University of Padova

Department of Information Engineering
Master Degree in Computer Engineering
3 October 2016

A Multi-modal Biometric System for Selective Access Control

Candidate:
Elisabetta Stefani
Matricola 1106106

Thesis Advisor:
Chiar.mo Prof. Carlo Ferrari

Academic Year 2015-2016

Alla mia mamma,
riesco a percepire quanto tu sia orgogliosa di me.

Al mio papà,
perché un ingegnere può.

A Marco,
inestimabile compagno di vita.

Abstract

The goal of this thesis is the design and the implementation of an adaptive multi-modal biometric system with serial aquisition mode, intended to manage the accesses of structure, e.g. a company structure, according to a predefined set of security levels and requirements, stated in a formal way in the Service Level Agreement at a negotiation phase.

In a multi-modal process multiple biometric traits are collected from the same individual, requiring different sensors. The chosen and combined traits are two physical characteristics: face, the most common biometric feature used by humans to recognize one another, employed for face identification and iris, the annular region of the eye bounded by the pupil and the sclera (white of the eye) on either side, employed for iris verification.

The system has not a fixed and predefined behaviour. There are many possible working flows, since the actual response of the recognition process depends on the output of the decision making modules that compose the system.

The proposed system is tested on the AT&T Face Database and the UBIRIS database.

The deployment phase is described, together with the results from the testing.

Contents

List of Figures

Chapter 1

Introduction

1.1 General scenario

Infrastructure security is the security provided to protect infrastructure, such as airports, hospitals, company structures or corporate headquarters. In the absence of robust control and supervision schemas, these critical infrastructures are vulnerable to the wiles or an impostor. Therefore, they normally utilize information technology to rely on a dependable security system. The need for reliable user authentication techniques has increased in the wake of heightened concerns about security and rapid advancements in networking, communication and mobility.

In the design of a security system there are three fundamental concepts to take into account: identity, authentication and authorization [1].

Identity is the answer to the question "Who are you?", a public declaration of who someone is (e.g. a user ID). Identity is a person's claim about themselves and that claim is made using something that is publicly available.

Authentication is the answer to the question "OK, how can you prove it?" and is necessary to prove that the presented identity is indeed linked to the right person and not to someone else.

Authorization, once the authentication phase succeeded, is the answer to the question "What can I do?" and says which resources a person is allowed to access, assigning them privileges based on some attribute of their identity.

In literature, the problem of studying and characterizing the best theoretical method to obtain an effective and efficient system for individuals authentication has been widely addressed: there is a huge amount of possible solutions, each employing different required credentials, different algorithms and different practical deployments [2]. Nevertheless, the goal of all those systems is to minimize (possibly set to zero) the number of users who are

impostors but manage to be authenticated, or are enrolled users but are rejected. The balance between the False Acceptance Rate (FAR) and the False Rejection Rate (FRR), possibly leading to an Equal Error Rate (EER) where the FAR and the FRR values are equal, depends on the security levels and requirements stated in the Service Level Agreement (SLA), a contract which formally describes particular aspects of a service (scope, qualities, responsibilities), agreed between the service provider and the service user at the negotiation phase [3].

1.2 Thesis scope and goal

The main focus of this work is on authentication, the process of confirming the identity of an entity, which involves verifying the validity of at least one form of identification (e.g. documents, digital certificate, biometric identifiers, etc.).

The most interesting way in which someone may be authenticated fall into the category having "something that the user *is* or *does*" as the factors of authentication, known as inherence factors, i.e. physical or behavioural characteristics of an individual.

Other factors of authentication are knowledge factors ("something that the user *knows"*) and ownership factors ("something that the user *has*").

Examples of inherence factors are static or dynamic biometric parameters: physical parameters like face, fingerprint, iris and retinal pattern; behavioural parameters like signature, voice and gait.

These biometric identifiers are distinctive and measurable characteristics which can be used to label and describe individuals in a unique way.

There is no proof of correlation between two different biometric parameters of an individual, so the combination of independent information in a system can lead to an improvement of the accuracy (but it is also more invasive). Multi-modal biometric system are more reliable than uni-modal systems due to the presence of multiple and independent piece of evidence. They address the problem of non-universality, since multiple traits ensure sufficient population coverage and provide anti-spoofing measures by making it difficult for an intruder to simultaneously spoof the multiple biometric traits of a legitimate user.

The goal of this thesis is the design and the implementation of a multi-modal biometric system with serial aquisition mode. In a multi-modal process multiple biometric traits are collected from the same individual, requiring different sensors. The chosen traits are two physical characteristics: face,

the most common biometric feature used by humans to recognize one another, and iris, the annular region of the eye bounded by the pupuil and the sclera (white of the eye) on either side.

The face parameter is employed for face identification; the iris parameter is employed for iris verification.

A challenge-response type of authentication can be facilitated using multimodal biometric systems, ensuring that a "live" user is indeed present at the point of data acquisition, since these systems ask them to present a particular and predefined subset of biometric traits [2][4].

The mode in which the multi-modal system operates is the serial mode: it means that the two biometric characteristics do not have to be acquired simultaneously and that a decision could be arrived at without acquiring all the traits. This last aspect is very important, especially when there are time constraints, because it leads to a reduction of the overall recognition time and the serial approach helps to save users' time, that can be a thorny issue in a working environment, especially if the employees have to be authenticated more than once a day.

Furthermore, this system is intended to operate both in the verification and in the identification mode [5]. The first phase is an identification process: a person's face is acquired to establish the identity of the individual; it is a one-to-many comparison and search process in which the biometric characteristic set against all the database to find biometric references with a specific degree of similarity.

The second phase is about verification: a person's iris is acquired in order to verify the claimed identity (e.g. user name) of the unknown individual; it is a one-to-one comparison of the submitted biometric characteristic set against a specified stored biometric references and returns both the comparison score and the decision.

Both these two operational modes are preceded by the enrolment process. In this process, a subject presents their biometric characteristics to the sensor along with their non-biometric information. These information related to subjects could be name, social security number, driver license's number, etc. Thus, biometric features extracted from the captured sample and the non-biometric information are stored in the database.

It is clear that determining "true" identity of an individual is beyond the scope of any biometric technology. Rather, biometric technology can only link a person to a biometric pattern and any identity data (e.g. name) and personal attributes (age, gender, profession, etc.) presented at the time of enrolment in the system [6].

The context in which this multi-modal biometric system is supposed to

operate is a real context; its deployment is within an indoor application, e.g. an infrastructure security system of a company, and it should be responsible for regulating the access to the (company) structure and inside the various (company) areas.

Noticeably, biometrics can be used more than once: it means that the use of biometric parameter is not limited to an entrance point, but they can provide a re-identification process (one or more) because of security constraints, which can vary in time or space. Otherwise, they can be useful to monitor the actual presence of individuals, e.g. employees, inside a building or a room.

So, from a general point of view, the proposed multi-modal biometric system can be also seen as an application of a pervasive indoor system (e.g. domotics).

It is clear that, in commercial applications, the addition or replacement of existing personal recognition methods with biometrics-based solutions should be based on a cost-benefit analysis, that will not be taken into consideration throughout this work.

Chapter 2

State of the art

In this chapter several existent solutions of biometric system are taken into account and two of the biometric traits a system can depend on are discussed: face and iris.

2.1 Multi-biometric systems

Multi-biometric systems are biometric systems which consolidate multiple sources of biometric evidences; the integration of evidences is known as fusion and there are various levels of fusion, which can be divided into two broad categories: pre-classification (fusion before matching) and post-classification (fusion after matching). Besides, depending on the nature of the sources, multi-biometric systems can be classified into different categories; for instance, multi-sensor systems, multi-algorithm systems, multi-instance systems, multi-sample systems, multi-modal systems and hybrid systems [7].

2.1.1 Levels of Fusion

Fusion in multi-biometric systems can be performed utilizing information available in any of the modules (data capture module to decision module). Fusion can take place at these levels: sensor level, feature level, score level, rank level and decision level [7].

In sensor level fusion raw data captured by the sensor(s) are combined. In feature level fusion features originating from each individual biometric process are combined to form a single feature set or vector. In score level fusion, match scores provided by different matchers indicating degree of similarity (differences) between the input and enrolled templates, are consolidated to reach the final decision. In rank level fusion each biometric sub-system as-

signs a rank to each enrolled identity and the ranks from the subsystems are combined to obtain a new rank for each identity. Lastly, in decision level fusion the final boolean result from every biometric subsystem are combined to obtain final recognition decision.

2.1.2 Sources of Evidences

Various sources of biometric information can be used in a multi-biometric system. Based on these sources, multi-biometric systems can be classified into six different categories: multi-sensor, multi-algorithm, multi-instance, multi sample, multi-modal and hybrid [7].

Multi-sensor systems employ multiple sensors to capture a single biometric trait in order to extract diverse information. In multi-algorithm systems, multiple algorithms are applied to the same biometric data. Multi-instance systems use multiple instances of the same body trait (for example, left and right irises or left and right index fingers). In multi-sample system, multiple samples of the same biometric trait are acquired using the same sensor in order to obtain a more complete representation of the underlying trait. Multi-modal systems combine evidences obtained from different (two or more) biometric traits. In literature, hybrid is used to refer to those systems integrating two or more of the scenarios mentioned earlier.

2.2 Biometric identifiers

A number of biometric characteristics exist and are in use in various application. Examples of biometric characteristics are: DNA, ear, face, facial thermogram, hand thermogram, hand vein, fingerprint, gait, hand geometry, iris, palm-print, retina, signature and voice[Fig.2.1].

Each biometric has its strengths and weaknesses, and the choice depends on the application. No single biometric is expected to effectively meet the requirements of all the applications, thus each biometric technique is admissible and there is no "optimal" biometric characteristic [2]. The match between a specific biometric and an application is determined depending upon the operational mode of the application and the properties of the biometric characteristic [Fig.2.2]. They are evaluated especially on their performance, cost, intrusiveness and accuracy.

As mentioned earlier, there are several advantages in using biometrics: they cannot be lost or forgotten and they require the person to be recognized to be present at the point of recognition. Besides, it is difficult to forge them.

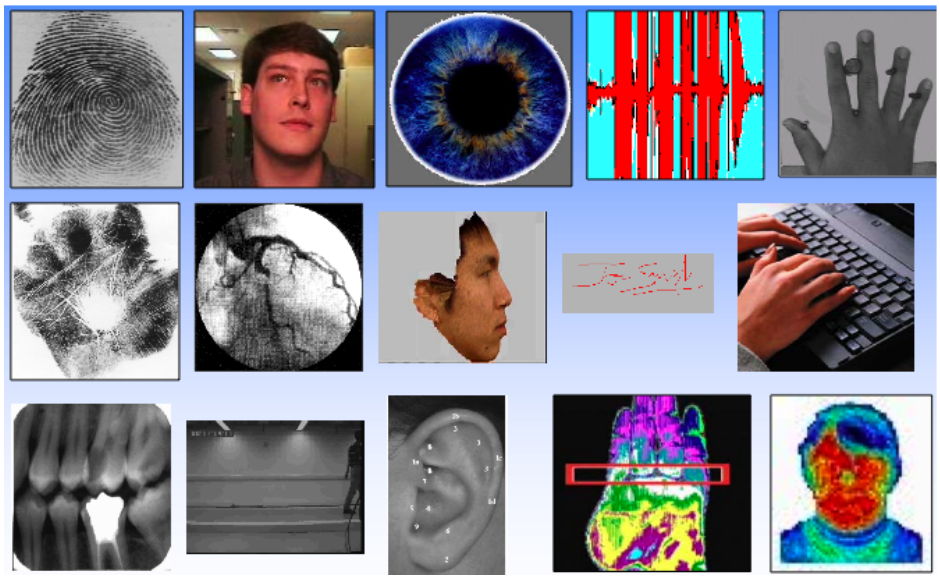

Figure 2.1: Example of physical or behavioural biometric identifiers

Biometric identifier	Universality	Distinctiveness	Permanence	Collectability	Performance	Acceptability	Circumvention
DNA	H	H	H	L	H	L	L
Ear	M	M	H	M	M	H	M
Face	H	L	M	H	L	H	H
Facial thermogram	H	H	L	H	M	H	L
Fingerprint	M	H	H	M	H	M	M
Gait	M	L	L	H	L	H	M
Hand geometry	M	M	M	H	M	M	M
Hand vein	M	M	M	M	M	M	L
Iris	H	H	H	M	H	L	L
Keystroke	L	L	L	M	L	M	M
Odor	H	H	H	L	L	M	L
Palmprint	M	H	H	M	H	M	M
Retina	H	H	M	L	H	L	L
Signature	L	L	L	H	L	H	H
Voice	M	L	L	M	L	H	H

Figure 2.2: Comparison of biometric technologies (High, Medium, Low).

