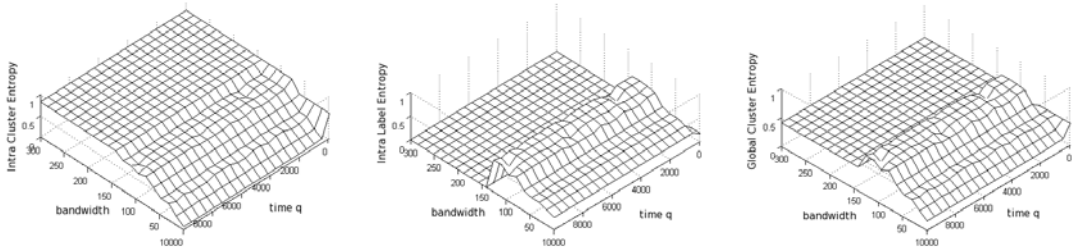
Figure 3.10: Plot of Entropy for background as function of bandwidth and alpha parameter



(a) Face IntraCluster **(b)** Face IntraLabel **(c)** Face Global Entropy

Figure 3.11: Plot of Entropy for faces as function of bandwidth and eigenspace dimension

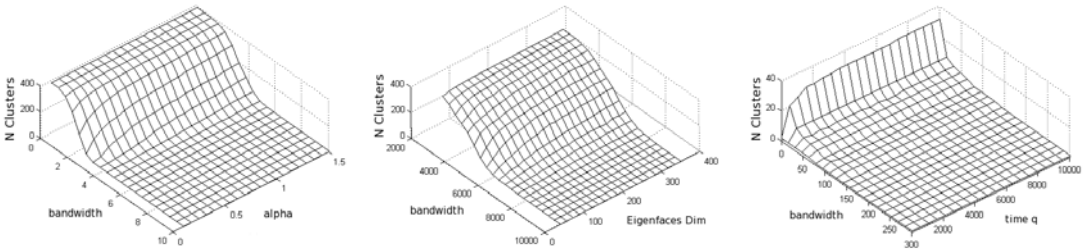
We evaluated entropy as a function of the bandwidth used in the clustering process and of parameters used in processing visual data, namely the α coefficient of the sigmoid when processing the background data, the dimension of the eigenspace for face data and the q values. Since the variation of clusters composition according to these parameters is smooth, the evaluation for a reduced set of combination of parameters produces significant results.
Values of entropy evaluation considering the Intra-Cluster Entropy, the Intra-Label Entropy and the composition of both for the clusterization in the three domains are shown in the plots in Fig. 3.10, Fig. 3.11, and Fig. 3.12.



(a) Time IntraCluster **(b)** Time IntraLabel **(c)** Time Global Entropy

Figure 3.12: Plot of Entropy for time as function of bandwidth and parameter q

The background clustering has been evaluated with a range of α parameter from 0.01 to 1 and with a bandwidth from 0 to 10. Face clustering has been evaluated with an eigenspace dimension from 0 to 250 and with a bandwidth



(a) Back.d Number of Clusters

(b) Face Number of Clusters

(c) Time Number of Clusters

Figure 3.13: Number of clusters for the different domains as function of bandwidth and of a domain dependent parameter

from 4000 to 9000. Time clustering has been evaluated with a q parameter from 10^{-9} to 10^{4} and a bandwidth from 10^{-8} to 300. Each range has been decomposed into 20 values producing a grid of 400 samples for each domain.

The plots in Fig. 3.13 show the number of clusters for the three feature spaces when the parameters are varied in the same way as Entropy diagrams Fig. 3.10, Fig. 3.11 and Fig. 3.12. The background data analysis showed that the clustering providing the best value of Global Clustering Entropy occurred with a value of α equal to 0.05 and a bandwidth of 3. From the face data analysis we observed the best results for a dimension of eigenspace equal to 25 and a bandwidth of 7421. Best parameters for time q equal to 10000 and bandwidth equal to 1e-8.

Data clustered in each single domain have been evaluated with the found parameters. Background images have been classified according to six categories (*beach*, *green*, *indoor*, *pool &sea*, *snow*, *urban*) representing six typical contexts mainly present in the collection. The results for the clustering of background are shown in the Table 3.2. All the clusters with a single element are discarded, for the remaining 10 clusters the label distribution is shown.

Faces have been clustered according to the parameters of the *Global Clustering Entropy*. Discarding all the clusters with less than two elements, the number of remaining clusters is equal to 8 and the distribution is shown in Table 3.3. The IDs from 1 to 5 are the most recurrent in image repository, all the other faces are associated to a “unknown” label.

	beach	green	indoor	pool&sea	snow	urban
Cl 1	-	31%	4%	-	-	65%
Cl 2	7%	-	-	-	86%	7%
Cl 3	-	67%	20%	-	-	13%
Cl 4	23%	-	-	32%	36%	9%
Cl 5	-	41%	47%	6%	-	6%
Cl 6	-	20%	56%	4%	-	20%
Cl 7	4%	45%	7%	-	-	44%
Cl 8	7%	-	80%	-	-	13%
Cl 9	-	30%	-	-	10%	60%
Cl 10	8%	58%	-	-	-	34%

Table 3.2: Percentage occurrence of labels in generated clusters

The time information is clustered considering the found parameters and evaluating results according to manually set temporal labels such as (*kate birth*, *summer 06*, *summer 07*, *wedding*, *winter 06/07*, *winter 07/08*, *older photos*)

The clusters for the personal album using information from multiple domains are created using the procedure described in Sect. 3.3.4. An evaluation of this clusterization is achieved calculating the Global Clusterization Entropy (Equation (3.13)) using labels given by 3-uples (*identity*, *context label*, *time label*).

	Pers 1	Pers 2	Pers 3	Pers 4	Pers 5	unknown
Cl 1	-	100%	-	-	-	-
Cl 2	10%	6%	-	-	71%	13%
Cl 3	-	-	25%	-	-	75%
Cl 4	-	-	-	-	-	100%
Cl 5	-	-	100%	-	-	-
Cl 6	-	-	29%	-	14%	57%
Cl 7	-	-	-	100%	-	-
Cl 8	-	-	25%	-	-	75%

Table 3.3: Percentage occurrence of identities in generated clusters

In Table 3.4, the values of Global Entropy, when data are clustered with the multiple kernels, are shown. The value of *alpha* for background feature filtering is set to 0.05, the dimension of the eigenspace for face representation is set to 25 and the q parameter for time data is set to 10000. Value for background clustering bandwidth varies from 1.0 to 10.0 and the bandwidth for faces varies from 4000 to 7400 and the time clusterization bandwidth is set to 10^{-8}.

	4000	4850	5700	6550	7400
1.0000	0.0012	0.0013	0.0019	0.0038	0.0038
3.2500	0.0045	0.0141	0.0653	0.0878	0.1756
5.5000	0.0044	0.0142	0.0608	0.0878	0.0878
7.7500	0.0045	0.0139	0.0659	0.0878	0.0878
10.000	0.0045	0.0147	0.0567	0.0878	0.0878

Table 3.4: Value of global entropy for clusterization with background clustering bandwidth varying from 1.0 to 10.0 and bandwidth for faces varying from 400 to 7400. Time clusterization bandwidth is set to 10^{-8}

An example of clusters for the clusterization with composite data is shown in Fig. 3.14. The most frequent 3-uples for each cluster are:

- Cluster 1: *(Winter07/08, indoor, Person 5)*
- Cluster 2: *(Winter 07/08, indoor, Person 2)*
- Cluster 3: *(Summer 06, green, Person 2)*
- Cluster 4: *(Summer 07, indoor, unknown)*
- Cluster 5: *(Winter 07/08, indoor, Person 4)*
- Cluster 6: *(Summer 07, green, Person 4)*

Figure 3.14: Image clusterization exploiting multi-domain representation

Figure 3.15: Example of Cluster with background *indoor*, taken in *Winter 07/08* and depicting *Person 2*

3.5 Discussions

In this Section a novel approach to cluster visual data driven by a clusterization measure has been presented.

In the case of personal photo collection, the most important aspects are bound to who (faces/people), where (background) and when (time). Each of them has its own dimensionality and its own statistical distribution so that it is difficult to manage these pieces of information with a single vector. An initial problem is represented by the metric used to compute distances since the low-dimensionality aspects will probably not affect the final result. Considering larger dimension may lead to find accidental regularities and system may result prone to overfitting. Thus our approach is to firstly manage data in reduced dimension spaces and then cluster and aggregate them.

The aim of the proposed work is to use a common approach to manage multiple aspects in a personal photo album. Known systems manage faces and typically allow for queries about them. Here, queries regarding people, time and background are dealt with in a homogeneous way. The proposed system has been tested on a realistic set, i.e. a personal photo album, and experimental results are very interesting.

The system results can be considered as an objective clustering of the input data. However, we understand the importance of subjective evaluation, even if it is usually expensive and in some cases unreliable. Thus, we informally analyzed the behavior of the system in different cases and an example of clustering based on different aspects has been shown to give an insight of typical results.

4

Data Association Approach

In this Chapter we present a method to automatically segment a photo sequence in groups containing the same persons. Many methods in literature accomplish to this task by adopting clustering techniques.

The main motivations of this work is the observation that clustering-based methods do not consider an important cue: a person can not be present two times in the same photo and if a face is associated to an identity, the remaining faces in the same photo must be associated to other identities. We call this property "mutual exclusivity constraint" (ME).

We model the problem as the search for probable associations between faces detected in subsequent photos using face and clothing descriptions. In particular, a two level architecture (see Fig. 4.1) has been adopted: at the first level, associations are computed within meaningful temporal windows (events); at the second level, the resulting clusters are re-processed to find associations across events.

Only some works, for example (7, 67), use the ME but as post-processing step to remove incorrect associations or just to suggest probable tags to each face. In contrast with all these approaches, we present an algorithm that takes advantage from the mutual exclusivity among persons within the same photo while inferring the associations between identities and depictions over time. The estimated associations are then used to update online the identity models. Intuitively, this approach should be more effective than those based on clustering, in particular when persons in a photo show very similar characteristics; in such cases, it is challenging to associate each depiction to the correct identity but the mutual

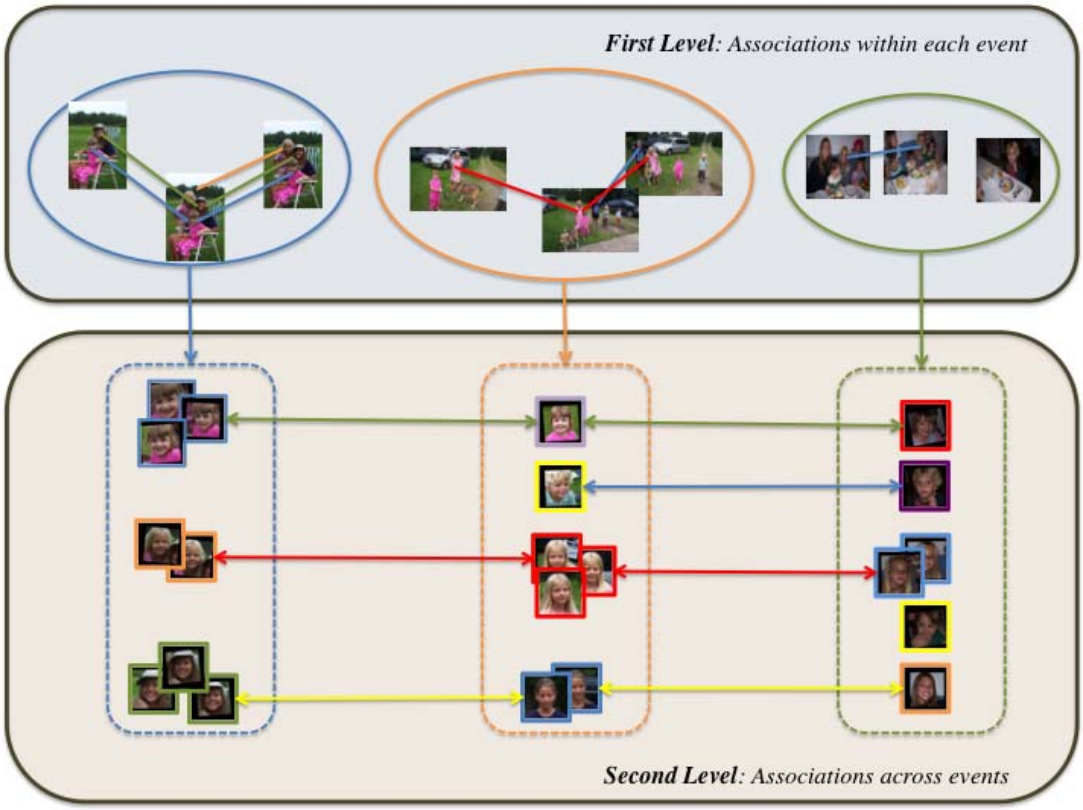


Figure 4.1: The proposed approach works on two levels: at the first level, associations between faces are discovered within each event considering both face and clothing descriptors; at the second level, identities are associated across events considering only face information.

exclusivity constraint could increase the accuracy of the people re-identification process.