A previous work involving face and iris for identity verification is described in [8]. These two biometric traits are combined using two different strategies of fusion: the computation of either an unweighted or weighted sum and the comparison of the result to a threshold and the treatment of the matching distances of face and iris classifiers as a two-dimensional feature vector and the use of a classifier such as Fisher's discriminant analysis and a neural network with radial basis function (RBFNN) to classify the vector as being genuine or an impostor.

2.3 Face recognition

Face recognition, i.e. determining whose face the face detected in the image is, is a non-intrusive method and also requires minimum cooperation from the subject. The dimensions, proportions and physical attributes of a person's face are unique.

Face recognition is different from face detection since the goal of this last technique is just determining whether a certain picture contains a face (or several) or not. As can be assumed, this task is simpler than recognizing a face, since a general structure of a face can be defined. Template matching, Statistical Pattern Recognition or Haar cascades can be used.

Face recognition can be in a static controlled environment or a dynamic uncontrolled environment. One popular approach to face recognition is based on the location, dimensions and proportions of facial attributes such as eyes, eyebrows, nose, lips, and chin and their spatial relationships. Another approach being widely used is based on the overall analysis of the face image that represents face as a weighted combination of a number of canonical faces.

Face recognition involves two major tasks: face location and face recognition. Face location is determining the location of face in the input image (if there is one). Recognizing the located face means that the face is recognized from a general viewpoint (i.e. from any pose).

Different approaches for face recognition will be now presented.

OpenCV provides three methods of face recognition [9]: Eigenfaces, Fisherfaces and Local Binary Patterns Histograms (LBPH). All three methods perform the recognition by comparing the face to be recognized with some training set of known faces.

2.3.1 Eigenface

The eigenface approach is one of the very popular methods. The eigenface-based recognition method consists of two stages: the training stage and the

operational stage. In the training stage, training set of face images are acquired. The acquired face images are projected into lower dimensional subspace using Principle Component Analysis (PCA). A set of images that best describe the distribution of training images in a lower dimensional face-space (the eigenspace) is computed. Then the training facial images are projected into this eigenspace to generate representation of the training images in the eigenspace. In the operational stage, the input face image is projected into the same eigenspace that the training samples were projected into. Then, recognition can be performed by a classifier operating in the eigenspace.

The Principal Component Analysis (PCA) was independently proposed by Karl Pearson (1901) and Harold Hotelling (1933) to turn a set of possibly correlated variables into a smaller set of uncorrelated variables. The idea is, that a high-dimensional dataset is often described by correlated variables and therefore only a few meaningful dimensions account for most of the information. The PCA method finds the directions with the greatest variance in the data, called principal components.

Algorithmic Description of Eigenfaces method

Let $X = \{ x_1,x_2,...,x_n \}$ be a random vector with observations $x_i \in R^d$. The mean μ is computed:

$$\mu = \frac{1}{n}\sum_{i=1}^{n} x_i. \tag{2.1}$$

And the covariance matrix S:

$$S = \frac{1}{n}\sum_{i=1}^{n}(x_i - \mu)(x_i - \mu)^T. \tag{2.2}$$

Then the eigenvalues λ_i and the eigenvectors v_i of S:

$$Sv_i = \lambda_i v_i, i = 1, 2, ..., n. \tag{2.3}$$

Order the eigenvectors descending by their eigenvalue. The k principal components are the eigenvectors corresponding to the k largest eigenvalues. The k principal components of the observed vector x are then given by:

$$y = W^T(x - \mu), \tag{2.4}$$

where $W = \{ v_1,v_2,...,v_n \}$.

The reconstruction from the PCA basis is given by:

$$x = Wy + \mu. \tag{2.5}$$

The Eigenfaces method then performs face recognition by projecting all training samples into the PCA subspace, projecting the query image into the PCA subspace and finding the nearest neighbour between the projected training images and the projected query image.

To be computationally feasible, it is possible to take the eigenvalue decomposition $S = X^T X$ of size NxN instead:

$$X^T X v_i = \lambda v_i \tag{2.6}$$

and get the original eigenvectors of $S = X X^T$ with a left multiplication of the data matrix:

$$X X^T (X v_i) = \lambda (X v_i). \tag{2.7}$$

The resulting eigenvectors are orthogonal, to get orthonormal eigenvectors they need to be normalized to unit length.

2.3.2 Fisherface

As Eigenfaces, Fisherfaces finds a mathematical description of the most dominant features of the training set as a whole.

The Fisherfaces method uses the Linear Discriminant Analysis (LDA), which performs a class-specific dimensionality reduction and in order to find the combination of features that separates best between classes, it maximizes the ratio of between-classes to within-classes scatter [9].

Algorithmic Description of Fisherfaces method

Let X be a random vector with samples drawn from c classes: $X = \{ X_1, X_2, ..., X_c \}$ and $X_i = \{ x_1, x_2, ..., x_n \}$.

The scatter matrices S_B and S_W are calculated as:

$$S_B = \sum_{i=1}^{c} N_i (\mu_i - \mu)(\mu_i - \mu)^T \tag{2.8}$$

$$S_W = \sum_{i=1}^{c} \sum_{x_j \in X_i} (x_j - \mu_i)(x_j - \mu_i)^T \tag{2.9}$$

where μ is the total mean, calculated as the Eigenfaces method. And μ_i is the mean of the class $i \in 1, ..., c$.

Fisher's classic algorithm then looks for a projection W, that maximizes the class separability criterion:

$$W_{opt} = argmax_W \frac{W^T S_B W}{W^T S_W W}. \tag{2.10}$$

As seen before, a solution for this optimization problem is given by solving the General Eigenvalue Problem:

$$S_B v_i = \lambda_i S_W v_i \tag{2.11}$$

$$S_W^{-1} S_B v_i = \lambda_i v_i. \tag{2.12}$$

To obtain the Fisherfaces, S_W^{-1} needs to be computed; if the total number of samples N is smaller than the dimension of the input data (in patter recognition problem it almost always happens), the scatter matrix S_W becomes singular. A typical solution is the projection of the sample vectors onto the PCA space of r dimensions, with $r \leq rank(S_W)$ and the computation of the Fisherfaces in the PCA space.

2.3.3 Local Binary Patterns Histogram

LBPH analyses each face in the training set separately and independently.

The underlying idea is to describe only local features of an object in the image; the image is not seen as a high-dimensional vector any more and the extracted features have a low-dimensionality implicitly.

The basic concept of Local Binary Patterns is to summarize the local structure in an image by comparing each pixel with its neighbourhood. Taking a pixel as center, and its neighbours are compared against it using a threshold. If the intensity of the center pixel is greater-equal its neighbour, then denote it with 1 and 0 if not. A binary number for each pixel is obtained as result. For example, with 8 surrounding pixels there will be 2^8 possible combinations, called Local Binary Patterns.

Algorithmic Description of LBPH method

A formal description of the LBP operator is:

$$LBP(x_c, y_c) = \sum_{p=0}^{P-1} 2^p s(i_p - i_c) \tag{2.13}$$

with (x_c, y_c) as central pixel with intensity i_c, i_n being the intensity of the the neighbour pixel and s the sign function defined as:

$$s(x) = \begin{cases} 1 \; if \; x \geq 0 \\ 0 \; otherwise \end{cases} \tag{2.14}$$

Since a fixed neighbourhood fails to encode details differing in scale, the operator, which is robust against monotonic gray scale transformations, was

extended (and called *Extended LBP*) to use a variable neighbourhood in order to align an arbitrary number of neighbours on a circle with variable radius, which enables to capture different neighbourhoods [Fig.2.3].

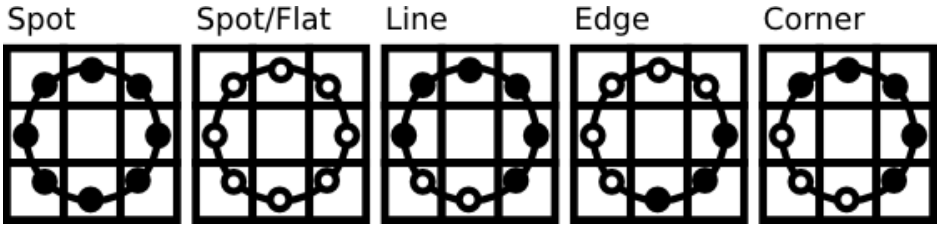

Figure 2.3: LBPH algorithm: possible patterns as neighbourhoods.

For a given point (x_c,y_c) the position of the neighbour (x_p,y_p), $p \in P$ can be calculated by:

$$x_p = x_c + Rcos(\frac{2\pi p}{P}) \tag{2.15}$$

$$y_p = y_c - Rsin(\frac{2\pi p}{P}), \tag{2.16}$$

where R is the radius of the circle and P is the number of sample points. If a points coordinate on the circle does not correspond to image coordinates, the point is interpolated. The OpenCV implementation does a bilinear interpolation:

$$f(x,y) \approx \begin{bmatrix} 1 & -x & x \end{bmatrix} \begin{bmatrix} f(0,0) & f(0,1) \\ f(1,0) & f(1,1) \end{bmatrix} \begin{bmatrix} 1-y & y \end{bmatrix}. \tag{2.17}$$

Thus, this spatial information needs to be incorporated in the face recognition model. It is obtained by dividing the LBP image into m local regions and extracting a histogram from each. The spatially enhanced feature vector is then obtained by concatenating the local histograms, called *Local Binary Patterns Histograms*.

The LBPH method has been chosen for the face recognition phase of the biometric system. The reason why this choice was made is that the LBPH method is robust against scale, translation and rotation in images, thanks to its local description of them. Moreover, the code was available in the documentation of OpenCV and ready to be properly modified and adapted to the scope of the system, as described in the next chapters.

2.3.4 Pyramid Match Kernel

Another interesting approach which can be used for face recognition is The Pyramid Match Kernel [10], based on a fast kernel function which maps un-

ordered feature sets to multi-resolution histograms and computes a weighted histogram intersection in this space.

Each feature set is mapped to a multi-resolution histogram that preserves the individual features' distinctness at the finest level. The histogram pyramids are then compared using a weighted histogram intersection computation, which defines an implicit correspondence based on the finest resolution histogram cell where a matched pair first appears. The comparison of the histograms with a weighted histogram intersection measure is used to approximate the similarity of the best partial matching between the feature sets [Fig.2.4].

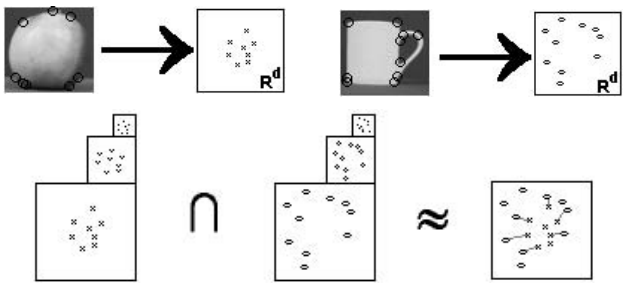

Figure 2.4: Pyramid Match Kernel: intersection of histogram pyramids formed over local features.

Algorithmic Description of Pyramid Match Kernel

Formally, there is an input space X of sets of d-dimensional feature vectors that are bounded by a sphere of diameter D and whose minimum inter-vector distance is $\frac{\sqrt{d}}{2}$. The feature extraction function Ψ is defined as:

$$\Psi(x) = [H_{-1}(x), H_0(x), ..., H_L(x)], \tag{2.18}$$

where $L = \lceil log_2 D \rceil$, $x \in X$, $H_i(x)$ is a histogram vector formed over data x using d-dimensional bins of side length 2^i and $H_i(x)$ has a dimension $r_i =$

$(\frac{D}{2^i\sqrt{d}})^d$. In other words, $\Psi(x)$ is a vector of concatenated histograms where each subsequent component histograms has bins that double in size (in all d dimensions) compared to the previous one. The bins in the finest-level histogram H_1 are small enough that each d-dimensional data point from sets in X falls into its own bin, and then the bin size increases until all data points from sets in X fall into a single bin at level L.

The pyramid match kernel K_Δ measures similarity between point sets based on implicit correspondences found within this multi-resolution histogram space. The similarity between two input sets is defined as the weighted sum of the number of feature matchings found at each level of the pyramid formed by Ψ:

$$K_\Delta(\Psi(y), \Psi(z)) = \sum_{i=0}^{L} w_i N_i, \tag{2.19}$$

where N_i signifies the number of newly matched pairs at level i. A new match is defined as a pair of features that were not in correspondence at any finer resolution level.

The kernel implicitly finds correspondences between point sets, considering two points matched once they fall into the same histogram bin (starting at the finest resolution level where each point is guaranteed to be in its own bin). The matching is equivalent to a hierarchical process: vectors not found to correspond at a high resolution have the opportunity to be matched at lower resolutions. K_Δ's output value reflects the overall similarity of the matching: each newly matched pair at level i contributes a value w_i that is proportional to how similar two points matching at that level must be, as determined by the bin size. Note that the sum in (2.19) starts with index $i = 0$, because the definition of Ψ insures that no points match at level $i = -1$.

To calculate N_i, the kernel makes use of a histogram intersection function I, which measures the "overlap" between two histograms' bins:

$$I(A, B) = \sum_{j=i}^{r} min(A^{(j)}, B^{(j)}), \tag{2.20}$$

where A and B are histograms with R bins and $A^{(j)}$ denotes the count of the j^{th} bin of A.

Histogram intersection effectively counts the number of points in two sets which match at a given quantization level, i.e., fall into the same bin. To calculate the number of newly matched pairs N_i induced at level i, it is sufficient to compute the difference between successive histogram levels' intersections:

$$N_i = I(H_i(y), H_i(z)) - I(H_{i-1}(y), H_{i-1}(z)), \tag{2.21}$$

where H_i refers to the i^{th} component histogram generated by Ψ in (2.18). Note that the kernel is not searching explicitly for similar points - it never computes distances between the vectors in each set. Instead, it simply uses the change in intersection values at each histogram level to count the matches as they occur.

The number of new matches found at each level in the pyramid is weighted according to the size of that histogram's bins: matches made within larger bins are weighted less than those found in smaller bins. Since the largest diagonal of a d-dimensional hypercube bin with sides of length 2^i has length $2^i\sqrt{d}$, the maximal distance between any two points in one bin doubles at each increasingly coarser histogram in the pyramid. Thus, the number of new matches induced at level i is weighted by $\frac{1}{2^i}$ to reflect the (worst-case) similarity of points matched at that level. Intuitively, this means that similarity between vectors (features in y and z) at a finer resolution - where features are most distinct - is rewarded more heavily than similarity between vectors at a coarser level.

Then, the value is normalized by the product of each input's self-similarity to avoid favouring larger input sets.

An example can be seen in [Fig.2.5].

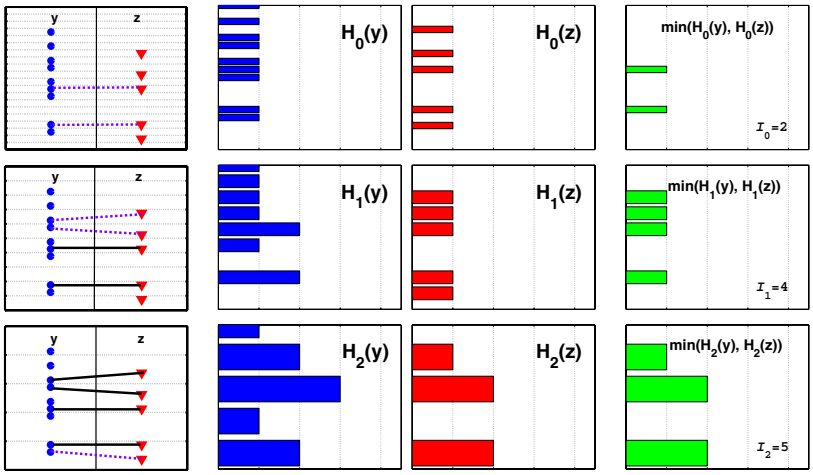

Figure 2.5: Example of Pyramid Match Kernel method.

Two $(1-D)$ feature sets are used to form two histogram pyramids. Each row corresponds to a pyramid level. H_1 is not pictured because no matches are formed at the finest level. In (a), the set y is on the left side and the set z is on the right side. Points are distributed along the vertical axis, and these same points are repeated at each level. Light dotted lines are bin boundaries, bold dashed lines indicate a pair matched at this level, and bold solid lines indicate a match already formed at a finer resolution level. In (b) multi-resolution histograms are shown, with bin counts along the horizontal axis. In (c) the intersection pyramid between the histograms in (b) are shown. K_Δ uses this to measure how many new matches occurred at each level. I_i refers to $I(H_i(y), H_i(z))$. Here, $I_i = 2, 4, 5$ across levels, and therefore the number of new matches found at each level are $N_i = 2, 2, 1$. The sum over N_i, weighted by $w_i = 1, \frac{1}{2}, \frac{1}{4}$, gives the pyramid match similarity.

2.3.5 Other face recognition methods

The Gaussian Mixture Models (GMM)-based human face identification technique built in the Fourier or frequency domain that is robust to illumination changes and does not require illumination normalization, i.e. removal of illumination effects, prior to application, unlike many existing methods. The importance of the Fourier domain phase in human face identification is a well-established fact in signal processing, providing a suitable semi-parametric framework for modelling unknown and complex distributional shapes. Therefore, mixtures can handle situations where a single parametric family fails to provide a satisfactory model for local variations in the observed data, and offer the scope of inference at the same time [11].

The Gaussian Face method, described in [12], addresses the problem of face verification in complex conditions with large variation as pose, illumination, expression and occlusions, relying on a principled multi-task learning approach based on Discriminative Gaussian Process Latent Variable Model, to enrich the diversity of the training data and thus leading to an improvement of the generalization performance of face verification in an unknown target-domain.

It is worth to note that beyond the cited tradition method for face recognition there are also non traditional techniques. Examples are:

- the three-dimensional face recognition, which uses 3D sensors to capture information about the shape of a face and using it to identify distinctive features on the surface of a face, such as the contour of the eye sockets, nose, and chin [13]; it achieve improved accuracy, it is not

affected by changes in lighting, it can identify a face from a range of viewing angles, including a profile view

- the skin texture analysis, which uses the visual details of the skin, as captured in standard digital or scanned images; this technique, called skin texture analysis, turns the unique lines, patterns, and spots apparent in a person's skin into a mathematical space.

- the thermal cameras, where cameras only detect the shape of the head and it will ignore the subject accessories such as glasses, hats, or make up [14].

2.4 Iris recognition

Iris recognition is generally considered to be one of the most effective modalities for biometric identification. The iris is both protected and accessible and can be accessed in a non-contact manner from moderate distances [15]. The visual texture of the iris is formed during fetal development and stabilizes during the first two years of life. Besides, it has many degrees of freedom: the complex iris texture carries very distinctive information useful for personal recognition of high accuracy.

Moreover, it is extremely difficult to surgically tamper the texture of the iris and it is rather easy to detect artificial irises (e.g., designer contact lenses).

Traditional iris acquisition devices are good for capturing high quality iris image, but extremely inflexible and not user-friendly. It requires high level of user cooperation to successfully acquire iris image. In the past last years, progress of innovation in iris acquisition device has made it possible to lessen the requirement of user-cooperation while maintaining high quality iris images. They can be can be categorized as middle distance iris acquisition devices, because they can capture users' iris images even when users stand at a distance of 50-100 cm away from the camera and users are not required to put their heads against a rack. Long distance iris acquisition devices (e.g. high resolution infrared cameras) are able to capture iris images even when they are moving [15].

The process of locating iris region in an image is called iris segmentation. As stated above, iris region is of annular shape. Therefore, it is intuitive to segment iris with two circles: one circle indicates the boundary between iris and pupil, the other indicates the boundary between iris and sclera. The

texture in the region between those two circles is what it is needed for further process.

There exist different algorithms which serve the same goal of iris segmentation. One of the most popular algorithms is proposed by Wildes, another by Daugman.

2.4.1 Wildes

Wildes proposed an algorithm consisting of two main steps [16].

The first step is to apply a gradient-based edge detector on the whole eye image to generate an edge-map, which tells the position where strong edges exist (strong differences in pixel values). Intuitively, those positions are possible candidates of the iris boundaries since the two boundaries of iris, both inner and outer, are the positions where high pixel contrast takes place.

The second step is to find the exact two boundaries since this edge-map. The used method is a voting scheme. Every circle on a 2D plane can be described by three parameters, coordinate in x and y axis and the radius r. Therefore a three-dimensional (3D) space can be constructed, where each dimension represent one parameter. Every positive pixel on the edge-map can vote to the point (x,y,r) in the 3D parameter space as long as this positive pixel is on the perimeter of the circle which is parametrized (x,y,r). Since different circles can pass through the same point, it is possible for 1 pixel on the edge-map to vote to multiple points in the 3D parameter space.

At last, the point which has the highest accumulated votes represents the most likely circle in the original eye image [Fig.2.6].

2.4.2 Daughman

According to John Daugman, to capture the rich details of iris patterns, an imaging system should resolve a minimum of 70 pixels in iris radius, but a resolved iris radius of 100 to 140 pixels is more typical. In his proposed method [15][17], images passing a minimum focus criterion are analysed to find the iris, with precise localization of its boundaries using a coarse-to-fine strategy terminating in single-pixel precision estimates of the center coordinates and radius of both the iris and the pupil. Although the results of the iris search greatly constrain the pupil search, concentricity of these boundaries cannot be assumed. Very often the pupil center is nasal, and inferior, to the iris center. Its radius can range from 0.1 to 0.8 of the iris radius. Thus, all three parameters defining the pupil circle must be estimated separately from those of the iris. A very effective integrodifferential operator for determining these

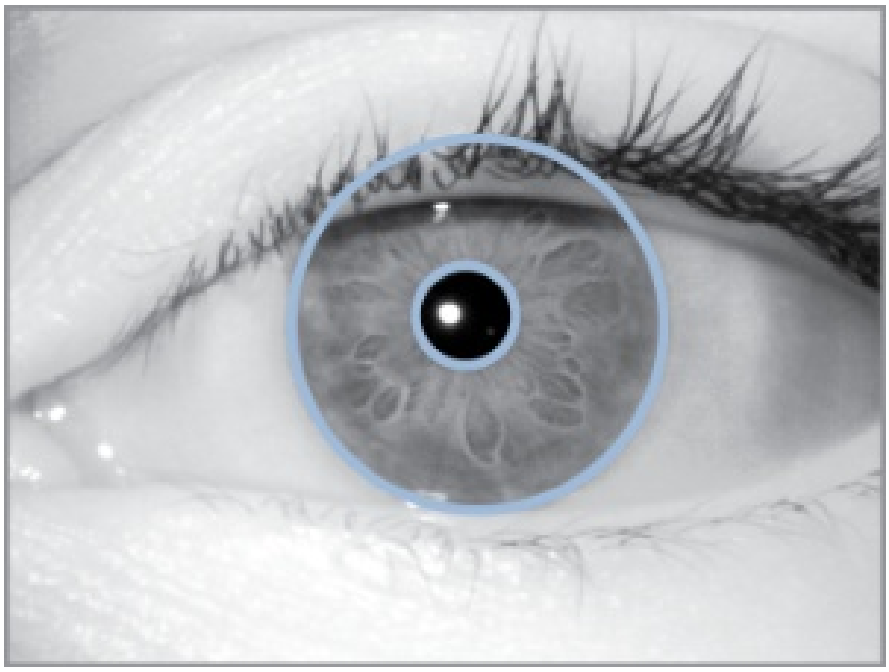

Figure 2.6: Example of iris segmentation.

parameters is:

$$max_{(r,x_0,y_0)} = |G_\sigma(R) * \frac{\partial}{\partial r} \oint_{r,x_0,y_0} \frac{I(x,y)}{2\pi r} ds| \tag{2.22}$$

where $I(x,y)$ is an image containing an eye. The operator searches over the image domain (x,y) for the maximum in the blurred partial derivative with respect to increasing radius r, of the normalized contour integral of $I(x,y)$ along a circular arc ds of radius r and center coordinates (x_0,y_0). The symbol $*$ denotes convolution and $G_\sigma(r)$ is a smoothing function such as a Gaussian of scale σ. The complete operator behaves in effect as a circular edge detector, blurred at a scale set by σ, which searches iteratively for a maximum contour integral derivative with increasing radius at successively finer scales of analysis through the three parameter space of center coordinates and radius (x_0,y_0,r) defining a path of contour integration.