In the following, first some relevant related work will be reported, then we will describe our technique providing details about the feature extraction process and the algorithms we use for detecting and representing the set of persons in a photo collection. Finally, experimental results we obtained on two private photo collections, and the comparison with the results obtained solving the problem with clustering techniques, will be shown. Moreover, we present results on a publicly available dataset (67) enabling future comparison.

4.1 State of the art

In recent years, many works focusing on methods for photo collection management were proposed. In (68), many aspects of photo segmentation are pointed out. Collections can be organized and browsed by using contextual information from EXIF data to consider where and when photos were taken. It is also possible to use an a priori known event-model adopting a hierarchical approach and providing useful insights to understand and represent the structure of the photo stream.

Many approaches were based on partitive clustering techniques for classifying images into a prefixed number of categories. However such techniques, derived from classic image retrieval studies, do not allow to obtain satisfactory results when applied to personal photo collections. In (8), K-means algorithm is applied in several, subsequent steps for identifying persons using both face and clothing information. The first step performs clustering on Principal Components Analysis (PCA) based face descriptors; then re-clustering is performed moving faces whose associated clothing descriptors largely differ from the averaged cluster-descriptor. To improve performance, scale-invariant feature transform (SIFT) are extracted from the faces and used to find within each cluster faces that largely differ from the others. Finally, a further post-processing step picks-up all the faces within a cluster that are detected in the same photo and move them in the remaining clusters. However, not enough details are provided for this latter step as, for example, how to prefer a face than another and how to guarantee that the face is not moved in a cluster where the same photo was already used. Moreover, this approach has a strong limitation: the number of face clusters is specified by the user.

In (7), for re-identifying persons across a personal library, cues such as the frontal detected face and a clothing descriptor are used. Situation clusters are defined as sets of visually similar photos taken at approximately the same time and then used to find the persons in the photo collection. Events are detected by evaluating time and visual differences between adjacent photos. However, pictures belonging to the same, real, event are often split into distinct “situation clusters”, each one containing a couple of images. Hierarchical clustering is used

to group faces considering a weighted average of the distances among face and clothing descriptors.

A similar approach is presented in (69), however a priori knowledge of the number of identities is required to cluster depictions within events by using a constrained K-means algorithm. The authors evaluated their approach on a personal collection of size comparable to the ones we used, however their dataset is probably private so that it is not possible to make a comparison of the two systems. Moreover events seem to be used to reduce the size of the collection, while it is not explained if, and how, the models obtained from each event are combined to build overall person models.

In contrast to these approaches, in (70), a technique is proposed in which users are allowed to multi-select a group of photos and assign a name/tag to one of the persons appearing in all the selected photos. The method attempts to propagate the name from photo level to face level, i.e. to infer correspondences between name and faces. However, whilst the user's effort for tagging is minimized, still the user has to manually identify the group of photos where a person appears. Moreover, in some cases the method is not able to disambiguate between persons in the photos (i.e., when some persons always appear together in the set of photos).

Faces and clothing are the main features used to group persons, also in video-content (71). Face matching is performed by considering the N-minimum pair distances between local invariant features, while clothing matching is based on 3D histogram of the dominant color.

In (72), a hierarchical clustering method is used to group the detected faces in sets each one representing a different individual. For each of these sets a clothing model is estimated. This model is then used to reprocess all the photos and recover the mis-detected faces.

In (73) clothing representation is performed by means of visual terms learned from a set of detected clothing. Face and clothing cues are then combined using time information and clustered by means of a constrained clustering algorithm.

In (67), the authors propose a technique for clothing detection by using mutual information between multiple images of the same person taken during the same event. Both face and clothing features (i.e., texture and color description) are

used to recognize people and a K-NN classifier is used to find probable label for each detected person. Within each photo, the most probable label assignment is found by maximizing its likelihood using the Hungarian algorithm (74).

Picasa (6) is an application to organize digital photos and is able to perform face recognition over photo collections. Each time the user loads a new collection, Picasa detects faces and organizes them in groups where each group should contain faces of the same person. Then it assists the user in the tagging task asking for a confirmation for every suggested tag before propagating it to the faces within the group. While the tags are added, if more than a group refers to the same person, the groups are merged.

4.2 Proposed Approach

Our method has been conceived to organize a photo stream based on *who* appears in the photo.

In order to better explain this aspect, we define a **"depiction"** as both the image of a person detected in a photo and the set of features used to represent it.

In particular, associations between persons detected in different photos are found considering only face and clothing descriptors. That is, each depiction is represented by $\mathbf{o} = \{\mathbf{o}^f, \mathbf{o}^c\}$ where $\mathbf{o}^f$ and $\mathbf{o}^c$ are its face and clothing descriptor respectively.

We also define the **"identity"** as the set of depictions referring to the same real person along the photo sequence, while the **"identity model"** is the appearance model used to represent an identity. Organizing photos based on who is depicted is a person re-identification process and the problem to solve is the association of each depiction to the correct identity.

Fig. 4.2 shows a diagram of all the components and steps that compose our method. The time ordered photo sequence is analyzed to partition it in meaningful events. Each event is composed of photos taken within a short temporal window; these photos are generally taken in similar conditions so that person's appearance and pose do not strongly change across photos within the same event. Features to represent each depiction have been extracted as described in Section

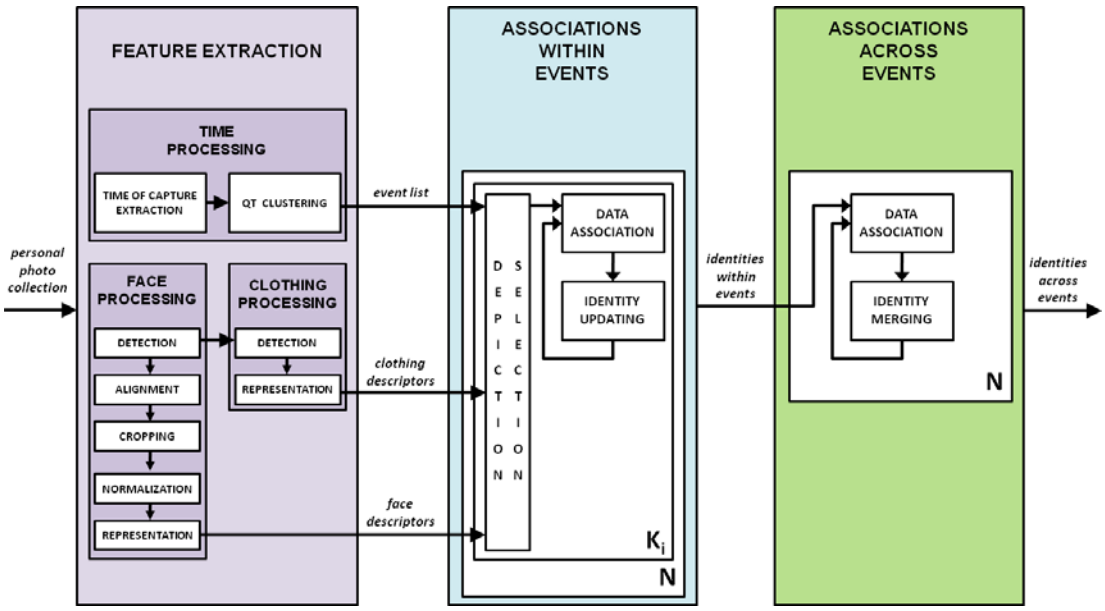


Figure 4.2: The diagram shows the main components in our system. The "feature extraction" block is devoted to process the photo sequence and extract time, face and clothing information. The event list, clothing and face descriptors are analyzed by the "associations within events" block. The "associations across events" block aims to merge such identities. The sequence of N sets of identities are sequentially analyzed and associated identities are merged. The output of the whole system is the set of merged identities.

4.2.2 and 4.2.3 where details about the face and clothing processing steps are provided.

The re-identification has been performed in two steps: first associations among depictions in photos within the same event are found. Associations are used to cluster depictions in group each one corresponding to a different identity. The discovered identities are then used to compute associations across events. Clusters corresponding to associated identities are then merged (see Fig. 4.1).

Associations are computed by formulating an assignment problem. Within each event, the data association consists in inferring the most probable associations between the depictions in a photo and the identities already discovered within the same event. Each depiction can be assigned to at most an identity and vice versa so that the mutual exclusivity constraint holds. In Fig. 4.2 depic-

tions in a photo are computed by the *depiction selection* block whose role is to select the face and clothing descriptors for each analyzed photo.

If a depiction has not been associated to any already known identities, a new identity is added. The same process is repeated for each photo within the same event by solving a sequence of $K_i - 1$ data association problems, where K_i is the number of photos in the *i-th* event. Identities are automatically initialized by the depictions in the first photo guaranteeing they refer to different persons. The output of such step is a set of identities, that is a set of depiction clusters. However, the same person can be present in more than an event so that these identities can be merged across events. To account for this, a sequence of N assignment problems are formulated in order to find associations among identities discovered in different events, where N is the number of detected events.

In this case, such associations are computed considering only face information, being the clothing information unreliable across events. Every time two identities are associated, they are merged.

The final merged identities are then presented to the user, who can tag the entire cluster instead of tagging each image. Alternatively, methods such as those presented in (4, 67) can be used for suggesting likely tags.

4.2.1 Event Detection

We note that because of adjustments in the clothing or because the photo sequence has been taken in different time, i.e. different days or months, people appearance does not change smoothly across the photo sequence. However it seems reasonable to consider these changes negligible when photos belong to the same event. In our formulation an **"event"** represents a sequence of photos taken within a meaningful short temporal window. Some features, i.e. faces descriptors, show to be more reliable across events while others, i.e. clothing information, are reliable only within the same event.

For this reason, our method segments the photo sequence in "meaningful" time intervals by considering the timestamp associated with each photo in the sequence. This temporal information, namely *when* the picture was shot, is available through the extraction of exchangeable image file format (EXIF) data. This

Figure 4.3: Events detected on a sub-sequence of 16 time ordered pictures. Under each photo we report the event number each photo belong to.

metadata is attached when the picture is captured and stores, among others, information about the date and hour of the image shot.

For the whole collection, the time of capture (ToC) value, converted into an integer number counting seconds from a reference date (i.e., Jan. 1, 1970), is used to group the pictures into clusters representing events.

Comparison between two timestamps can be simply achieved considering the absolute value of their difference. However, challenges are posed when it is necessary to segment the photo sequence in a meaningful way. In particular, we generally do not know how many meaningful temporal intervals can be extracted from the sequence so that clustering methods requiring a priori knowledge of the number of photo segments can not be applied here. Instead, it is possible to specify the maximal duration of an interval to be considered "meaningful".

Fig. 4.3 shows an example of how the photo sequence was partitioned in segments. As the image shows, within each event, persons have very similar pose and appearance.

In our approach we used a quality threshold algorithm (75) (QT clustering). This method, initially conceived for grouping gene expression patterns, does not require to specify a fixed number of clusters (namely the events) and seems the most natural way to solve our problem. Clusters are constructed in such a way that the dissimilarity within each cluster is always below a certain threshold while the cluster size is the greatest as possible. In practice, each data is considered

as a potential seed for a cluster; this last is incrementally built until no other elements can be added without increasing the cluster dissimilarity more than the prefixed threshold. Once the candidate clusters have been computed, the biggest one is chosen as cluster and the algorithm is applied again on the remaining data. In our experiments we set the quality threshold to 1 hour, being that a reasonable "meaningful" temporal interval.

Alg. 1 reports the pseudocode for the QT clustering (75). G represents the sequence of timestamps to segment while d is the threshold representing the maximum diameter of each cluster. Each timestamp i is candidate to initiate a cluster A_i. Every timestamp j is analyzed and added to A_i only if the diameter of A_i is lower than d. Among all the possible clusters, the most populated one is retained and the unclustered timestamps are reprocessed to find the next cluster. As pointed in (75), such clustering technique has some resemblance to the complete linkage hierarchical procedure. However, at a specified threshold, the QT clustering finds clusters larger on average and the method shows to be less sensitive to small perturbations in the data than hierarchical methods.

```
QT_Clust(G, d)
if (|G| ≤ 1) then
    output G; {Base case}
else
    for i ∈ G do
        flag = TRUE;
        A_i = {i}; {A_i is the cluster started by i}
        while ((flag = TRUE) and (A_i ≠ G)) do
            find j ∈ (G − A_i) such that diameter(A_i ∪ {j}) is minimum
            if (diameter(A_i ∪ {j}) > d) then
                flag = FALSE;
            else
                A_i = A_i ∪ {j}; {Add j to cluster A_i}
            end if
        end while
    end for
    output C ∈ {A_1, A_2, ..., A_|G|} with maximum cardinality
    call QT_Clust(G − C, d)
end if
```

Algorithm 1: QT Clustering

4.2.2 Face Description

To characterize *who* is in the picture, we consider features representing the appearance of the detected person (i.e., face and clothing).

As face detector we used the VJFD described in Sect. 2.1 since it is able to detect in a robust way "frontal" face view with different expressions and lighting conditions. Once the set of faces has been computed from the sequence of photos, we perform their co-alignment automatically with the Miller's approach described in Sect. 2.2.

Each aligned face has been then cropped to preserve only the region in which face information is more likely to be, while discarding noisy areas such as hair and background. To reduce the effects of different illumination conditions, we used the method presented in (76). First a gamma correction is applied, then DoG (Difference of Gaussian) filtering is used for reducing shading effects. Finally, contrast equalization is performed to rescale the image intensities.

Fig. 4.4 shows the sequence of algorithms from detection to normalization applied to the face to be described.

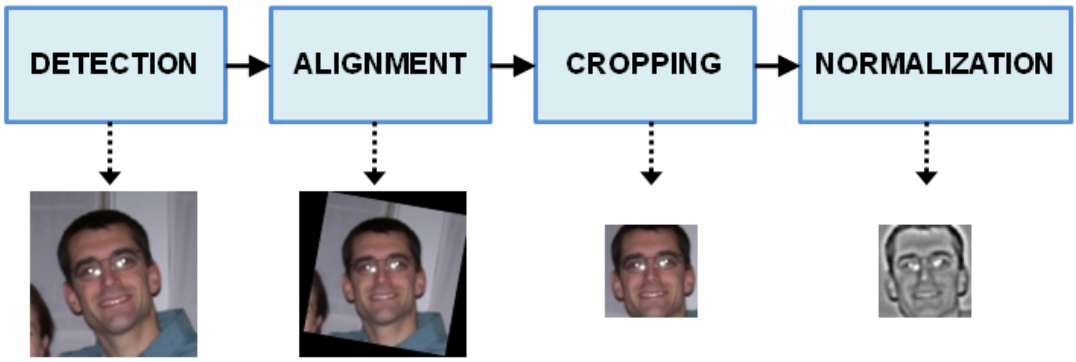


Figure 4.4: Steps for the face processing: first the face is detected, then is aligned and cropped, finally gray values are normalized to reduce illumination artifacts.

To get a face representation, we used and compared Eigenfaces (77) and Local Binary Patterns (42), described in Sect. 2.3. Other features could be used in our framework. For the sake of demonstrating our technique, we selected

and compared two features at the state of the art with different discrimination capabilities.

As in (7), we adopted a generic learning approach to estimate the face space offline by using a set of images taken from the Color FERET database (78, 79). When processing a new photo sequence, each detected face is projected into the pre-computed eigenspace and the vector of weights that describes the contribution of each eigenface in representing the input image is used as face descriptor.

We performed several tests for evaluating PCA results while changing the number of training images and the size of each face. We finally considered a training set of $M = 200$ face images of size 61×61, previously aligned and cropped. The first $M - 1$ eigenfaces have been used for describing the faces in the personal collection.

In (42), the authors suggest to use LBP in 18×21 pixel windows for images of 130×150 pixels; in our experiments, we tested the operator on 61×61 pixels and we noticed that the best representations were obtained using windows of 10×10 pixels.

4.2.3 Clothing Description

Person clothing is analyzed in terms of color information. First the region belonging to the body of each person is computed, then color information is extracted. We computed a clothing region for each detected face. We compared two approaches for segmenting clothing: we considered either the rectangular region under each detected face or the clothing area segmented via the method proposed in (67). In both cases, clothing regions suffered from some noise due to persons occluding each other and different strategies were applied to improve the clothing mask. Here, we note that the accuracy of the clothing descriptor depends on the face detector performance: we compute a clothing mask only if the face is detected; then, mis-detections and false positives can affect negatively the accuracy of the computed clothing mask. To represent clothing, we used a uniformly quantized RGB histogram of 512 bins that showed to be sufficiently robust to noise and slight illumination changes.

4.2.3.1 Finding Clothing Region

For each face, a coarse clothing region is obtained by a heuristic rule based on empirical observations: a rectangular region 40% wider and 3-times higher than the detected face is selected below the face region. In this way, the mask is computed proportionally to the face area accounting for scale changes across the photo sequence. This approach is similar to that presented in (7, 71). In (7), clothing areas are detected considering the region with fixed size at a predefined distance from the face. In (71), a heuristic similar to ours has been used.

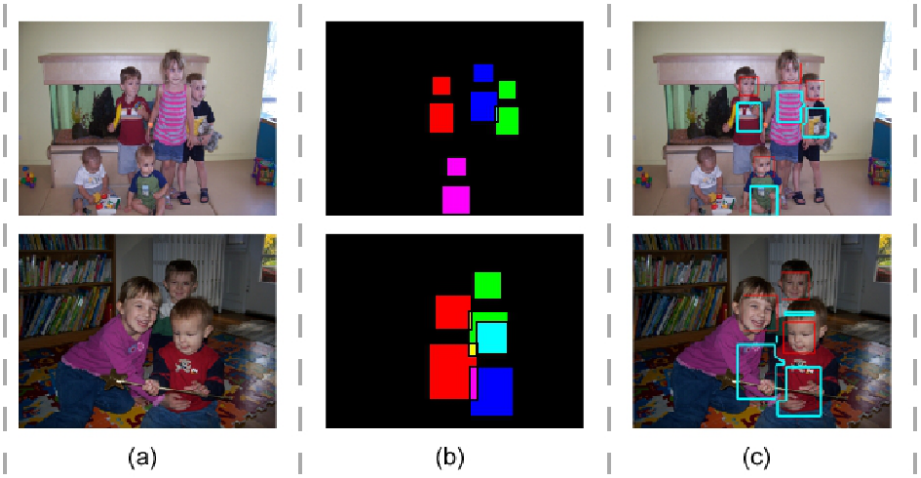


Figure 4.5: Scheme of the clothing extraction technique. (a) Input image. (b) Overlap of faces and clothing. (c) Faces (red) and clothing (cyan) areas.

Once all the clothing masks are computed for the image, each mask is refined removing overlapping areas (Fig. 4.5b). This mask refinement accounts for body occlusions of the persons detected in the photo. For each person, then, a clothing descriptor is obtained as RGB histogram of the pixels within the clothing mask. Alg. 2 outlines the main steps taken to segment the clothing region. *BBF* is the list of bounding boxes of faces detected in the current photo (*image*). Each bounding box is defined by its centre (x,y), width(w) and height(h). The heuristic rule is applied to each *BBF* to obtain the bounding box corresponding to its clothing region, *BBC*. Then the function *refine* removes overlapping areas between faces and clothing bounding boxes. RGB histogram of each refined clothing region

refined_BBC is finally computed by the function *RGB_Hist* and results are stored in the structure *clothing_descriptor*.

clothing_Representation$(image, BBF)$
{/* bounding boxes for faces (BBF) and clothing (BBC) are defined by centre (x,y), width (w) and height (h) */}
for $i = 1$ to $length(BBF)$ **do**
  $BBC.x_i = BBF.x_i + (0.2 * BBF.w_i)$;
  $BBC.y_i = BBF.y_i + (0.2 * BBF.h_i)$;
  $BBC.w_i = 1.4 * BBF.w_i$;
  $BBC.h_i = 3 * BBF.h_i$;
end for
{/* removing of overlapping areas */}
$refined\_BBC = refine(BBC, BBF)$
{/* clothing description */}
for $i = 1$ to $length(BBF)$ **do**
  $clothing = image(refined\_BBC_i)$;
  $clothing\_descriptor_i = RGB\_Hist(clothing))$;
end for
output *clothing_descriptor*

Algorithm 2: Clothing Processing

4.2.3.2 Clothing Segmentation

The method in (67) partitions a rectangular region under the face in super-pixels – found by normalized cut – and computes a color description for each of them. Starting from a seed clothing mask, the segmentation is performed using graph cut: each super-pixel is a node of the graph and the similarity to the initial mask represents the weight associated to that node. The method considers as foreground (clothing) all the super-pixels whose weights permit to minimize a certain energy function.

Fig. 4.6 shows some results we got applying such method. In contrast to (67), we did not perform co-segmentation as we do not have any knowledge about the identity of the persons in each photo and we used a simple rectangular mask as seed for the segmentation process. Whilst in many cases the method guarantees an high accuracy in segmenting clothing, we noted that skin negatively affects the results. We then improved the segmentation by eliminating the skin from the seed clothing mask. There is a huge literature on skin detection; in (80), many important aspects to consider when detecting skin are presented. In our

implementation, we used the face of each person to estimate a specific model for the skin. In practice, we transformed the image in the YCbCr color space and processed the Cb and Cr channels to estimate a Gaussian probability distribution to model the pixel values belonging to the skin by using the face pixels. Then we processed the seed region and considered as skin all those pixel whose Cb and Cr values fall in the 95% of confidence interval. These pixels were set to 0. Improving in this way the initial mask resulted in an improvement of the clothing segmentation as can be visually seen in Fig. 4.6.

Figure 4.6: Clothing segmented via Gallagher's method. The seed mask is a rectangular region centered in the image. The final row has been obtained removing skin from the seed mask.

The clothing mask was used to extract an RGB color histogram. Alg. 3 outlines the main steps taken to segment the clothing region using the method just described. The algorithm works in the same way as Alg. 2, however here a segmentation step is performed before computing RGB histograms of the clothing regions. Once the refined clothing regions *refined_BBC* have been calculated, for each detected face a skin model is estimated by the function *estimate_skin*. For each face, the skin area (*skin*) within the corresponding clothing region ($refined\_BBC_i$) is computed by means of the estimated *skin_model* and the function *estimate_skin_area*. The *seed* for the segmentation method is computed by removing skin information from the clothing region. Clothing segmen-

tation is performed by the function *gallagher_segmentation*. Finally the clothing descriptors are computed by the function *RGB_Hist* and results are stored in the structure *clothes_descriptor*.