The operator in (2.22) serves to find both the pupil boundary and the outer (limbus) boundary of the iris, although the initial search for the limbus also incorporates evidence of an interior pupil to improve its robustness since the limbic boundary itself usually has extremely soft contrast. Once the coarse-to-fine iterative searches for both these boundaries have reached single pixel precision, then a similar approach to detecting curvilinear edges

is used to localize both the upper and lower eyelid boundaries. The path of contour integration in (2.22) is changed from circular to arcuate, with spline parameters fitted by standard statistical estimation methods to describe optimally the available evidence for each eyelid boundary. The result of all these localization operations is the isolation of iris tissue from other image regions [Fig.2.7].

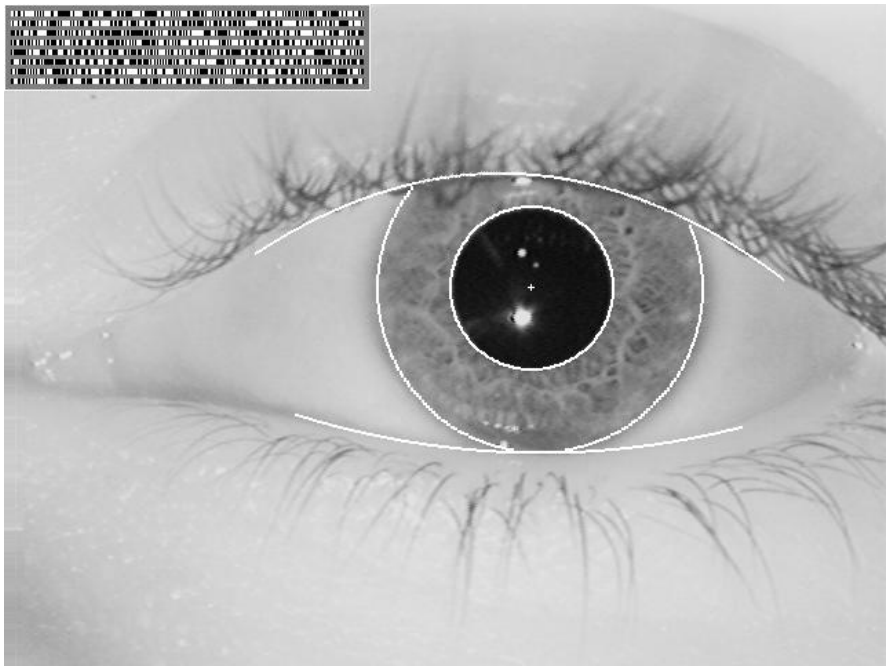

Figure 2.7: Daugman's method: example of iris pattern.

Therefore, Daugman, in his later work [15], proposed a new segmentation method, called *active contour* or *snakes*, that is to use a more flexible model to represent both boundaries: the boundary of the pupil and iris is not bounded circular, but they can be of any shape. Therefore, the discovered boundaries can fit to the real data more closely and bring performance enhancement in the pattern matching phase.

After iris localization, it useful to perform an image coordinate transformation on the iris image. The goal is to transform the annular-shaped iris region into rectangular shape, by mapping pixel values from Cartesian coordinate to polar coordinate. Polar coordinate is parametrized with two parameters: r and θ [Fig.2.8].

A valuable advantage of coordinate transformation can be appreciated in practical situations. The size of the pupil are is not always the same, but it contracts and dilates because of the ambient lighting. Therefore, performing

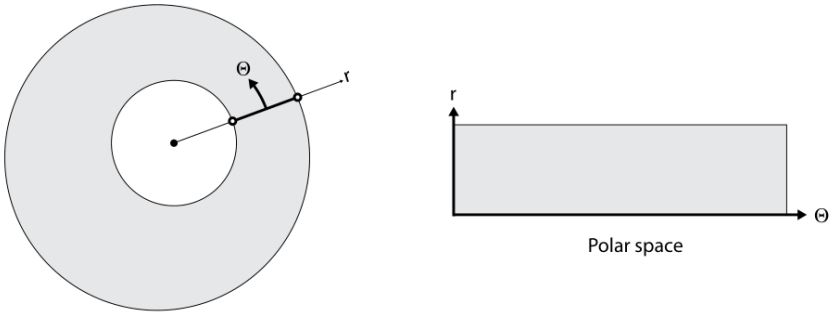

Figure 2.8: Daugman's rubber sheet model.

polar transformation for every iris texture the resulting iris texture map in polar coordinate remains the same. Other sources of inconsistency include: varying imaging distance, rotation of the camera, head tilt and rotation of the eye within the eye socket. The normalisation process will produce iris regions, which have the same constant dimensions, so that two images of the same iris under different conditions will have characteristic features at the same spatial location.

The next step is to perform the iris feature extraction on the iris texture. The goal of feature extraction is to extract the discriminative characteristics of the iris texture and store it in a more compact way so that is more effective to perform patter matching in a later stage.

In the field of image processing and pattern recognition, the feature extraction is accomplished by applying filters on the input image. In literature, the filter which have been used extensively in iris recognition are Gabor filters, proposed precisely by Daugman. Gabor filters can be seen as complex sinusoids modulated by Gaussian envelope. They can be expressed as the following equation:

$$\Psi(x,y) = A \cdot e^{[-\frac{1}{2}(\frac{x^2}{\sigma_x^2}+\frac{y^2}{\sigma_y^2})-j\omega(cos\theta)x-j\omega(sin\theta)y]} \tag{2.23}$$

where, by convention, the standard deviations of the envelope are inversely proportional to frequency ω:

$$\sigma_x = k_x \frac{2\pi}{\omega} \tag{2.24}$$

$$\sigma_y = k_y \frac{2\pi}{\omega}. \tag{2.25}$$

Gabor filters of different sizes, orientation or frequency can be generated by substituting different parameters in (2.23); they can capture different texture details in iris texture.

A two-dimensional (2D) complex-valued plane is obtained after applying each Gabor filter on iris texture. Daugman proposed a phase-quadrant scheme to further quantize the 2D complex-valued plane. Phase information on a complex-valued filter response plane is important to have much more discriminative information in biometric recognition. Therefore, performing phase quantization would discard useless information and make feature representation more compact and easy to handle.

The phase-quadrant quantization scheme amounts to a patch-wise phase quantization of the iris pattern, by identifying in which quadrant of the complex plane each resultant phasor lies. For every location on the complex plane, the sign of real and imaginary part jointly defines in which quadrant the phasor lies. Two bitecodes are used to encode the four possible quadrants [Fig.2.9].

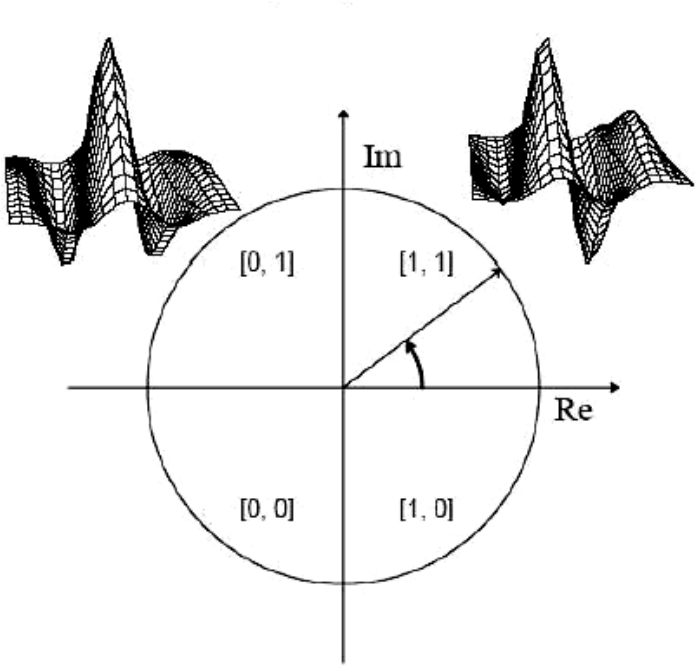

Figure 2.9: Daugman's method: demodulation by complex-valued wavelets.

The phase-quadrant demodulation process is repeated across all the plane of the response of the filters, with many different wavelet sizes, frequencies and orientation to extract totally 2048 bits.

An important fact to note is that, often, iris texture is occluded by other objects (e.g. eye lashes, eye lids, specular reflection from eye gasses). Not every point on the iris texture map is useful for pattern matching. As a consequence, it is necessary to compute an iris mask, which is exactly the same size as iris texture, to indicate which part of the map is really iris texture and which is not. This mask will be used in the stage of pattern matching.

Pattern Matching

The key to iris recognition is the failure of a test of statistical independence, which involves so many degrees-of-freedom that this test is virtually guaranteed to be passed whenever the phase codes for two different eyes are compared, but to be uniquely failed when any eye's phase code is compared with another version of itself.

The test of statistical independence is implemented by the simple Boolean Exclusive-OR operator (XOR) applied to the 2048 bit phase vectors that encode any two iris patterns, masked (AND'ed) by both of their corresponding mask bit vectors to prevent non-iris artefacts from influencing iris comparisons. The XOR operator $\otimes$ detects disagreement between any corresponding pair of bits, while the AND operator $\cap$ ensures that the compared bits are both deemed to have been uncorrupted by eyelashes, eyelids, specular reflections, or other noise.

The norms of the resultant bit vector and of the AND'ed mask vectors are then measured in order to compute a fractional Hamming Distance (HD) as the measure of the dissimilarity between any two irises, whose two phase code bit-vectors are denoted $codeA, codeB$ and whose mask bit vectors are denoted $maskA, maskB$:

$$HD = \frac{\|(codeA \otimes codeB) \cap maskA \cap maskB\|}{\|maskA \cap maskB\|} \tag{2.26}$$

The denominator tallies the total number of phase bits that mattered in iris comparisons after artefacts such as eyelashes and specular reflections were discounted, so the resulting HD is a fractional measure of dissimilarity; 0 would represent a perfect match. Because any given bit in the phase code for an iris is equally likely to be 1 or 0, and different irises are uncorrelated, the expected proportion of agreeing bits between the codes for two different irises is $HD = 0.500$.

Chapter 3

System Design

In this section there is a complete description of the project, including its context and its requirements analysis.

The goal is the design of an adaptive multi-modal biometric system for indoor selective access control.

3.1 System requirements

This system should operate within a scope of a medium/big company, which needs to keep some areas restricted to particular employees, depending on their roles or responsibilities. Moreover, there could be different levels of security required throughout the structure of the company; everything, as mentioned earlier, as expressed in the Service Level Agreement at a design phase.

Therefore, proper sensors should be placed in those access points which need control and in every room or structure in which only authorized users are allowed.