Clothing_Representation$(image, BBF)$
{/* bounding boxes for faces (BBF) and clothing (BBC) are defined by centre (x,y), width (w) and height (h) */}
for $i = 1$ to $length(BBF)$ **do**
  $BBC.x_i = BBF.x_i + (0.2 * BBF.w_i)$;
  $BBC.y_i = BBF.y_i + (0.2 * BBF.h_i)$;
  $BBC.w_i = 1.4 * BBF.w_i$;
  $BBC.h_i = 3 * BBF.h_i$;
end for
{/* removing of overlapping areas */}
$refined\_BBC = refine(BBC, BBF)$
{/* skin color estimation from faces; seed initialization; clothing segmentation and description */}
for $i = 1$ to $length(BBF)$ **do**
  $skin\_model = estimate\_skin(image(BBF_i))$;
  $skin = estimate\_skin\_area(skin\_model, refined\_BBC_i)$;
  $seed = refined\_BBC - skin$;
  $clothing = gallagher\_segmentation(image(refined\_BBC_i), seed)$;
  $clothing\_descriptor_i = RGB\_Hist(clothing)$;
end for

Algorithm 3: Clothing Processing (Segmenting by Gallagher's Method)

4.2.3.3 Distance between descriptors

To establish matches between depictions across photos, it is necessary to compute distances among the descriptors. We tested our technique with several distance functions described in (81) and chose that one able to better separate between matches and mismatches. For the face descriptor we used a normalized version of the angle descriptor defined as:

$$d(x^f, y^f) = \begin{cases} \arccos \frac{x^f \cdot y^f}{|x^f| \cdot |y^f|} & \text{if } \arccos \frac{x^f \cdot y^f}{|x^f| \cdot |y^f|} \leq \pi, \\ 2 \cdot \pi - \arccos \frac{x^f \cdot y^f}{|x^f| \cdot |y^f|} & \text{otherwise.} \end{cases} \tag{4.1}$$

For the clothing descriptors we used Chi-square distance defined as:

$$d(x^c, y^c) = \frac{1}{2} \cdot \sum_i \left(\frac{(x_i^c - y_i^c)^2}{x_i^c + y_i^c}\right). \tag{4.2}$$

Both these distances assume values in $[0, 1]$

4.2.4 Measuring Matching between Detections and Identities

We set a probabilistic framework for establishing the probability that a depiction corresponds to a certain identity using face and clothing descriptors. We indicate an identity model as $\mathbf{i}$ and a depiction as $\mathbf{o}$. Each identity model is represented by a face model and a clothing appearance model that can be updated across time when new associations are recovered. The depiction is represented by face and clothing descriptors of the person detected.

In our framework, a match between an identity model and a depiction is indicated as $\mathbf{i} \sim \mathbf{o}$ and is represented by a binary variable assuming value 1 in case of match.

We defined the probability of a match as

$$p(\mathbf{i} \sim \mathbf{o}|\lambda_f, \lambda_c) = p(\mathbf{i^f} \sim \mathbf{o^f}|\lambda_f) \cdot p(\mathbf{i^c} \sim \mathbf{o^c}|\lambda_c) \tag{4.3}$$

where λ_f and λ_c are parameters used to represent the probability that face and clothing match.

In Eq. 4.3 we are simply considering that clothing and face are represented by two independent descriptors so that their matching can be considered independently. The main advantage in establishing this framework is that it permits to easily fuse information that ranges in different intervals and, moreover, this formulation can be easily extended to consider also other attributes about the person (i.e. texture descriptions).

Intuitively, the probability distribution should be a function whose value decrease when the distance between the two descriptors increase and can be simply computed by applying the Bayes' rule. We adopt a generative approach and compute the posterior of a match (given the distance) as:

$$p(\mathbf{i^x} \sim \mathbf{o^x}|d(\mathbf{i^x}, \mathbf{o^x}), \lambda_x) = \frac{p(d(\mathbf{i^x}, \mathbf{o^x})|\mathbf{i^x} \sim \mathbf{o^x}) \cdot p(\mathbf{i^x} \sim \mathbf{o^x})}{p(d(\mathbf{i^x}, \mathbf{o^x}))} \tag{4.4}$$

where x can be either f or c, while $d(\cdot)$ is the distance function defined in Section 4.2.3.3.

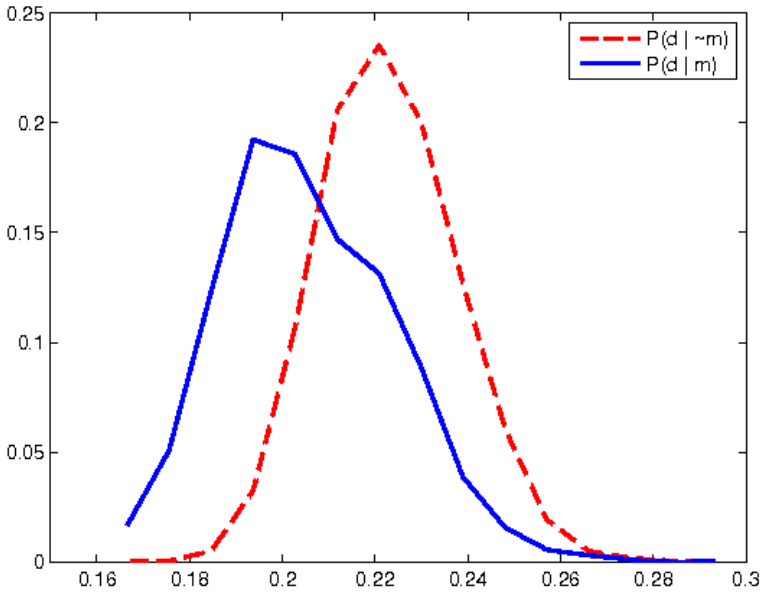


Figure 4.7: The figure shows the histograms of the distances of couple of matching and not matching face descriptors.

Fig. 4.7 shows the histograms of the distances of each couple of LBP-based face descriptors in a training set in case of match and mismatch. The two distributions are strongly overlapping making more challenging the recognition given just one sample.

In our framework, we used gamma density functions to model $p(d(\mathbf{i}^{\mathbf{x}}, \mathbf{o}^{\mathbf{x}})|\mathbf{i}^{\mathbf{x}} \sim \mathbf{o}^{\mathbf{x}})$ and $p(d(\mathbf{i}^{\mathbf{x}}, \mathbf{o}^{\mathbf{x}})|\overline{\mathbf{i}^{\mathbf{x}} \sim \mathbf{o}^{\mathbf{x}}})$ for $x \in \{c, f\}$; the denominator in Eq. 4.4 has been computed by means of the sum rule. Eq. 4.4 provides the probability of a match given the distance.

With such formulation, the probability of a match is very high when the distance is near 0 and decreases as the distributions become more and more overlapping, being more unreliable to establish a match.

The parameter λ_x is the vector representing the shape and scale parameters for the Gamma probability density function. We trained the system on sets of images corresponding to the same persons in order to estimate the parameters λ_f, λ_c by maximum likelihood estimation. We considered a small set of photos

taken in a short temporal window and detect and represent both faces and clothing. Then, using the related ground truth, we estimated the parameters for the four Gamma distributions considering the distances between couples of corresponding faces/clothing and couples of non matching faces/clothing respectively. Maximum likelihood estimation of the parameters for the Gamma distributions has no solution in closed-form but can be computed by numerical approximation. We used the method detailed in (82)[1]. We empirically set $p(\mathbf{i}^{\mathbf{x}} \sim \mathbf{o}^{\mathbf{x}}) = 0.5$, representing the a priori probability of a match.

4.3 Data Association

We formulate the data association problem as an assignment problem by means of a bipartite graph (see Fig. 4.8). A bipartite graph is a graph whose nodes can be partitioned in two disjoint subsets such that not two linked nodes belong to the same set (83). In our framework, the two subsets correspond to the set of identities and the set of depictions in the current photo respectively (compare with Fig. 4.8). Finding the set of probable associations among nodes in the two sets simply means finding the maximum matching in the graph. A matching in a bipartite graph is the set of links between nodes such that a node in a subset can be connected to at most a node in the other subset. The maximum matching is the matching providing the optimal value of a function over the weights associated to each edge in the graph (83) and can be found maximizing such function over the links in the bipartite graph. We used the Hungarian algorithm (74), sometime called also Munkres association algorithm, to maximize the function and finding the matching. The Hungarian algorithm is a combinatorial optimization algorithm which solves the assignment problem in polynomial time, while brute force would have exponential time complexity.

[1] In Matlab this computation can be easily performed by means of the *gamfit* function.

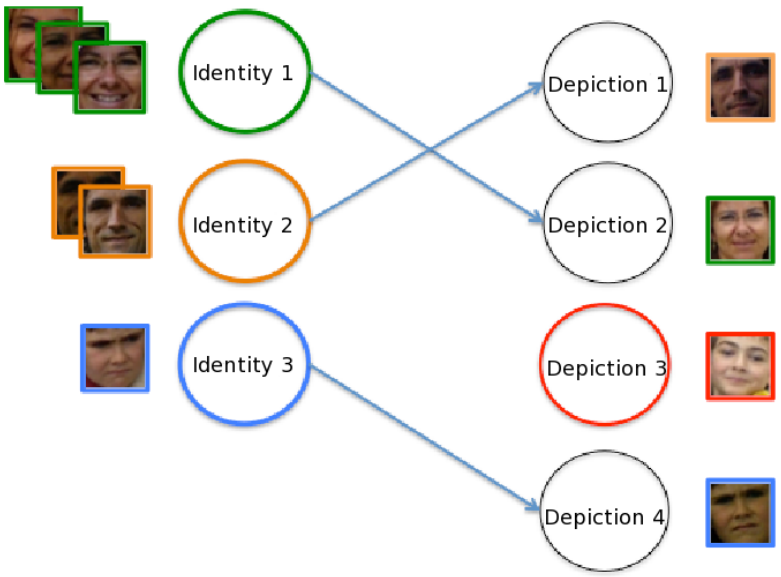


Figure 4.8: Bipartite graph to formulate the data association problem: nodes on the left represent identities while nodes on the right represent faces detected in the currently analyzed photo.

4.3.1 People Re-Identification within Events

Given a time ordered photo sequence, we initialize each identity model with the depictions detected in the first photo. Then, for each photo, we set a data association problem using a bipartite graph to represent identities and depictions, and compute the maximum matching of such graph. The set of links are then used to update the identity models.

In our domain, to solve each association problem, it makes sense to maximize the joint probability of the set of associations. To each link in the graph – corresponding to a possible association – has been assigned the logarithm of the posterior probability computed by Eq. 4.3. In our application, it is needed to consider that the two subsets in the graph have different size, that is not all the identities are jointly present in the current photo and, at each photo, new persons can be observed. Instead of taking fixed the number of identities, we gave to the system the possibility to learn new identities across time.

Inspired by (84), we added to the bipartite graph a certain number of nodes to account for new persons appearing in the collection and/or for known persons

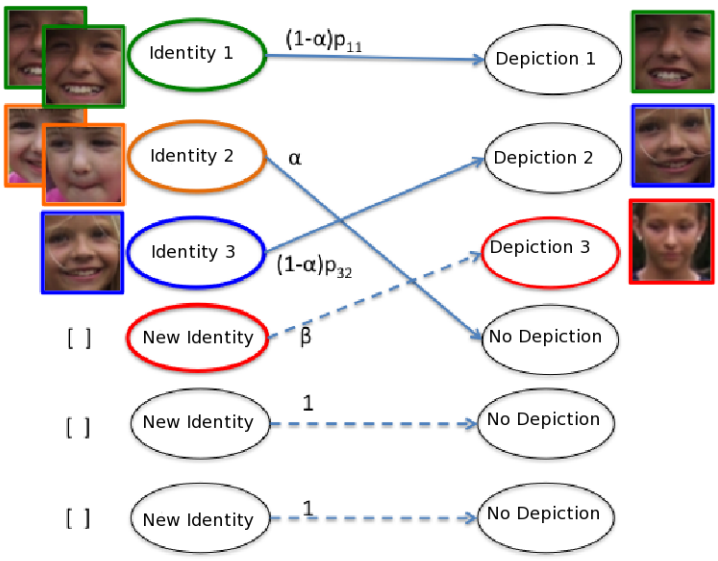


Figure 4.9: Bipartite Graph for person re-identification: the edge weights are the probabilities of each association.

that do not appear at all in the current photo. In practice, if we already have detected k identities in the sequence and in the currently analyzed photo there are w observed persons, then each subset of the bipartite graph will be composed of $k + w$ nodes because in principle none of the k already known identities could appear in the photo and all of the w depictions could initiate a new identity. The probability α represents the probability that a known identity $\mathbf{i_i}$ does not appear in the *j-th* photo so that there is a probability $(1 - \alpha)$ that he is detected. The probability β is the probability that a new identity has been discovered and initiated. We defined the probabilities α and β empirically. Learning such probabilities from a collection would cause overfitting to such collection. To get a more reliable parameter estimation, a huge set of annotated collections. However, we empirically noted that using a prefixed value for such parameters provides good performance on several datasets without lost of generality. In the experimental section we show how such parameters can affect the performance of the method.