The system is intended to handle several different situations:

- different types of restricted area have to be accessed by different types of user, depending on their role
- company employees should be given an easy way to access, since they have to do it at least twice a day (they are enrolled in a special database)
- partners, suppliers or auditors, provided with their own company device (thus known by the company), are submitted to an indirect authentication (e.g. having their template stored on their device and sending it to the company for the matching)

- since the company is open to the public, clients are supposed to access only the reception area

- some employee is allowed to bring someone else in with them (the system is supposed to know who and where)

- the system works autonomously, but proper human intervention is always possible

In order to satisfy all of these requirements a multi-modal biometric system, which operates both in verification and identification mode, has been chosen as the best solution.

Furthermore, the system is also required to guarantee some requirements of the SLA concerning specific parameters. In particular, the set FAR and FRR values can be different depending on the various areas or rooms within the company. For example a rather high False Acceptance Rate could be reasonable at the very entrance point (one or more) of the whole structure, because it would be just the first control point, protecting a general area thus far and followed by many others; then, a higher level of accuracy in terms of FAR, which is supposed to be now very slow, could be asked for specific restricted buildings or rooms.

3.2 Solution description

The multi-modal biometric system relies on two different modules: the module for face identification and the module for iris verification. The fusion methodology adopted at decision level follows the *or rule*, i.e. it is sufficient that only a biometric trait is recognized as genuine to lead in a positive final decision. This serial matching approach gives the possibility of not acquiring all the traits; for example, only face recognition is computed if the information collected at the first module is enough to determine if a user is genuine or an impostor.

As seen in Section 1.2, in identification the user attempts to positively identify himself to the system without explicitly claiming an identity. The system determines the identity of the user from a known set of identities.

Formally, the problem of identification can be stated as follows: given a query feature set X_Q, it is necessary to decide the identity I of the user, where $I \in \{ I_1, I_2, ..., I_n, I_{N+1} \}$. $I_1, I_2, ..., I_n$ correspond to the identities of the N users enrolled in the system and I_{N+1} indicates the case where no suitable identity can be determined for the given query. If X_{I_n} is the stored template

corresponding to identity I_n and S_n is the match (similarity) score between X_Q and X_{I_n}, for $n = 1, 2, ..., N$, the decision rule for identification is:

$$X_Q \in \begin{cases} I_{n_0},\ if\ n_0 = argmax_n\ S_n\ and\ S_{n_0} \geq \eta, \\ I_{N+1},\ otherwise \end{cases} \tag{3.1}$$

where η is a pre-defined threshold.

In the proposed solution, as the result of a session of identification, the system gives in output the score of the confidence specifying the degree of trust that there is in the identification of the acquired person with an enrolled person in the database.

In verification, the user claims an identity and the system verifies whether the claim is genuine. In this scenario, the query is compared only to the template corresponding to the claimed identity. If the user's input and the template of the claimed identity have a high degree of similarity, then the claim is accepted as *genuine*. Otherwise, the claim is rejected and the user is considered an *impostor*. Formally, verification can be posed as the following two-category classification problem: given a claimed identity I and a query feature set X_Q, it is necessary to decide if (I, X_Q) belongs to *genuine* or *impostor*. Let X_I be the stored template corresponding to identity I. Typically, X_Q is compared with X_I and a *match score* S, which measures the similarity between X_Q and X_I, is computed. The decision rule is given by:

$$(I, X_Q) \in \begin{cases} genuine,\ if\ S < \eta, \\ impostor,\ if\ S \geq \eta, \end{cases} \tag{3.2}$$

where η is a pre-defined threshold. In this formulation, the match score S is assumed to measure the dissimilarity between X_Q and X_I, i.e., a large score indicates a poor match.

The designed system works in the same way, except the fact that there is one (or two) region of uncertainty, inserted to better handle every possible situation.

Generally, the presence of large number of identities in the database makes the identification task significantly more challenging than verification.

In multi-modal biometric system it is worth to couple non invasive easy-to-use biometrics at lower accuracy together with more robust but expensive parameters, which can intervene any time a higher degree of confidence is required.

Face and iris recognitions have been chosen for the proposed system because of their counterbalancing characteristics.

3.2.1 Face

This identifier has been chosen because face recognition is a non-intrusive method of authentication. Moreover, face matching is typically fast, even if not very accurate.

The application of facial recognition is expected to happen in a dynamic but controlled environment in which a person who wants to be authenticated has to walk along a hallway following an ordered queue so that the sensor can easily detect whether a face is present in the acquired image and locate it. A second (third, ..., k-th) acquisition is potentially possible, in case the first is not precise enough, and the user is required to stand still for a few seconds, while their face is tried to be recognized again.

The chosen methodology is the Local Binary Patterns (LBP). As explained in the previous section (2.3.3), it is based on the extraction of local features from images; the basic idea is to summarize the local structure in an image by comparing each pixel with its neighbourhood. Then the LBP image is divided into local regions and a histogram is extracted from each. Finally, the spatially enhanced feature vector is then obtained by concatenating the local histograms, called Local Binary Patterns Histograms (LBPH) [9].

3.2.2 Iris

The reason why iris has been chosen as the second identifier is that its complex texture carries very distinctive information useful for personal recognition of high accuracy; however, iris matching is slower than face matching and it is also more invasive, i.e. it requires the user's collaboration. The recognition process needs a proper camera to acquire the iris image, while the user stands still and close enough (see 2.4).

For iris recognition the Daughman's method has been chosen. As seen in the previous section 2.4.2, it consists of several phases: segmentation, i.e. the location of the iris in the eye image; normalization, with the Daugman's Rubber Sheet Model; encoding, with Log-Gabor wavelet; matching, using the Hamming Distance [17][18].

3.3 System architecture

The system consists of several different modules, each of them providing its own functionality [2]. There are two *sensor modules*, for either face and iris acquisition, which capture the biometric data. In the *feature extraction*

modules the acquired data is processed to extract a set of salient and discriminatory features. In the *matcher modules* the extracted features are compared against the stored templates, providing a matching score. These last modules encapsulate the *decision making modules*, which operate in verification or identification mode. Moreover, there is the *system database module*, which stores the biometric templates of the enrolled users [Fig.3.1].

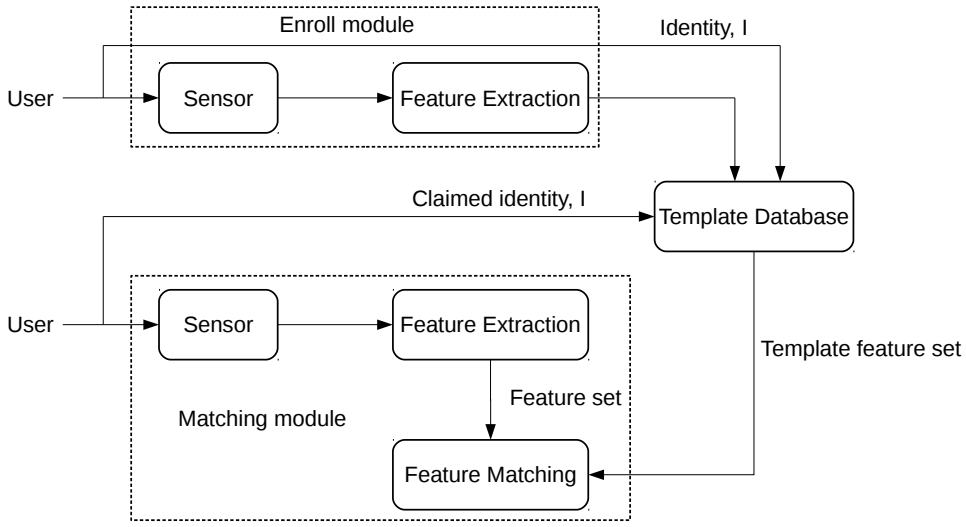

Figure 3.1: General biometric system

The system is intended to be adaptive; it means that it has not a fixed and predefined behaviour, but its working flow depends on the response of the actual recognition process.

Each decision module has three possible different outputs depending on predefined thresholds t_i, with $i = 1, 2, 3$, both for the verification and the identification mode [Fig.3.2]; the output can be:

- *YES*, if the user is recognized as one of the enrolled user or their identity has been verified
- *NO*, if the user is rejected as if they were impostors
- *MAYBE(?)*, if the decision module is not able to make a decision with a sufficient degree of confidence

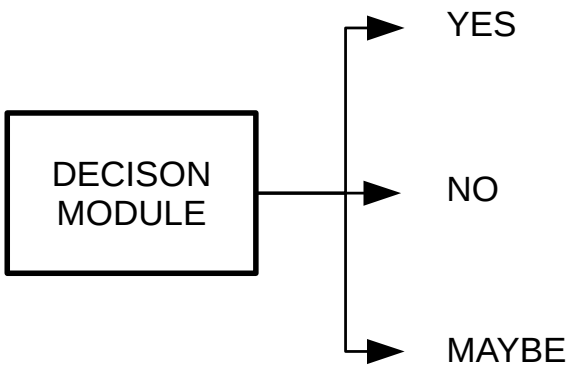

Figure 3.2: Decision making module

Actually, the third possibility (?) consists of two different outputs if that indecision comes from the first attempt of face identification; as a matter of fact, one output leads to a second (or k) attempt of face identification in a more controlled way, the other output leads directly to iris verification [Fig.3.3].

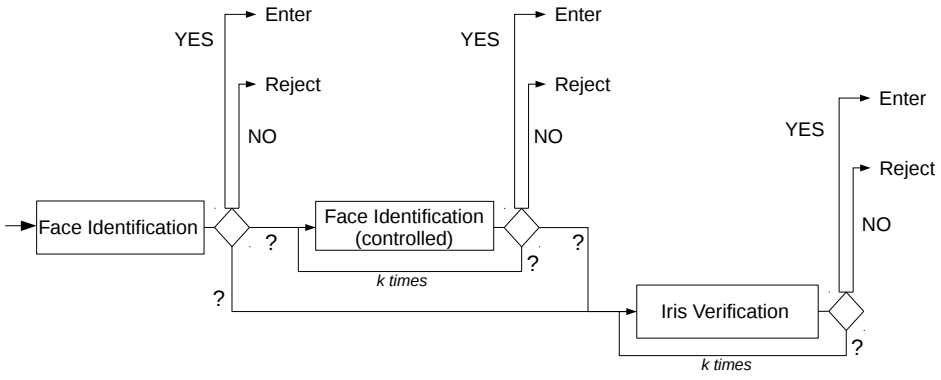

Figure 3.3: Possible System Flows

This choice is based on a special threshold, t_3, set in the middle of the *MAYBE* region [Fig.3.4]. It means that if a user obtains an output score that is similar to a genuine user's score (but not enough), they can try face identification again, because it is more fast and less invasive. Conversely, if the obtained score is nearer to the *NO* region, it is better in terms of

accuracy if the user is asked to submit to iris verification, thus requiring more collaboration.

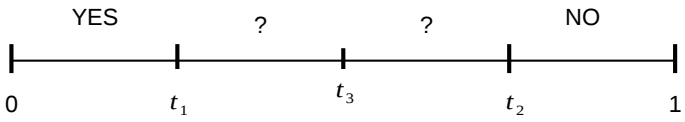

Figure 3.4: Face identification thresholds

Therefore, there are several different ways of working. For example, a user can be immediately identified and authenticated using their face as they approach the entrance; or they can be asked to repeat the face acquisition in a more controlled way; they can be asked to get closer and the authentication mode can be switched to the verification mode using the iris; again, this last phase can be repeated.

Formally, the system model consists of some rules, on which the decision making modules are based:

- require a controlled face identification, if MAYBE (between t_1 and t_3) is the result of the first face identification
- require a controlled face identification, if MAYBE (between t_1 and t_3) is the result of the following attempts of controlled face identification (max k times)
- require iris verification, if MAYBE (between t_3 and t_2) is the result of the first face identification
- require iris verification, if MAYBE (between t_3 and t_2) is the result of any of the controlled face identifications
- require iris verification, if (?) is obtained as the result of the previous iris verifications (max k times)
- ask for human intervention, if NO is the result
- ask for human intervention, if the number of k repetitions has been reached

Obviously, the number of repetitions k can be arbitrarily set by the programmer.