Fig. 4.9 shows an example of matching in a bipartite graph where the edge weights represent the probability of each association.

The set M of links $\mathbf{i_i} \sim \mathbf{o_j}$ between $\mathbf{i_i}$and $\mathbf{o_j}$ in the maximum matching can be found by maximizing the function:

$$L = \sum_{\mathbf{i_i} \sim \mathbf{o_j} \in M} \log(p_A(\mathbf{i_i} \sim \mathbf{o_j})) \tag{4.5}$$

where $p_A(\mathbf{i_i} \sim \mathbf{o_j})$ is the probability of an association between the *i-th* identity and the *j-th* depiction, and is the weight of the corresponding edge in the graph. In practice,

$$p_A(\mathbf{i_i} \sim \mathbf{o_j}) = \begin{cases} (1-\alpha) \cdot p(\mathbf{i_i} \sim \mathbf{o_j} | \lambda_f, \lambda_c) \text{ if } i \leq k \text{ and } j \leq w \\ \alpha \text{ if } i \leq k \text{ and } j > w \\ \beta \text{ if } i > k \text{ and } j \leq w \\ 1 \text{ otherwise} \end{cases} \tag{4.6}$$

where $p(\mathbf{i_i} \sim \mathbf{o_j} | \lambda_f, \lambda_c)$ is computed by Eq. 4.3.

4.3.2 People Re-Identification across Events

Once identities have been found within each event, it is necessary to find correspondences among them across events. As stated in previous sections, clothing appearance does not change smoothly across time so it is an unreliable feature for re-identification in wide temporal window. To reduce the number of clusters and finding associations among identities we considered only face information. To compute the probability of a match between two identities we used a framework similar to that described in Section 4.2.4. In this context, across events, $\mathbf{o}$ does not represent anymore a depiction but, instead, the cluster associated to an identity discovered within an event.

Across events, the identity model is constituted by the whole set of faces to account for pose changes. Moreover, this limits the impact on the identity model of wrong associations that can be computed during the within-event processing step.

The Earth Mover's Distance (EMD) (85) is used to compute the similarities/associations between two identities belonging to different events. The EMD is based on a solution to the transportation problem and it is often used in computer vision, particularly to compare distributions with different dimensional

size. The transportation problem consists of finding the minimal cost that must be paid to transform one distribution into the other and is solved formulating a minimum flow problem in a graph. Given two clusters, F_1 and F_2, with m and n faces respectively, the EMD has been computed as:

$$EMD(F_1, F_2) = \sum_{i=1}^{m} \sum_{j=1}^{n} d_{i,j} \cdot f_{i,j} \tag{4.7}$$

where $d_{i,j}$ represents the distance between the *i-th* and *j-th* face descriptors respectively in clusters F_1 and F_2, while $f_{i,j}$ is the associated flow automatically computed solving the transportation problem. Node capacities were set to constant values in such a way that their sum is equal to one. We used the EMD distance to establish the probability of a match between two identities. As explained in section 4.2.4, we estimated the gamma distributions for the EMD distances in case of matching and mismatching face clusters and then used Eq. 4.4 to compute the probability of a match.

Given a time ordered event sequence the method has to merge the clusters of the matching identities across time.

4.4 Results

We performed two kind of experiments: one to test data association algorithm along the whole sequence, while the other testing the event-driven data association. In each kind of experiment, features described in sections 4.2.2 and 4.2.3 were used and compared to represent faces and clothing. Our goal is to show how face features with different discriminant capabilities can affect the overall performance of the proposed technique. Moreover, we compare two different techniques for clothing detection to evaluate if the higher computational cost needed to improve clothing segmentation would increases the re-identification accuracy.

In the following, we present extensive experiments on two private collections and on the Gallagher Dataset (67) enabling future comparison.

Table 4.1: Characteristics of the datasets used to perform our experiments

	Gallagher	Family-1	Family-2
# Photos	589	458	463
# Detected Faces	1080	648	944
# Annotated Faces	850	325	561
# Identities	32	5	5
Avg N. of faces/photo	1.89	1.84	2.04
# Detected Events	212	131	180

4.4.1 Dataset

We performed our experiments on a publicly available dataset (67) and on two private personal collections.

The public dataset is composed of 589 photos detected in different days during a period of about six months. From the ground truth, we knew that 32 different individuals where annotated. We noted that some faces were not annotated at all in the dataset. The ground truth provided along with the dataset annotates each face by means of the eye positions. We applied our own face detector on the dataset and assigned to each detected face the label in the ground truth corresponding to the person whose eyes are inside the detected face itself.

Our face detector found 1080 faces; all the non annotated faces have been treated as false positive so that 850 faces had an associated tag while the 21.3% of the faces resulted to be false positives. The last 36 photos from the Gallagher Dataset seem to be taken with a different camera and were used to learn parameters λ_c and λ_f and taken off from the testing set.

The two private collections were downloaded from Flickr. The first, we call *Family-1*, consists of 458 photos taken during a year. Our face detector found 648 faces. The second dataset, *Family-2*, consists of 463 photos were 944 faces were detected and was acquired during 11 consecutive months. The two datasets were hand labeled and a tag was assigned only to the most recurrent persons, while the remaining were treated like "Unknown".

In Table 4.1, we summarize the characteristics of each collection we used to test our technique (N. of photos, N. of detected faces, N. of annotated faces, N. of identities, average N. of detected faces in each photo, N. of events).

4.4.2 Evaluation and Comparison

To evaluate the performance of our algorithm, we associated to each identity the predominant tag we got from the ground truth and then we measured the mis-classification rate, that is the number of mis-classified faces within each identity over the total number of detected faces.

We note that many works, e.g. (7, 67), measure performance considering the accuracy rate instead of the mis-classification rate. The two measures are, of course, related (being their sum 1). However, we believe the mis-classification rate is, in this context, a more appropriate metric representing better how the user would perceive the performance of the system. Our experience suggests us that, given a face group, users tend to check if there are wrong faces within the group more than correct ones.

To measure how false positives affect the performance of the method, we computed the mis-classification rate considering the resulting identities over the collections with and without false positives.

For comparison purposes, we considered how the same problem could be solved by using a clustering method such as k-means and hierarchical clustering instead of finding the maximum matching. Basically we used methods similar to (7, 8). In (8) k-means has been used on PCA-based face descriptor; then faces were reclustered using clothing information. We did not consider the further post-processing steps for refining the results based on SIFT extraction from the faces. In (7), the first step is to use hierarchical clustering on faces and clothing descriptors using a weighted distance; these clusters are then refined by means of a measurement-level fusion method. Here, we are just adopting the clustering method and their weighted distance. Performances were measured in similar way: the predominant tag coming from the ground truth was assigned to each cluster. Then, the mis-classification rate was computed.

For evaluating the capability of the method to automatically compute the correct number of individuals in the photo collection, we present the track ratio (TR) that is the ratio between the number of detected identities and the true number of individuals in the collection. While computing the track ratio, clusters composed of a single photo were counted too. Note that in (8), the true number of individuals is assumed to be known so the track ratio is always equal to 1. On

Table 4.2: Mis-Classification Rates (%) - 0% of FPs α=0.45 β = 0.5 on the Gallagher Dataset

Method	PCA + BBox	PCA + Segm.	LBP + BBox	LBP + Segm.
K-Means	39.48	40.10	29.44	28.29
H. Clust	43.17	46.37	32.72	37.76
Ours	**29.29**	**34.51**	**23.23**	**25.43**
Track Ratio	8.5	6.8	9.97	10.09
K-Means Track-Ratio=1	48.58	48.95	31	30.75
H. Clust Track-Ratio=1	65.43	63.84	45.68	64.82

Table 4.3: Mis-Classification Rates (%) - with all the FPs α=0.45 β = 0.5 on the Gallagher Dataset

Method	PCA + BBox	PCA + Segm.	LBP + BBox	LBP + Segm.
K-Means	40.84	42.46	29.58	**28.34**
H. Clust	52.19	54.20	39.65	46.75
Ours	**36.48**	**38.92**	**24.9**	31.56
Track Ratio	9.47	7.5	13.12	11.09
K-Means Track-Ratio=1	50.86	53.43	33.97	34.06
H. Clust Track-Ratio=1	71.34	70.7	48.5	72.33

the other side our technique allows over-clustering. As mis-classification rate is computed relatively to the predominant tag in the cluster, the mis-classification rate tends to decrease in case of over-clustering (in the limit case if each cluster is made of only one face the mis-classification rate is 0%). To make a fair comparison, we also computed the mis-classification rate for the other two methods using a number of clusters equals to the number of identities we automatically found with our technique. In this way the mis-classification rate is measured on equal terms of track ratio.

4.4.3 Re-Idendification across the sequence

We first tested our technique without considering the event segmentation step and tried to compute associations across the whole photo sequence. We tested how the method works considering both clothing and face. The columns with *PCA+BBox* and *LBP+BBox* report the results we got by computing the clothing descriptor

Table 4.4: Mis-Classification Rates (%) - 0% of FPs α=0.45 β = 0.5 on the Gallagher Dataset

Method	PCA + BBox	PCA + Segm.	LBP + BBox	LBP + Segm.
`Whole Collection`	29.29	34.51	23.23	25.43
`Track Ratio`	8.5	6.8	9.97	10.09
`Event-driven`	34.2	22.78	17.58	17.17
`Track Ratio`	7.56	7.72	7.62	8.12

on the rectangular area estimated as described in Section 4.2.3.1 and using as face descriptor eigenfaces and LBP respectively. The columns with *PCA+ Segm* and *LBP+ Segm* report the results we got by computing the clothing descriptor on the area extracted using the method in (67) and described in section 4.2.3.2 and using as face descriptor eigenfaces and LBP respectively. An example of the identities we got is shown in Fig. 4.12.

Tables 4.2 and 4.3 summarize the results we got on the Gallagher Dataset with and without false positives (FPs). Results reported in the tables show that all the methods are sensitive to false positives and that generally our technique outperforms the clustering-based ones. In (86), where associations were found by maximizing their likelihood (obtained estimating exponential distributions to model the match) and experiments were performed considering only the features *PCA+BBox*, we always got an higher TR (9.47 with 0% FPs and 11.37 with 100% of FPs). The mis-classification rate was instead comparable to the one computed by the present approach.

The parameters α and β were set to 0.45 and 0.5 respectively for all the experiments reported in the aforementioned tables. To measure how the parameter selection affects the performances, we computed the mis-classification rate and the track ratio for different values of the parameters α and β. We report the analysis on the Gallagher Dataset using PCA + BBox, albeit similar trends were gotten with the other feature sets. Fig. 4.10 shows the trend we got setting α to 0.4, 0.5 and 0.6 while β changes between 0 and 1. Of course, increasing β – the probability that a new identity has been discovered – the track ratio increases too, while the mis-classification rate decreases because of the over-clustering.

Table 4.5: Mis-Classification Rates (%) - with all the FPs α=0.45 $\beta = 0.5$ on the Gallagher Dataset

Method	PCA + BBox	PCA + Segm.	LBP + BBox	LBP + Segm.
`Whole Collection`	36.48	38.92	24.9	31.56
`Track Ratio`	9.47	7.5	13.12	11.09
`Event-driven`	39.86	27.12	24.2	21.73
`Track Ratio`	8	7.11	9.69	11.3

4.4.4 Performance of the Event-Driven Data Association algorithm

Fig. 4.11a and Fig. 4.11b show how in general the mis-classification rate and the track ratio increase depending on the length of the photo sequences (we used the Gallagher Dataset). Photos are not taken at the same instant but during a long period. In this case, persons' appearance, namely clothing, changes and it is difficult to find a reliable matching between instances. This is also the reason why the track ratio increases. Of course, the trend of such curves is collection dependent. In particular, for the mis-classification rate, there can be photo subsequences for which the re-identification rate is very high because, for example, all the faces are frontal, or clothing are well detectable, or few noise is present in the data. In Fig. 4.11a, the curve does not show a monotonic trend. The slight inflection is due to a group of photos that is organized with low error so that the overall mis-classification decreases.

Finally, we performed experiments segmenting the photos in events and then computing associations across events. In our experiments, the threshold for the QT clustering used to find the events has been set to 1 hour, and the distance between two timestamps is simply their absolute difference.

Tables 4.4 and 4.5 summarize the results we got considering different sets of features. For the sake of clarity, we report both the mis-classification rate and track ratio computed considering associations along the whole collection (that is using only the first level of our architecture) and considering event-driven associations (that is both the first and second levels of our architecture, see Fig. 4.1). Whilst performances are generally comparable in terms of mis-classification rate, the track ratio improves suggesting that probably some over-clustering in

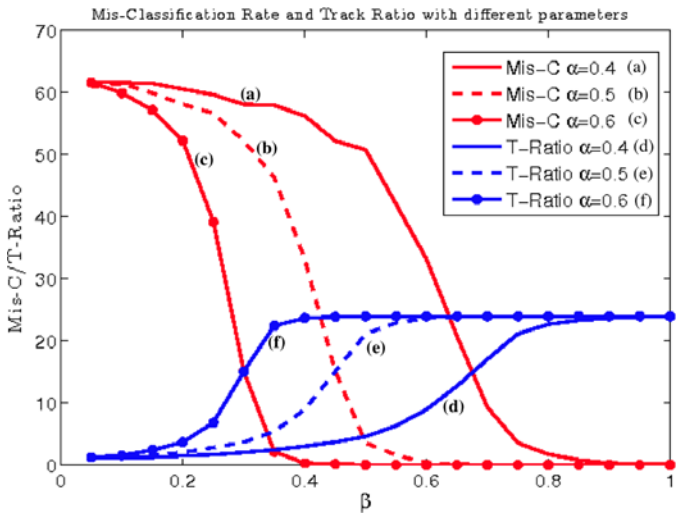
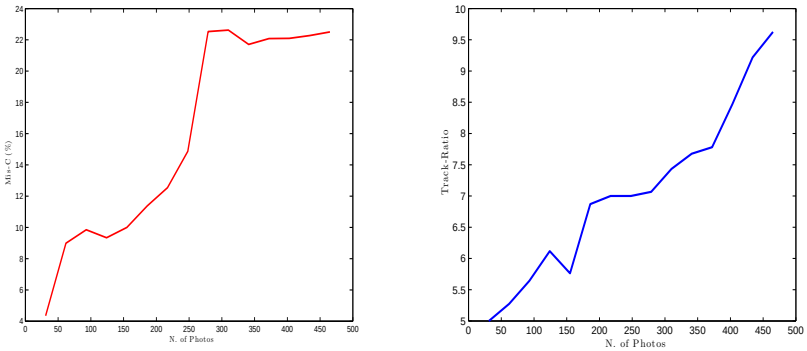


Figure 4.10: Mis-classification error (curves a, b and c) and track ratio (curves d, e and f) while changing the parameters α and β on the Gallagher dataset. Increasing the probability that a new person can be found increases the track ratio and, of course, decreases the mis-classification rate. In particular, the track ratio increases until the ratio between the number of detected faces and the number of detected identities has not been reached. In this case, the mis-classification rate will be 0% being each cluster composed by only a face.

the first case is due to unreliable clothing descriptions. Looking more closely at the results, we discover also that, in case of PCA features, improvements in performance are quite limited; indeed, whilst the track ratio decreases, the mis-classification rate increases. This result confirmed us that LBP-based descriptors is more accurate than PCA-based one. As concern the clothing detection, the detection method does not strongly affect the re-identification along the whole sequence. Improvements in the event driven association method are due to the more reliable clothing model estimated within each event. Moreover, identities are merged considering only the faces. The appearance model estimated within a rectangular bounding box and that estimated in the segmented clothing region (by (67)) give comparable performance. However, the two methods are not equivalent in terms of computational cost being clothing segmentation more time consuming.



(a) (b)

Figure 4.11: a) Mis-classification rate vs. length of the photo sequence; b)Track ratio vs. length of the photo sequence. These results were obtained on the Gallagher dataset.

4.4.5 Experiments on private collections

We performed the same kind of experiments on two private data collections we got from Flickr and we already described in Section 4.4.1. In Table 4.6, we report the results of the tests on the LBP+BBox features in the ideal case of no Unknown and/or FPs. Results confirm that generally our technique outperforms clustering techniques and that using event segmentation improves performance either in terms of mis-classification rate or in terms of track ratio. On these two collections, K-means performs in a competitive way, particularly at the same track ratio. However, we stress that this method needs the number of clusters being specified. These experiments also show as event segmentation permits to improve the performance of the method once again.

4.5 Discussions

Our experimental results show how solving a sequence of data association problems to cluster faces and clothing generally outperform other clustering techniques. Experimental evidence confirmed us that clothing information is reliable only locally in time, that is within short temporal windows, and performance sig-

Table 4.6: Mis-Classification Rates (%) - only known persons α=0.45 $\beta = 0.5$

Method	Family-1	Family-2
N. of Identities	5	5
K-Means	**16.92**	23.05
H. Clust	40.31	41.26
Ours - no Events	20.58	**22**
Track Ratio	12.2	28.8
K-Means Track-Ratio=1	24.92	34.26
H. Clust Track-Ratio=1	51.63	62.03
N. of Events	131	180
Ours - with Events	18.64	24.7
Track Ratio - with Events	12.8	20.6

nificantly decreases as the length of the photo sequence increases (see Fig. 4.11a and Fig. 4.11b). For this reason we associated faces within events using face and clothing descriptors, and then we associated face clusters across events considering only the face descriptors. This strategy showed to reduce over-clustering and/or the mis-classification rate.

We performed our experiments using a Matlab prototype on a Intel Core 2 Duo 2.53GHz. The most time consuming part is the feature extraction process that took several minutes (in particular segmenting the clothing region by the method in (67) where normalizing cut and graph cut are employed for each depiction). Once features where computed, finding the set of associations took very few seconds/minutes depending on the length of the sequence and on the length of the features used. For the Gallagher dataset, using eigenfaces (200 length feature vector) our method requires approximatively 1 minute to perform the re-identification. K-means is randomly initiated and in some cases the maximum number of iterations is required. On average, on 100 runs, K-means took approximatively 20 seconds. Hierarchical Clustering instead took about 2 minutes. When using LBP (a vector with a length greater than 2000), our method required about 3 minutes, K-means 25 minutes and Hierarchical Clustering about 10 minutes.

Finally we note that photo analysis tools recently started to appear on widely used services. Picasa (6) is receiving a lot of attention for its new tool "find person" that is closely related to our work; for comparison purposes, we tested

Picasa on the Gallagher Dataset. Picasa finds 1063 faces and, when using the "find persons" tool, it finds 84 groups. Looking more closely at these groups, we found that they were composed by only 805 faces among which we visually recognized 25 different individuals, whilst in the dataset there are 32 annotated persons. Moreover, Picasa discovered at least 2 identities that were not annotated in the dataset and another one that does not refer to any person (a statue). Analyzing the results, they are affected by over-clustering presenting a track ratio equal approximatively to 3.4. However, this result is not fully comparable with ours because not all the faces were clustered by Picasa and only clusters with more than a photo were considered (all the non clustered photos should be considered as groups composed by one face); the same would be true for the mis-classification rate.

Figure 4.12: Results on a sequence of five photos: each row represents a cluster of faces associated to the same identity. Last row (in red) is a case of over-clustering.

5

Mobile Multimedia

In this Section it will be shown an approach for automatic photo album management considering the scenario of mobile devices.

With the wide diffusion of mobile digital image acquisition devices, the need for managing a large number of digital images is quickly increasing. In fact, the storage capacity of such devices allow users to store hundreds or even thousands, of pictures that, without a proper organization, become useless. Users may be interested in using (i.e., browsing, saving, printing and so on) a subset of stored data according to some particular picture properties.

Previous work regarding CBIR on mobile devices has been mainly limited to particular problems like the generation of an initial query set (87), to energy efficiency (88) or to the development of a mobile front-end to traditional CBIR systems (89, 90, 91) in a client server framework. In our work we propose a fully automatic approach for image searching and browsing on mobile devices based on image clustering. Our goal is in fact the development of a system to improve user experience in managing photos on the mobile device itself without the need of transferring and processing them on a host computer.

Here image analysis and description is performed using the three-domain representation framework shown in Sect. 3, namely, faces, background and time of capture.

In our system, the faces are extracted from images so that it is possible to identify *who* is in the picture while the remaining part of the image is considered as image context. Low-level features, based on color and texture, are used to

identify different contexts (*where*) by analyzing the information stored in the image background. Nowadays more and more devices are equipped with GPS that allow to store camera position within the EXIF information, however GPS data is available only in outdoor environments so that visual analysis is required in any case. The *when* aspect is bound to when the picture was captured and is typically referred to temporal ranges or particular user events (e.g. *birthdays, weddings, travels*).

Faces, background and time information of each image in the collection is automatically organized using a mean-shift based clustering.

The proposed system has been tested on a real photo collection, i.e., 1000 images (VGA and double VGA images), captured in about two months by a mobile device. Tests have been performed on a traditional computer while taking into account the cost of each operation to be sure that the whole processing may be performed on a mobile device. Each image has been manually labeled to store information on the presence of faces, background characteristics and time of shooting. The face detection step brought to the extraction of 734 images of faces and four known people have been chosen so that each face is defined by an ID as reported in Table 5.1.

Table 5.1: Faces and background labels.

Faces	
ID	*id1, id2, id3, id4, unknown*
Background	
Type	*indoor, urban, green, snow*
Time	
Type	*birthday, Christmas, Christmas trip, winter 08/09, ski holiday*

Background images are classified according to four categories (*indoor*, *urban*, *green*, *snow*) representing some typical contexts mainly present in the collection. The results for the clustering of background are shown in the Table 5.2. All clusters with a single element have been discarded and label distribution is shown for the remaining seven clusters.

Faces are clustered according the parameters of the *Global Clustering Entropy*. Discarding all the clusters with less than two elements, the number of remaining

Table 5.2: Percentage occurrence of labels in generated clusters

	indoor	urban	green	snow
Cl 1	-	64%	36%	-
Cl 2	12%	-	-	88%
Cl 3	-	58%	27%	15%
Cl 4	43%	36%	-	21%
Cl 5	-	55%	41%	4%
Cl 6	44%	-	56%	-
Cl 7	24%	-	27%	49%

clusters is equal to 6 and the distribution is shown in Table 5.3. The id from 1 to 4 are the most recurrent in image repository, all the other faces are associated to a "unknown"label.

Table 5.3: Percentage occurrence of identities in generated clusters

	Pers 1	Pers 2	Pers 3	Pers 4	unknown
Cl 1	100%	-	-	-	-
Cl 2	3%	-	5%	78%	14%
Cl 3	-	32%	-	-	68%
Cl 4	-	-	74%	-	26%
Cl 5	77%	-	-	-	23%
Cl 6	-	67%	19%	14%	-

Time information is clustered considering the found parameters and results evaluated according to manually given temporal labels (*birthday*, *Christmas*, *Christmas trip*, *winter 08/09*, *ski holiday*).

The most frequent 3-tuple for each cluster are shown in Fig. 5.1 and reported in Table 5.5.