Finally, human intervention is always possible; so, if a genuine user is rejected or receives a number of k repetitions, they still have the possibility to ask for help to a human operator.

Chapter 4

System Implementation

4.1 Databases

Two free available databases have been exploited.

The database of faces AT&T Facedatabase from the AT&T Laboratories of Cambridge [19], containing 10 different images of each of 40 distinct subjects; for some subjects, the images were taken at different times, varying the lighting, facial expressions (open/closed eyes, smiling/not smiling) and facial details (glasses/no glasses). All the images were taken against a dark homogeneous background with the subjects in an upright, frontal position (with tolerance for some side movement). The 40 subjects of this database can be seen in [Fig.4.1].

The files are in PGM format. The size of each image is 92x112 pixels, with 256 grey levels per pixel. The images are organised in 40 directories (one for each subject), which have names of the form sX, where X indicates the subject number (between 1 and 40). In each of these directories, there are ten different images of that subject, which have names of the form Y.pgm, where Y is the image number for that subject (between 1 and 10).

In [Fig.4.2] there are some face images taken from the directory of the subject number 1, $s1$; it is possible to see the different facial expressions that the users has, together with the different inclinations of his head.

The AT&T Face Database is labelled as a fairly easy database; in the next chapter, *Testing and Results*, the results obtained in the testing phase with this database will be presented.

The database of irises UBIRIS [20], from which the first 40 subjects (of 241) have been selected, each having 5 different images. Examples of these images can be find in [Fig.4.3].

It is worth to note that the performance and reliability of all biometric

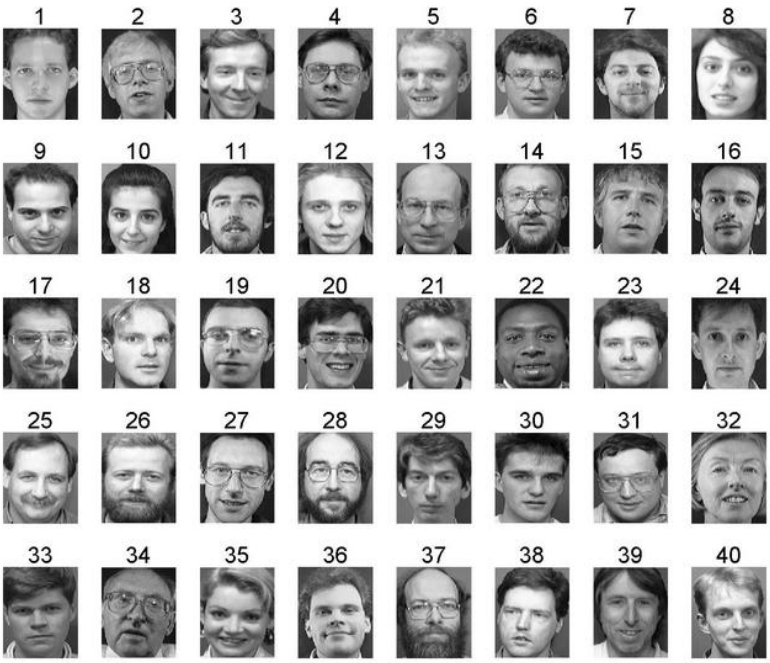

Figure 4.1: AT&T Face Database: 40 users

Figure 4.2: AT&T Face Database: example of a user's face images

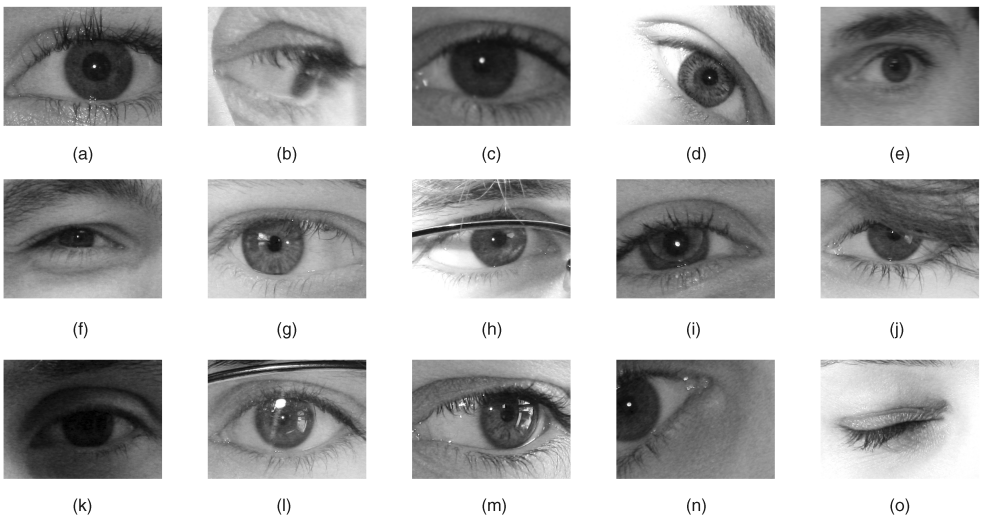

Figure 4.3: Ubiris Database: example of a user's iris images

recognition systems depend crucially on the quality of the enrolment data [15].

UBIRIS is a noisy database, it provides images with different types of noise, simulating image captured without or with minimal collaboration from the subjects, pretending to become an effective resource for the evaluation and development of robust iris identification methodologies [20]. A Nikon E5700 camera has been used, with software version E5700v1.0, 71 mm focal length, 4.2 F-Number, 1/30 sec. exposure time, RGB color representation and ISO-200 ISO speed. Images dimensions were 2560x1704 pixels with 300 dpi horizontal and vertical resolution and 24 bit depth. They were saved in JPEG format with lossless compression.

4.2 Code

For the implementation of the described system two separate source codes have been adopted. The LPPH algorithm used is the face recognition algorithm from OpenCV written in C++. This code has been chosen because it was ready to use and easily adjustable. Indeed, it has been modified as necessary.

The code gets the path to a CSV file where the users' images are listed (their path, folder, file name and label) and creates two vectors: a vector of *Mat* to store the images (their paths) and a vector of *int* to store the corresponding labels. Then it gets the last image from the database, removing

it from the images vector and its label from the labels vector. This step is done so that the training data and the test data do not overlap. After, a LBPH model for face recognition is created and trained with the images and labels read from the CSV file (except the last line). The following lines of code are used to predict the label of that given test image. In order to get as single output just the confidence value, i.e. the degree of certainty the model has on his prediction, the lines giving in output the actual and the predicted label of the image have been commented.

Moreover, a method for the normalization of the values has been added. It takes as input the values of confidence c (they range from ($min = 56.31$ to $max = 103.96$) and it returns a normalized, $(c - min)/(max - min)$, and rounded up to the next integer confidence value.

The core code used for the iris verification is written in Java and has been developed by Simionato [21]. This code is the implementation of the Daughman's method, except the segmentation phase, inspired by the Wildes' method [16]. It relies on several different Java classes, each of them conducting a precise task. For reasons of consistency, the starting point class is called from a C++ file and they are connected by Java Native Interface (JNI) functions [22]. This class calls other classes and functions that are responsible for creating, if possible, an iris code (the iris template and the iris mask) out of a passed iris image and for performing the Hamming Distance between two specified irises.

Moreover, there is a check function which gives the possibility to verify if a user has enough good images stored in the database, by using a special counter. In a real context, this function gives the user the possibility to save their time, since when the user's iris is acquired, the image is immediately checked and if it not good enough to be segmented or encoded, the user is asked to submit to the iris acquisition again, without waiting for an invalid comparison.

The JNI connection, which has to be created and destroyed runtime, is used to exchange useful parameters between the two programs. The C++ program pass the path to the database, the users and the images to match to the Java program; vice versa, the Java program returns the a single value, that is the computed matching score.

These two sections have been combined using scripts which manage the working flow. The scripts consist of a rigid structure of $if - then - else$ statements.

A script is used to handle the face identification phase. Depending on the output value, it decides whether the user can go in or if they are rejected, or they can be given a second (ore more) possibility of authentication. Another

script is used to take care of iris verification. In this phase, too, there are three possible outputs. There is possibly another script between these two, which has been added later and is responsible for a more controlled way of face identification.

The different conducted tests will be explained in the next chapter.

Chapter 5

Test and Results

The face identification tests and the iris tests will be here described and their results commented. In the testing phase they have been conducted separately, since no proof of correlation exists between the face and the iris parameter of a person, so the tests ad the outcomes are completely independent in this sense.

5.1 Face Identification Testing

For the face identification phase, the database has been divided in train set and test set. Each of the 40 users have their best 3 of 10 face images in the train set (used to train the LBPH model) and the remaining 7 in the test set, thus composed of 280 images.

The first test has been conducted using 120 training images and 280 testing images (all together). So this is a two-script model in which the first script handles face identification and the second script, if necessary, handles iris verification. The three outcomes, as shown in [Fig.5.1], are:

- 176 of 280 users are correctly allowed to enter directly (confidence under the threshold t_1)
- 9 of 280 users are rejected (confidence over the threshold t_2)
- 95 of 280 users are in the middle MAYBE region.

Therefore, there is a percentage of True Positive (TP) of 62.86; a False Negative (FN) value of 3.21%, thus meeting the False Rejection Rate (FRR) of about the 3%; and the remaining 33.93% belonging to the MAYBE region.

This last set of users can be required to submit to iris verification or it can be split in two sets: two third of them (63 users, 22.5%), between t_1 and t_3

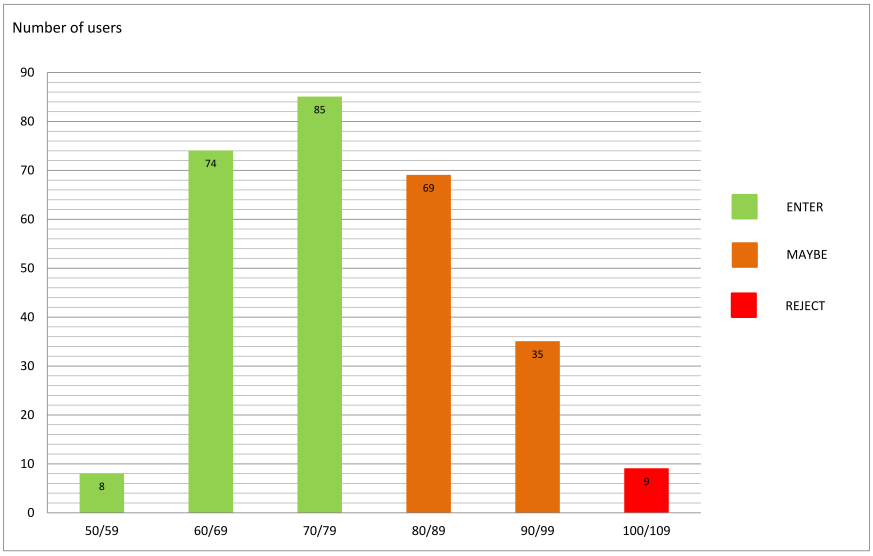

Figure 5.1: Face identification test (before normalization).

(threshold fixed in the exact middle of the other two thresholds [Fig. 3.3]), can be given the possibility to try face identification again; one third (32 users, 11.43%), between t_3 and t_2, have to be submitted to iris verification [Fig.5.2].

This last idea leaded to a new possible branch for the working flow: the addition of a script managing a controlled face identification and the division of the test set of 280 images. A test set $T1$ is intended to be dedicated to the first face identification attempt. $T1$ consists of 160 images, the 4 worst images for every user in terms of confidence result. This test set simulate a real first attempt of identification, which happens while a user is walking towards the entrance and they are not paying much attention on this process. The second test set $T2$, consisting of the remaining 120 images (the 3 best images of each user), is used for the controlled face identification (possibly more than once, up to k times). In this case the user knows that their first attempt is failed and they are trying to be authenticated again in a more collaborative way (e.g. standing still and keeping their head straight).