The main contribution of the proposed work is to consider the mobile multimedia device as a standalone device that allows an user to instantly manage its own pictures collection. Thus, all image representation and clustering steps have been performed simulating the device constraints, that is taking into account the cost of each operation while optimizing the whole system. Some points (e.g.,

Table 5.4: Percentage occurrence of time labels (TL) in generated clusters

	birthday	*Christmas*	*Christmas trip*	*winter 08/09*	*ski holiday*
cl 1	-	23%	77%	-	-
cl 2	54%	-	-	26%	20%
cl 3	-	-	-	-	100%
cl 4	-	63%	11%	26%	-
cl 5	-	-	-	100%	-
cl 6	-	-	100%	-	-
cl 7	11%	7%	53%	-	29%
cl 8	-	-	-	64%	36%

Table 5.5: The most frequent 3-tuple for each cluster.

	Cluster 1	*Cluster 2*	*Cluster 3*	*Cluster 4*
who	person 3	person 2	person 4	person 1
where	indoor	indoor	snow	green
when	winter 08/09	winter 08/09	ski holiday	Christmas trip

objective vs subjective clustering evaluation, system performances while using collections of different size, time of execution on different mobile devices) could be better investigated and this will be subject of future work.

Figure 5.1: Image clusterization exploiting multi-domain representation.

6

Conclusions and Future Work

In this thesis, novel image and face analysis techniques for personal photo album management have been described and evaluated. In personal photo collection the user is mainly interested in *who* is in the picture (usually a relatively small number of different individuals) and *where* and *when* the picture was shot. This consideration allowed us for a precise definition of what we wanted to obtain from our research.

Who, *where* and *when* are the fundamental aspects of photo information we were interested in, thus input images have been intrinsically split in three domains of interest.

The problem has been tackled by considering two processing levels. At the first level, relevant features (i.e., faces, clothing, background, time information) are extracted from images. At the second level, extracted information is used as input for novel image organization systems.

We presented a novel facial feature extraction approach that could be used for normalizing Viola-Jones detected faces noticing that rectangle features creates some structure on features position over the detected faces. Experimental evaluation has been provided and the proposed method has been successfully used as preprocessing step for face recognition task.

Moreover, noticing that face recognition techniques are usually evaluated on public datasets, acquired in controlled conditions (fixed pose, illumination and so on), we reported a comparison of results obtained by applying some of the most used face recognition techniques on a personal photo collection. This study

allowed us for better understanding which method can be considered as the most suitable for obtaining discriminative face descriptors in our application domain.

In the second part of this thesis, a novel approach to cluster visual data driven by a clusterization measure has been presented. The aim of the proposed work was to use a common approach to manage multiple aspects in a personal photo album. Known systems manage faces and typically allow for queries about them. Here, queries regarding people, time and background are dealt with in a homogeneous way. The proposed system has been tested on a realistic set, i.e. a personal photo album, and experimental results are very interesting. The experimental results we reported can be considered as an objective clustering of the input data. Moreover, we informally analyzed the behavior of the system in different cases and an example of clustering based on different aspects has been shown to give an insight of typical results.

Another contribution to the problem of personal photo organization has been proposed in Sect. 4. We started our formulation by considering an event as a sequence of photos taken within a meaningful temporal window. The main intuition of our method relies in fact on the empirical observation that within the same event, photos are taken in similar conditions and depicted persons generally look similar, i.e. clothing and poses do not strongly change across photos and can be successfully used for the re-identification process. People re-identification has been formulated as a data association problem that is we re-identify persons by finding a maximum matching on a bipartite graph. Experimental results on three personal photo sequences showed that our approach outperforms clustering techniques and usually improves system performance either in terms of mis-classification rate or in terms of track ratio.

Considering also the innovative scenario of mobile multimedia, it has been described a fully automatic approach for image searching and browsing on mobile devices. The goal of the proposed system was to improve user experience in managing (i.e., browsing, saving, printing and so on) photos on the mobile device itself without the need of transferring and processing them on a host computer. The main contribution of the proposed work is to consider the mobile multimedia device as a standalone device that allows an user to instantly manage its own pictures collection.

Some points (e.g., objective vs subjective clustering evaluation, system performances while using collections of different size, time of execution on different mobile devices) could be better investigated and this will be subject of future work.

In future works it will be also investigated how to take advantage from other contextual information (such as where the picture was acquired) to organize the collection and if the event definition approach could be adapted to deal with information extracted from the background.

Moreover, with the rapid popularity of the Web 2.0 (blogs, social networks, etc.) users are encouraged to share their personal data. In such scenario, personal images are generally accompanied by rich contextual information, such as tag, category, title, metadata, comments, and viewer ratings. This information should be used to develop new techniques for organizing, indexing and managing such rich social media contents. In future work, social media analysis will be addressed.

References

[1] E. Ardizzone, M. La Cascia, and F. Vella. **A novel approach to personal photo album representation and management**. In *Proceedings of Multimedia Content Access: Algorithms and systems II. IS&T SPIE Symposium on Electronic Imaging*, **6820**, 2008. 1, 27

[2] E. Ardizzone, M. La Cascia, and F. Vella. **Automatic Image Representation for Content-Based Access to Personal Photo Album**. In *Advances in Visual Computing. LNCS*, **4842**, pages II: 265–274, 2007. 1

[3] H. Kang and B. Shneiderman. **Visualization Methods for Personal Photo Collections: Browsing and Searching in the PhotoFinder**. In *Proc. of International Conference on Multimedia & Expo (ICME 2000)*, 2000. 2

[4] L. Zhang, L. Chen, M. Li, and H. Zhang. **Automated annotation of human faces in family albums**. *Proc. of Conference on Multimedia (MM 2003)*, pages 355–358, 2003. 2, 57

[5] iPhoto. http://www.apple.com/ilife/iphoto, 2010. 2

[6] Picasa. http://picasa.google.com, 2010. 2, 55, 80

[7] J.Y. Choi, S. Yang, Y.M. Ro, and K.N. Plataniotis. **Face annotation for personal photos using context-assisted face recognition**. *Proc. of International Conference on Multimedia Information Retrieval (MIR 2008)*, pages 44–51, 2008. 2, 51, 53, 61, 62, 74

[8] Wei-Ta Chu, Ya-Lin Lee, and Jen-Yu Yu. **Using context information and local feature points in face clustering for consumer photos**. In *Proc. of International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2009)*, pages 1141 –1144, 2009. 2, 53, 74

[9] Ming-Hsuan Yang, D.J. Kriegman, and N. Ahuja. **Detecting faces in images: a survey**. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, **24**(1):34–58, Jan 2002. 8, 9

[10] Ashok Samal and Prasana A. Iyengar. **Automatic recognition and analysis of human faces and facial expressions: a survey**. *Pattern Recogn.*, **25**(1):65–77, 1992. 9

[11] G. Yang and T. S. Huang. **Human face detection in complex background**. *Pattern Recognition*, **27**(1):53–63, 1994. 9

[12] Kin Choong Yow and Roberto Cipolla. **Feature-based human face detection**. *Image Vision Comput.*, **15**(9):713–735, 1997. 9

[13] T.K. Leung, M.C. Burl, and P. Perona. **Finding faces in cluttered scenes using random labeled graph matching**. *Computer Vision, IEEE International Conference on*, **0**:637, 1995. 9

[14] Yali Amit, Donald Geman, and Bruno Jedynak. **Efficient Focusing and Face Detection**. In *FACE RECOGNITION: FROM THEORY TO APPLICATIONS*, pages 143–158. Springer-Verlag, 1998. 9

[15] M.F. Augusteijn and T.L. Skufca. **Identification of human faces through texture-based feature recognition and neural network technology**. In *Neural Networks, 1993., IEEE International Conference on*, pages 392 –398 vol.1, 1993. 9

[16] Ying Dai and Yasuaki Nakano. **Face-texture model based on SGLD and its application in face detection in a color scene**. *Pattern Recognition*, **29**(6):1007 – 1017, 1996. 9

[17] James L. Crowley and Francois Berard. **Multi-Modal Tracking of Faces for Video Communications**. In *Proceedings of the 1997 Conference on Computer Vision and Pattern*

Recognition (CVPR '97), CVPR '97, pages 640–, Washington, DC, USA, 1997. IEEE Computer Society. 9

[18] David Saxe and Richard Foulds. **Toward Robust Skin Identification in Video Images**. In *Proceedings of the 2nd International Conference on Automatic Face and Gesture Recognition (FG '96)*, FG '96, pages 379–, Washington, DC, USA, 1996. IEEE Computer Society. 9

[19] D. Chai and K. N. Ngan. **Locating Facial Region of a Head-and-Shoulders Color Image**. In *Proceedings of the 3rd. International Conference on Face & Gesture Recognition*, FG '98, pages 124–, Washington, DC, USA, 1998. IEEE Computer Society. 9

[20] J. Miao, B.C. Yin, K.Q. Wang, L.S. Shen, and X. Chen. **A hierarchical multiscale and multiangle system for human face detection in a complex background using gravity-center template**. *PR*, **32**(7):1237–1248, July 1999. 10

[21] Alan L. Yuille, Peter W. Hallinan, and David S. Cohen. **Feature extraction from faces using deformable templates**. *Int. J. Comput. Vision*, **8**:99–111, August 1992. 10

[22] Paul Viola and Michael J. Jones. **Robust Real-Time Face Detection**. *Int. J. Comput. Vision*, **57**(2):137–154, 2004. 10

[23] Yoav Freund and Robert E. Schapire. **A decision-theoretic generalization of on-line learning and an application to boosting**. In *EuroCOLT '95: Proceedings of the Second European Conference on Computational Learning Theory*, pages 23–37, London, UK, 1995. Springer-Verlag. 10

[24] W. Zhao, R. Chellappa, P. J. Phillips, and A. Rosenfeld. **Face recognition: A literature survey**. *ACM Comput. Surv.*, **35**(4):399–458, 2003. 11

[25] A.L. Yuille, D.S. Cohen, and P.W. Hallinan. **Feature extraction from faces using deformable templates**. *Computer Vision and Pattern Recognition, 1989. Proceedings CVPR '89., IEEE Computer Society Conference on*, pages 104–109, Jun 1989. 11

[26] T. F. Cootes, C. J. Taylor, D. H. Cooper, and J. Graham. **Active shape models—their training and application**. *Comput. Vis. Image Underst.*, **61**(1):38–59, 1995. 11

[27] Michael Kass, Andrew Witkin, and Demetri Terzopoulos. **Snakes: Active contour models**. *International Journal of Computer Vision*, **V1**(4):321–331, January 1988. 11

[28] T. F. Cootes, G. J. Edwards, and C. J. Taylor. **Active Appearance Models**. *Proceedings of the European Conference on Computer Vision*, **2**:484–498, 1998. 12

[29] Gary B. Huang, Vidit Jain, and Erik Learned-Miller. **Unsupervised joint alignment of complex images**. In *Proc. of International Conference on Computer Vision (ICCV 2007)*, 2007. 12

[30] T. L. Berg, A. C. Berg, J. Edwards, M. Maire, R. White, Yee-Whye Teh, E. Learned-Miller, and D. A. Forsyth. **Names and faces in the news**. In *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, **2**, pages II–848–II–854 Vol.2, 2004. 12

[31] A.C. Berg and J. Malik. **Geometric blur for template matching**. *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, **1**:I–607–I–614 vol.1, 2001. 12

[32] P. Viola and M. Jones. **Rapid object detection using a boosted cascade of simple features**. *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, **1**:I–511–I–518 vol.1, 2001. 12, 14

[33] M. Turk and A. Pentland. **Eigenfaces for Recognition**. *Journal of Cognitive Neuroscience*, **3**(1):71–86, 1991. 12

[34] Cordelia Schmid, Roger Mohr, and Christian Bauckhage. **Evaluation of Interest Point Detectors**. *International Journal of Computer Vision*, **37**(2):151–172, 2000. 13

[35] Konstantinos G. Derpanis. **The Harris Corner Detector**, 2004. 13

[36] J. Duchon. **Spline minimizing rotation-invariant semi-norms on sobolev spaces**. *Lecture Notes in Math*, **571**:85–100, 1977. 16