The results for the four possible outcomes of the first attempt on $T1$:

- 86 of 160 users enter directly, corresponding to the 53.75%

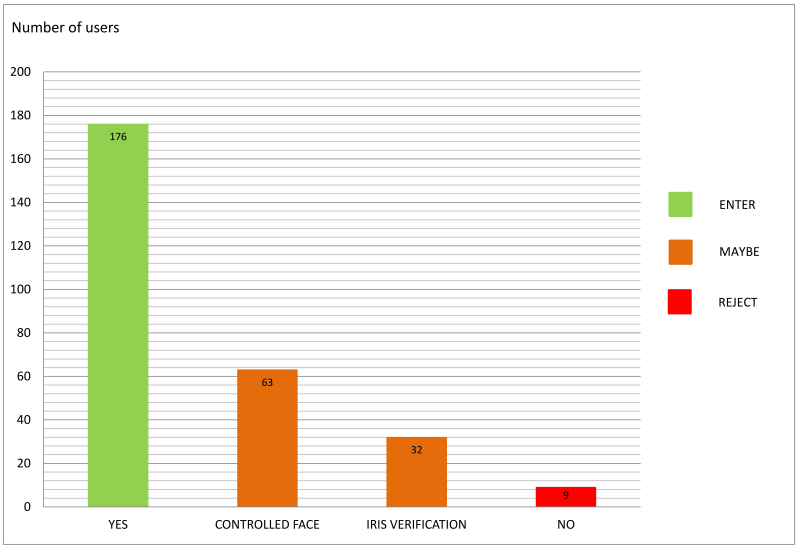

Figure 5.2: Normalized Face Identification test on the complete test set (280 images).

- 9 of 160 users are wrongly rejected, 5.63%
- 42 of 160 user, 26.25%, are allowed to repeat face identification (controlled way)
- 23 of 160 users, 14.37%, are asked to submit to iris verification

In this test the thresholds have been kept fixed to the previous values. This provides a FRR of 5.63%, which is too high assuming a FRR of about 3% agreed in the SLA. So the threshold t_2 has been moved in order to reach a lower FRR. As a result, 5 users instead of 9 are rejected and 4 users are asked to submit to the controlled face identification (they become 46, 28.75%) [Fig.5.3].

The results regarding the test set $T2$ can be seen in [Fig.5.4]. Numerically, they are:

- 98 of 120 users enter, 81.7%
- 2 out of 120 users are rejected, 1.6%
- 20 out of 120 users, 16.7%, are required to submit to iris verification

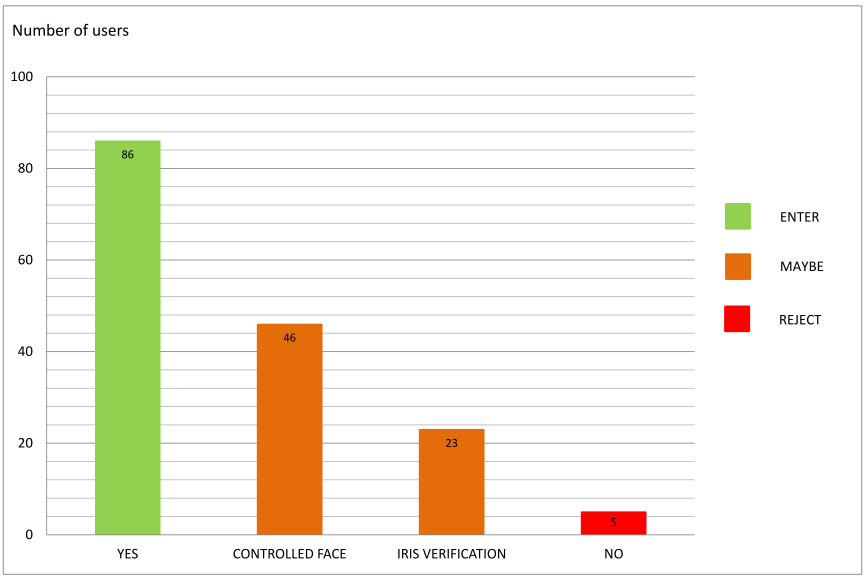

Figure 5.3: First face identification on $T1$.

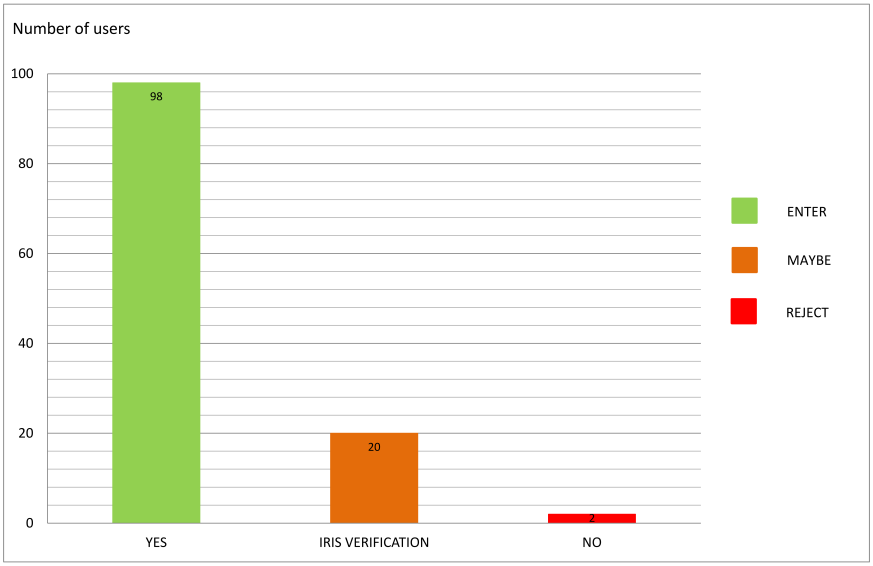

Figure 5.4: Controlled face identification on $T2$.

It is possible to add a further branch, as shown in Fig. 3.2, which gives the opportunity to repeat the controlled face identification many times (maximum k). Nevertheless, it is reasonable to decide that after two attempts (one not controlled and one controlled) of face identification it is safer to require more collaboration to the user in a iris verification process.

5.2 Iris Verification Testing

For the iris verification phase, only two thresholds are sufficient, because the scores obtained by two iris images belonging to the same user and two iris images belonging to different users are far enough [Fig.5.5].

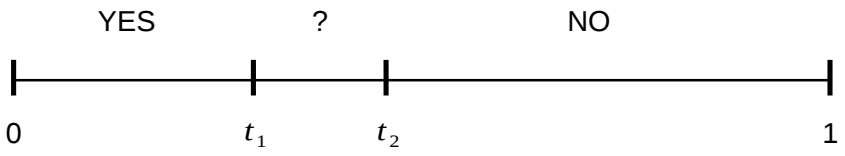

Figure 5.5: Iris verification thresholds.

The thresholds t_1 and t_2 (0.35 and 0.4, respectively) are set in order to distinctly separate the scores resulting from the two types of match: the matches between the irises of two different users lays all over the threshold t_2, so there is no possibility of error in this kind of comparison and the False Acceptance Rate is 0; the valid matches between the irises of the same user lays under the threshold t_1. Just about the 5% of the all comparison between two images of the same user falls into the middle region and thus the verification needs to be repeated.

The tests have been conducted between the different iris images (enumerated from 1 to 5) of the same user for each of the 40 users. In the first test, the first image or every user has been compared with the other four images, thus obtaining 160 comparisons [Fig.5.6], with:

- 118 of 160 positive results, the user is allowed to go in
- 33 of 160 invalid comparisons because at least one of the two images is not good

- 9 of 160 has MAYBE as answer and iris verification has to be repeated

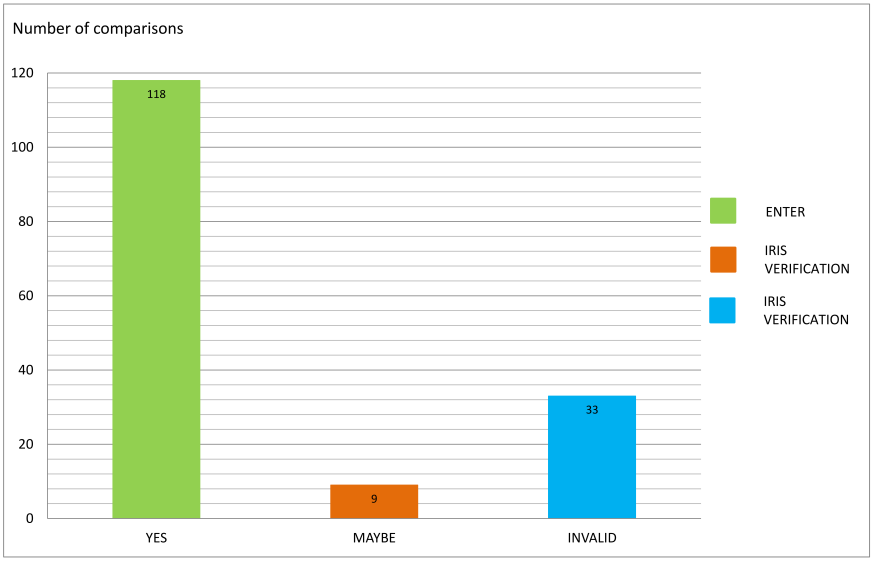

Figure 5.6: First iris verification test.

The second test was between the second image of every user and the other three images, thus there were 120 comparisons. The results are [Fig.5.7]:

- 92 of 120 positive results
- 25 of 120 invalid comparisons due to bad images
- 3 of 120 MAYBE

The third test was between the third image and the last two images, resulting 80 comparisons [Fig.5.8], of which:

- 52 of 80 positive results
- 21 of 80 invalid comparisons
- 7 of 80 MAYBE

The fourth test was between the last two images of every user, so 40 comparisons, and showed [Fig.5.9]:

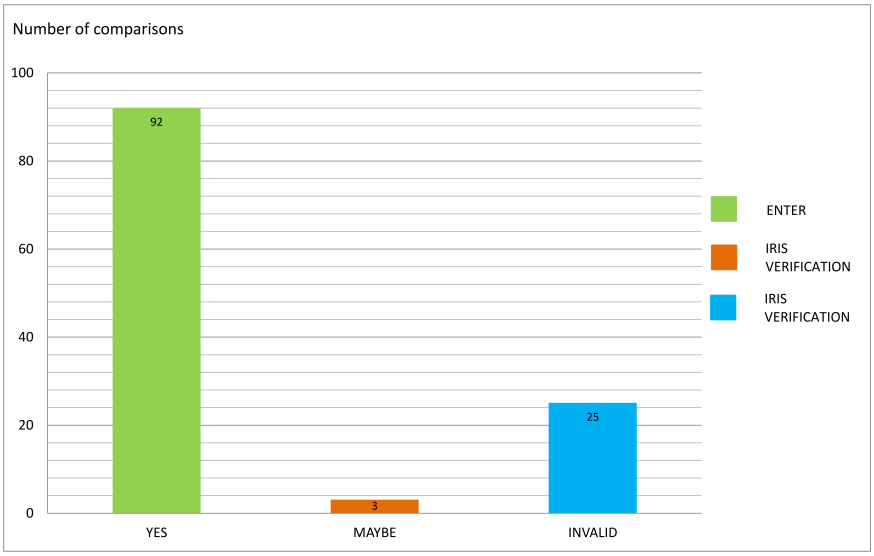

Figure 5.7: Second iris verification test.

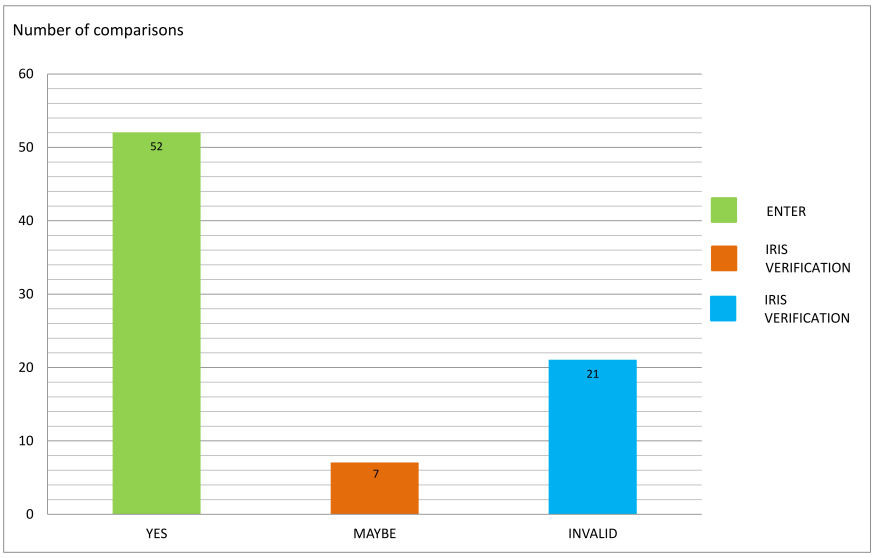

Figure 5.8: Third iris verification test.

- 28 of 40 positive results
- 11 invalid comparisons
- 1 of 40 MAYBE

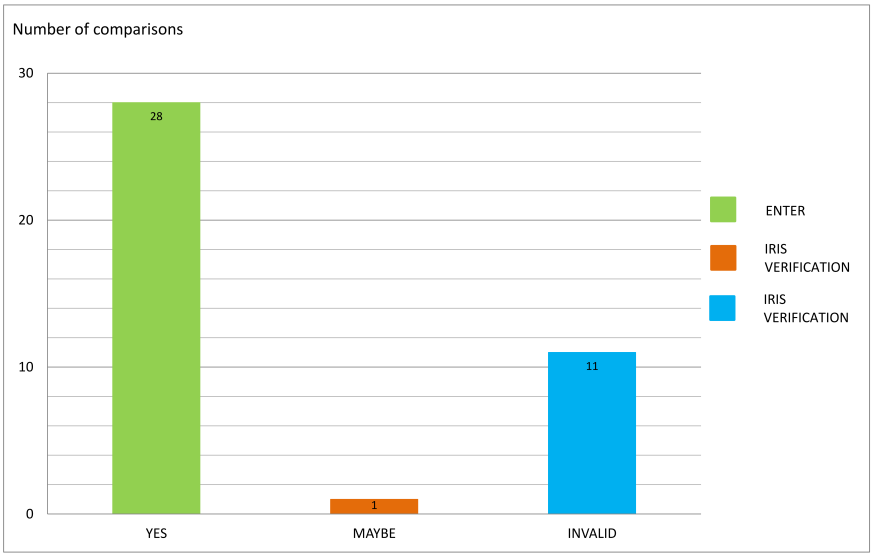

Figure 5.9: Fourth iris verification test.

A noticeable occurrence is that there are a lot of invalid comparisons, i.e. 90 of 400 (which is 160+120+80+40), corresponding to a very high percentage (22.5%) of tests which have to be conducted again, due to the fact that the images utilized are not good. The code used to perform the verification is not capable of extracting useful information from those noisy images, as they cannot be properly segmented or encoded.

The 5% of the comparisons (20 of 400) that lay in the MAYBE region leads to a repetition of the iris verification. Actually, the invalid comparison occurrence has the same result, since those matching needs to be repeated too.

290 of 400 (72.5%) comparisons have a positive result, i.e. the user is allowed to enter without any further verification [Fig.5.10].

Because of the noisy iris image database, some steps of pre-processing could be able to identify those users whose iris images are all unsuitable for

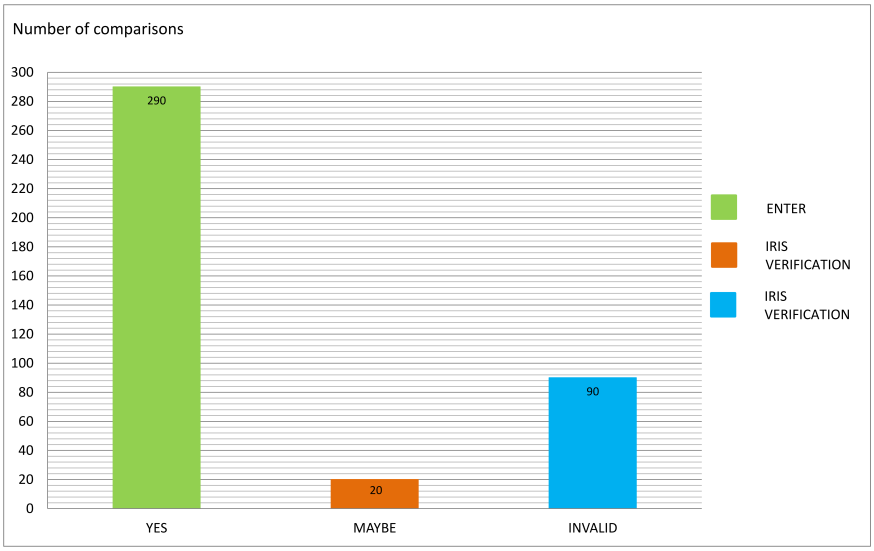

Figure 5.10: Total result of the iris verification test.

the task and thus point them out in order to avoid their usage. Therefore the code has been modified, adding the checking method described in the Section 4.2. The counter is intended to keep track of the number of suitable images of a user and it is checked every time the checking method is called because a verification session is happening.

The script receive as result:

- 0 if the user has to be changed because there is no good images
- 1 if the passed image for the acquired user (the first specified) is not good
- 2 if the passed image for the declared user (the second specified) is not good

This addition is useful in order to save time while testing but also in the real specified context, since when a user submits to iris verification is outright advised if his acquired image is not good enough to be segmented or encoded and so he can repeat the process right away.

In order to achieve an even better degree of security, the system has been provided the possibility of modifying the iris verification phase, so that a

user's acquired image is set against more than one user's remaining images (e.g. 2 or 3 images) and the resulting decision is a combination of the single comparisons. A number of 20 images leads to the MAYBE region, so every other image (even good) associated to them is forced to the MAYBE result and thus to request of iris verification repetition, if the fusion rule is an AND rule and more strictness is required. Conversely, it can be an OR, if there are temporal constraints, and so the system allows faster but less reliable verification.

Chapter 6

Conclusions

This thesis shows the design of an adaptive multi-modal biometric system which consolidates face identification and iris verification, operating in a serial mode.

This work has given the possibility to understand some practical issues and to learn how to face them.

First of all, the importance of a good database, both in terms of size and quality. The size is an important factor, since two quite small databases have been used, knowing that the bigger the database the lower the accuracy, so that there is a decrease in terms of performance. The quality of the stored data, images in this case, significantly affects the performance and the reliability of a biometric recognition system, as it can be noticed thanks to the iris verification testing part.

A good aspect is the adaptive approach wanted for this system. The possible working flows are many, and they represent the fact that the evolution of the process depends on the actual response of the variables involved; in this case, the actual response of the decision making modules.

Furthermore, the tests conducted confirm the advantages of using two different kinds of biometric parameters that differ on invasiveness and accuracy. Biometrics can be really useful and reliable for authentication processes if they are properly chosen, since their usefulness in the employment depends on the desired application and the requirements stated in the SLA.

As regards the future developments, there are several ideas. First of all, tests can be conducted with larger databases, in order to test the scalability of the two recognition processes (especially the identification process).

Then, there is the possibility of the addition of other biometric traits. For example, a third biometric trait can be an alternative solution to the iris parameter; after the first face identification and the second controlled face

identification, a user can be asked if they prefer to try iris verification, as it is now, or, e.g. fingerprint, palm or voice verification.

Another possible addition is that of other recognition algorithms, both for face identification and for iris verification. These algorithms can differ from each other in terms of performances, such as accuracy or computation velocity.

Lastly, if the goal is the design and the development of a complete security system, a presence monitoring section can be added to the proposed solution. The actual presence of users, e.g. employees, can be verified monitoring their behavioural characteristics in a room or within a building. A behavioural parameter can be the gait, i.e. the way of walking of an individual.

Bibliography

[1] https://technet.microsoft.com/en-us/library/cc512578.aspx

[2] Anil K. Jain, Arun Ross, and Salil Prabhakar, "An Introduction to Biometric Recognition", IEEE transaction on circuits and systems for video technology, VOL. 14, NO. 1, JANUARY 2004.

[3] SYRIS Technology Corp, "Technical Document About FAR, FRR and EER", 2004.

[4] Harbi AlMahafzah Ma'en Zaid AlRwashdeh, "A Survey of Multibiometric Systems", International Journal of Computer Application volume 43 No 15 April 2012, arXiv:1210.0829.

[5] Karthik Nandakumar, "Multibiometric Systems: Fusion Strategies and Template Security", Degree of Doctor of Philosophy, Department of Computer Science and Engineering, Michigan State University, 2008.

[6] James Wayman, Anil Jain, Davide Maltoni, Dario Maio, "An Introduction to Biometric Authentication Systems", Biometric Systems: Technology, Design and Performance Evaluation, pp 1-20, ISBN: 978-1-84628-064-1, Springer London, 2005.

[7] Pushpa Dhamala, "Multibiometric Systems", Master of Telematics - Communication Networks and Networked, Norwegian University of Science and Technology, Department of Telematics, June 2012.

[8] Wang, Yunhong and Tan, Tieniu and Jain, Anil K., "Combining Face and Iris Biometrics for Identity Verification", Proceedings of the 4th International Conference on Audio- and Video-based Biometric Person Authentication, Pages 805-813, ISBN: 3-540-40302-7, 2003.

[9] http://docs.opencv.org/3.1.0/da/d60/tutorial_face_main.html.

[10] K. Grauman and T. Darrell, "The pyramid match kernel: discriminative classification with sets of image features," Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1, Beijing, 2005, pp. 1458-1465 Vol. 2.

[11] Sinjini Mitra, "Gaussian Mixture Models for Human Face Recognition under Illumination Variations", Applied Mathematics, 2012, 3, 2071-2079, http://dx.doi.org/10.4236/am.2012.312A286 Published Online December 2012 (http://www.SciRP.org/journal/am).

[12] Chaochao Lu, Xiaoou Tang, "Surpassing Human-Level Face Verification Performance on LFW with GaussianFace", Appearing in Proceedings of the 29th AAAI Conference on Artificial Intelligence (AAAI-15), https://arxiv.org/abs/1404.3840.

[13] https://www.technologyreview.com/s/407976/better-face-recognition-software/

[14] Diego A. Socolinsky, Andrea Selinger, "Thermal face recognition in an operational scenario", Proceedings of the 2004 IEEE computer society conference on Computer vision and pattern recognition, Pages 1012-1019.

[15] Stan Z. Li, Anil Jain, "Encyclopedia of Biometrics", Springer Reference, 2009.

[16] R. P. Wildes, "Iris Regognition, An Emerging Biometric Thechnology Proceedings of the IEEE" , Vol. 85, no. 9 (September 1997).

[17] John Daugman, "How Iris Recognition Works", Image Processing. 2002. Proceedings. 2002 International Conference on, 2002, pp. I-33 - I-36 vol.1.

[18] P. Verma, M. Dubey, P. Verma, S. Basu, "Daughman's algorithm method for iris recognition - a biometric approach", International Journal of Emerging Technology and Advanced Engineering (ISSN 2250-2459, Volume 2, Issue 6, June 2012).

[19] http://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html

[20] Proença Hugo and Alexandre, Luís A., "UBIRIS: A noisy iris image database", Proceed. of ICIAP 2005 - Intern. Confer. on Image Analysis and Processing Vol.1, pp.970-977, ISBN:3-540-28869-4 (2005).

[21] Marco Simionato, "An algorithm for identity recognition using iris", Master Thesis, University of Padova, 2006.

[22] http://docs.oracle.com/javase/6/docs/technotes/guides/jni/spec/intro.html