[37] W. Zhao and R. Chellappa. **A Guided Tour of Face Processing**. In *Face Processing: Advanced Modeling and Methods*, pages 3–53. Academic Press, 2006. 21

[38] Peter N. Belhumeur, J. P. Hespanha, and David J. Kriegman. **Eigenfaces vs. Fisherfaces: Recognition Using Class Specific Linear Projection**. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **19**:711–720, 1997. 23

[39] Ronald A. Fisher. **The use of multiple measurements in taxonomic problems**. *Annals Eugen.*, **7**:179–188, 1936. 23

[40] A.M. Martinez and A.C. Kak. **PCA versus LDA**. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, **23**(2):228–233, Feb 2001. 23

[41] T. Ojala, M. Pietikäinen, and T. Mäenpää. **Multiresolution gray-scale and rotation invariant texture classification with local binary patterns**. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 971–987, 2002. 24, 26

[42] T. Ahonen, A. Hadid, and M. Pietikäinen. **Face recognition with local binary patterns**. *Proc. of European Conference on Computer Vision (ECCV 2004)*, pages 469–481, 2004. 24, 60, 61

[43] Yael Adini, Yael Moses, and Shimon Ullman. **Face Recognition: The Problem of Compensating for Changes in Illumination Direction**. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **19**:721–732, 1997. 25

[44] T. L. Berg, A. C. Berg, J. Edwards, M. Maire, R. White, Y. W. Teh, E. Learned-Miller, and D. A. Forsyth. **Names and Faces in the News**. In *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, 1994. 27, 30, 31

[45] H. Kang and B. Shneiderman. **Visualization Methods for Personal Photo Collections: Browsing and Searching in the PhotoFinder**. In *Proceedings of IEEE International Conference on Multimedia and Expo (ICME)*, 2000. 29

[46] L. Zhang, L. Chen, M. Li, and H. Zhang. **Automated Annotation of Human Faces in Family Albums**. In *Proceedings of ACM International Conference on Multimedia*, 2003. 29

[47] L. Zhang, Y. Hu, M. Li, W. Ma, and H. Zhang. **Efficient Propagation for Face Annotation in Family Albums**. In *Proceedings of ACM International Conference on Multimedia*, 2004. 29

[48] M. Abdel-Mottaleb and L. Chen. **Content-based Photo Album Management using Faces' Arrangement**. In *Proceedings IEEE International Conference on Multimedia and Expo (ICME)*, 2004. 29

[49] A. Girgensohn, J. Adcock, and L. Wilcox. **Leveraging Face Recognition Technology to Find and Organize Photos**. In *Proceedings of ACM International Conference on Multimedia Information Retrieval (MIR)*, 2004. 29

[50] M. Naaman, R. B. Yeh, H. Garcia-Molina, and A. Paepcke. **Leveraging Context to Resolve Identity in Photo Albums**. In *Proceedings of ACM/IEEE Joint Conference on Digital Libraries (JCDL)*, 2005. 29

[51] B. N. Lee, W.-Y. Chen, and E. Y. Chang. **A Scalable Service for Photo Annotation, Sharing and Search**. In *Proceedings of ACM International Conference on Multimedia*, 2006. 29

[52] J. Cui, F. Wen, R. Xiao, Y. Tian, and X. Tang. **EasyAlbum: An Interactive Photo Annotation System Based on Face Clustering and Re-ranking**. In *Proceedings of ACM Special Interest Group on Computer-Human Interaction*, 2007. 29

[53] S. Krishnamachari and M. Abdel-Mottaleb. **Hierarchical clustering algorithm for fast image retrieval**. In *IS&T SPIE Conference on Storage and Retrieval for Image and Video databases VII.*, 1999. 30

[54] J.Y. Chen, C.A. Bouman, and J.C. Dalton. **Hierarchical Browsing and Search of Large Image Databases**. *IEEE Transaction on Image Processing*, **9**(3):442–455, March 2000. 30

[55] D. Deng. **Content based comparison of image collection via distance measuring of self organized maps**. In *Proceedings of 10th International Multimedia Modelling Conference*, 2004. 30

[56] J. Goldberg, S. Gordon, and H. Greenspan. **Unsupervised Image-Set Clustering Using an Information Theoretic Framework**. *IEEE Transaction on Image Processing*, **15**(2):449–458, 2006. 30

[57] C.-H. Li, C.-Y. Chiu, C.-R. Huang, C.-S. Chen, and Lee-Feng Chien. **Image content clustering and Summarization for photo collections**. In *Proceedings of ICME*, pages 1033–1036, 2006. 30

[58] Y. Song and T. Leung. **Context-aided human recognition clustering**. In *Proceedings of ECCV*, **3**, 2006. 30

[59] J. Cuiy, F. Wenz, R. Xiaoz, Y. Tianx, and X. Tang. **EasyAlbum: An Interactive Photo Annotation System Based on Face Clustering and Re-ranking**. In *Proceedings of CHI*, 2007. 30

[60] P. Viola and M. Jones. **Rapid Object Detection using a Boosted Cascade of Simple Features**. In *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2001. 31, 34

[61] Matthew A. Turk and Alex P. Pentland. **Face Recognition Using Eigenfaces**. In *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, 1991. 32

[62] A.K. Jain and F. Farrokhnia. **Unsupervised texture segmentation using Gabor filters**. In *Systems, Man and Cybernetics, 1990. Conference Proceedings., IEEE International Conference on*, 1990. 34

[63] X.Z. Liu, L. Zhang, M.J. Li, H.J. Zhang, and D.X. Wang. **Boosting image classification with LDA-based feature combination for digital photograph management**. *Pattern Recognition*, **38**(6):887–901, June 2005. 35

[64] D. Comaniciu and P. Meer. **Mean Shift: A Robust Approach Toward Feature Space Analysis**. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, **24**:603–619, May 2002. 36, 42, 43

[65] J.C. Bezdek. *Pattern Recognition with Fuzzy Object Function*. Plenum, 1981. 38

[66] K.L. Wu and M.S. Yang. **A cluster validity index for fuzzy clustering**. *Pattern Recognition Letters*, **26**:1275–1291, 2005. 38

[67] A. Gallagher and T. Chen. **Clothing Cosegmentation for Recognizing People**. In *Proc. of Computer Vision and Pattern Recognition (CVPR 2008)*. IEEE, 2008. 51, 52, 54, 57, 61, 63, 72, 73, 74, 76, 78, 80

[68] R. Jain and P. Sinha. **Content Without Context is Meaningless**. *Proc. of Conference on Multimedia (MM 2010)*, 2010. 53

[69] Ming Zhao, Yong Teo, Siliang Liu, Tat-Seng Chua, and Ramesh Jain. **Automatic Person Annotation of Family Photo Album**. In *Image and Video Retrieval*, **4071** of *Lecture Notes in Computer Science*, pages 163–172. Springer Berlin / Heidelberg, 2006. 54

[70] L. Zhang, Y. Hu, M. Li, W. Ma, and H. Zhang. **Efficient propagation for face annotation in family albums**. *Proc. of Conference on Multimedia (MM 2004)*, pages 716–723, 2004. 54

[71] Elie El-Khoury, Christine Senac, and Philippe Joly. **Face-and-Clothing Based People Clustering in Video Content**. In *Proc. of International Conference on Multimedia Information Retrieval (MIR 2010)*, pages 1–10. ACM, 2010. 54, 62

[72] J. Sivic, C.L. Zitnick, and R. Szeliski. **Finding people in repeated shots of the same scene**. *Proc. of British Machine Vision Conference (BMVC 2006)*, **3**:909–918, 2006. 54

[73] Yang Song and Thomas Leung. **Context-Aided Human Recognition Clustering**. In *Computer Vision ECCV 2006*, **3953** of *Lecture Notes in Computer Science*, pages 382–395. Springer Berlin / Heidelberg, 2006. 54

[74] H.W. Kuhn. **The Hungarian method for the assignment problem**. *Naval research logistics quarterly*, **2**(1-2):83–97, 1955. 55, 68

[75] Laurie J. Heyer, Semyon Kruglyak, and Shibu Yooseph. **Exploring Expression Data: Identification and Analysis of Coexpressed Genes**. *Genome Research*, **9**(11):1106–1115, November 1999. 58, 59

[76] X. Tan and B. Triggs. **Enhanced local texture feature sets for face recognition under difficult lighting conditions**. In *Proc. of International Conference on Analysis and Modeling of Faces and Gestures*, pages 168–182. Springer-Verlag, 2007. 60

[77] M. Turk and A. Pentland. **Eigenfaces for Recognition**. *Journal of Cognitive Neuroscience*, **3**(1):71–86, 1991. 60

[78] P.J. Phillips, H. Wechsler, J. Huang, and P.J. Rauss. **The FERET database and evaluation procedure for face-recognition algorithms**. *Image and Vision Computing*, **16**(5):295–306, 1998. 61

[79] P.J. Phillips, H. Moon, S.A. Rizvi, and P.J. Rauss. **The FERET evaluation methodology for face-recognition algorithms**. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **22**(10):1090–1104, 2002. 61

[80] P. Kakumanu, S. Makrogiannis, and N. Bourbakis. **A survey of skin-color modeling and detection methods**. *Pattern Recogninton*, **40**(3):1106–1122, 2007. 63

[81] D. Androutsos, KN Plataniotiss, and AN Venetsanopoulos. **Distance measures for color image retrieval**. In *Proc. of International Conference on Image Processing, (ICIP 98)*, **2**, pages 770–774. IEEE, 1998. 65

[82] J.F. Lawless. *Statistical models and methods for lifetime data*. Wiley New York, 1982. 68

[83] J.L. Gross and J. Yellen. *Graph theory and its applications*. CRC press, 2006. 68

[84] T. Huang and S. Russell. **Object identification in a bayesian context**. *Int. Joint Conf. on Artificial Intel.*, **15**:1276–1283, 1997. 69

[85] Y. Rubner, C. Tomasi, and L.J. Guibas. **A metric for distributions with applications to image databases**. *Proc. of International Conference on Computer Vision, (ICCV 1998)*, 1998. 71

[86] L. Lo Presti, M. Morana, and M. La Cascia. **A Data Association Algorithm for People Re-Identification in Photo Sequences**. In *Proc. of International Workshop on Multimedia Information Processing and Retrieval (MIPR), (to appear)*. IEEE, 2010. 76

[87] Deok-Hwan Kim, Chan Young Kim, and Yoon Ho Cho. **Automatic Generation of the Initial Query Set for CBIR on the Mobile Web**. In *PCM (1)*, pages 957–968, 2005. 82

[88] Karthik Kumar, Yamini Nimmagadda, Yu-Ju Hong, and Yung-Hsiang Lu. **Energy conservation by adaptive feature loading for mobile content-based image retrieval**. In *ISLPED '08: Proceeding of the thirteenth international symposium on Low power electronics and design*, pages 153–158, New York, NY, USA, 2008. ACM. 82

[89] I. Ahmad, S. Abdullah, S. Kiranyaz, and M. Gabbouj. **Content-Based Image Retrieval on Mobile Devices**. In *Proceedings of SPIE (Multimedia on Mobile Devices)*, 2005. 82

[90] J. S. Hare and P.H. Lewis. **Content-based image retrieval using a mobile device as a novel interface**. In *Storage and Retrieval Methods and Applications for Multimedia*, 2005. 82

[91] M. Gabbouj, I. Ahmad, Malik Y. Amin, and S. Kiranyaz. **Content-based Image Retrieval for Connected Mobile Devices**. In *Proceedings of Second International Symposium on Communications, Control and Signal Processing (ISCCSP)*, 2006. 82

Declaration

I herewith declare that I have produced this paper without the prohibited assistance of third parties and without making use of aids other than those specified; notions taken over directly or indirectly from other sources have been identified as such. This paper has not previously been presented in identical or similar form to any other Italian or foreign examination board.

The thesis work was conducted from January 2008 to February 2011 under the supervision of Prof. Marco La Cascia at University of Palermo.

Palermo - February 15, 2011