VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV INTELIGENTNÍCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF INTELLIGENT SYSTEMS

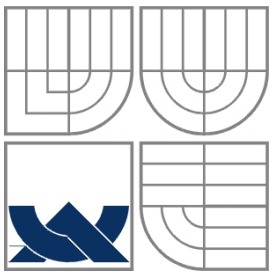
MULTIBIOMETRIC SYSTEM COMBINING IRIS AND RETINA

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

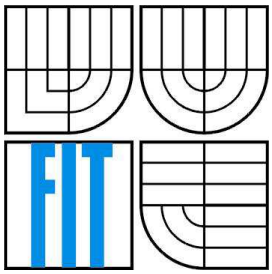
AUTOR PRÁCE
AUTHOR

Bc. PETR JANEČKA

BRNO 2015



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV INTELIGENTNÍCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF INTELLIGENT SYSTEMS

MULTIMODÁLNÍ BIOMETRICKÝ SYSTÉM KOMBINUJÍCÍ DUHOVKU A SÍTNICI

MULTIBIOMETRIC SYSTEM COMBINING IRIS AND RETINA

DIPLOMOVÁ PRÁCE
MASTER‘S THESIS

AUTOR PRÁCE
AUTHOR

Bc. PETR JANEČKA

VEDOUCÍ PRÁCE
SUPERVISOR

Ing. ŠTĚPÁN MRÁČEK

BRNO 2015

Abstrakt

Tato diplomová práce se zabývá multibiometrickými systémy, specificky potom biometrickou fúzí. Práce popisuje biometrii oka, tedy rozpoznávání na základě sítnice a duhovky. Stěžejní část tvoří návrh a implementace biometrického systému, který je založený na rozpoznání sítnice a duhovky.

Abstract

This diploma thesis focuses on multibiometric systems, specifically on biometric fusion. The thesis describes eye biometrics, i.e. recognition based on retina and iris. The key part consists of design and implementation specification of a biometric system based on retina and iris recognition.

Klíčová slova

biometrie, multibiometrický systém, biometrická fúze, duhovka, sítnice

Keywords

biometrics, multibiometric system, biometric fusion, iris, retina

Citace

Petr Janečka: Multibiometric system combining iris and retina, diplomová práce, Brno, FIT VUT v Brně, 2015

Multibiometric System Combining Iris and Retina

Prohlášení

Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně pod vedením pana Ing. Štěpána Mráčka. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

……
Petr Janečka
26. 5. 2015

Poděkování

Chtěl bych poděkovat vedoucímu práce Ing. Štěpánu Mráčkovi za jeho čas a vedení při vypracovávání diplomové práce.

Contents

1 Introduction

In today's security world, biometrics is an interesting approach. Ideally, the user interacts with a simple interface and in a matter of seconds, the biometric system scans the biometric, whether it is a fingerprint or the iris of the eye, and decides if the user is allowed to pass.

However, such systems are not perfect and there is always room for improvement. Recently, it has been discovered that a viable course of the future of biometrics may lie in multibiometric systems [1], which combine more than one source of biometric information for evaluation (such as fingerprint and palm veins), unlike unibiometric systems, which only use one.

This thesis focuses on multibiometric systems, specifically on biometric fusion, which, as the name suggests, fuses the various sources of biometric information in order to determine the user's identity. Furthermore, it focuses on ocular-based biometrics. The purpose of this thesis is to develop a multimodal biometric system combining iris and retina.

In the proceeding chapter, multibiometric systems are expanded upon. The focal point is biometric fusion and its methods. The third chapter is centred on eye biometrics, mainly on the modalities crucial for the multimodal biometric system – iris and retina. Each modality and the algorithms associated with it are discussed in its respective subchapter.

Next, the thesis focuses on the design of the multimodal biometric system. The architecture of the system is detailed along with the description of algorithms used for the selected modalities and for the fusion itself. The proceeding chapter concentrates on the implementation of the system and the technologies used. In the penultimate chapter, the results of the developed biometric system are presented. The conclusion of the performance analysis of the system is the primary topic of the last chapter.

2 Multibiometric Systems

2.1 Biometrics

Before the topic of this chapter can be discussed, *biometrics* should be introduced. The word itself originates from Greek language and literally means measurement of life [2]. The term was coined by an English polymath Francis Galton.

In the world of information technology, biometrics refers to technologies and algorithms employed to measure and analyze anatomic or dynamic features for security purposes such as authentication or recognition.

Not to be confused with biometrics definition above, biometric is a physical trait that can be measured, recorded, and quantified [3]. Iris, fingerprint, face, and the veins of the hand count amongst such traits and are one of the most frequent biometrics used.

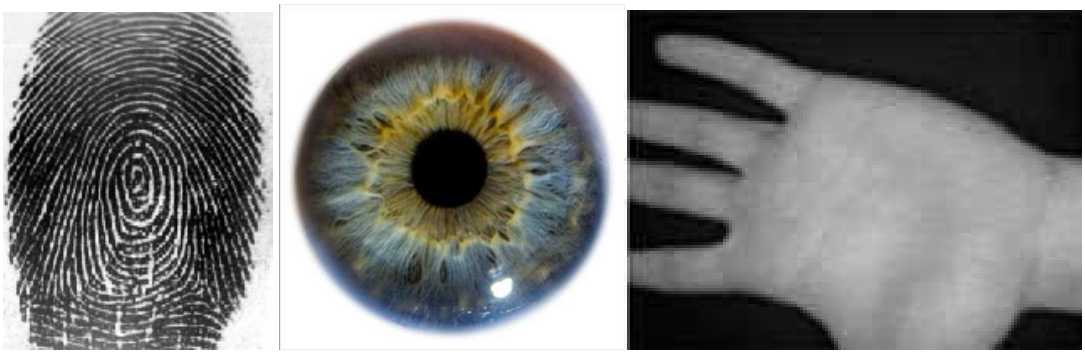


Figure 2.1: Fingeprint [4], iris [5], and palmprint [4] biometrics.

2.2 Biometric System

According to [6], a biometric system is a system that measures one or more physical characteristics, or modalities, of an individual in order to determine or verify their identity.

These two methods present two different principles of biometric systems. While verification serves to confirm a person's identity, the task of identification is to find that identity first. In the case of the former, the user inputs their identity along with the biometric trait. In the case of the latter, it is up to the system to find the user in its database and match the supplied trait with a stored template.

The general schema of a biometric system, as depicted in Figure 2.2, is divided into two modules:

- enrolment
- verification/identification

The user has to register with the system first. They provide their identity and a sensor captures their biometric feature, which is then passed to a feature extractor for further processing. Afterwards, the obtained feature set is stored in a template database and the user successfully completes the enrolment phase.

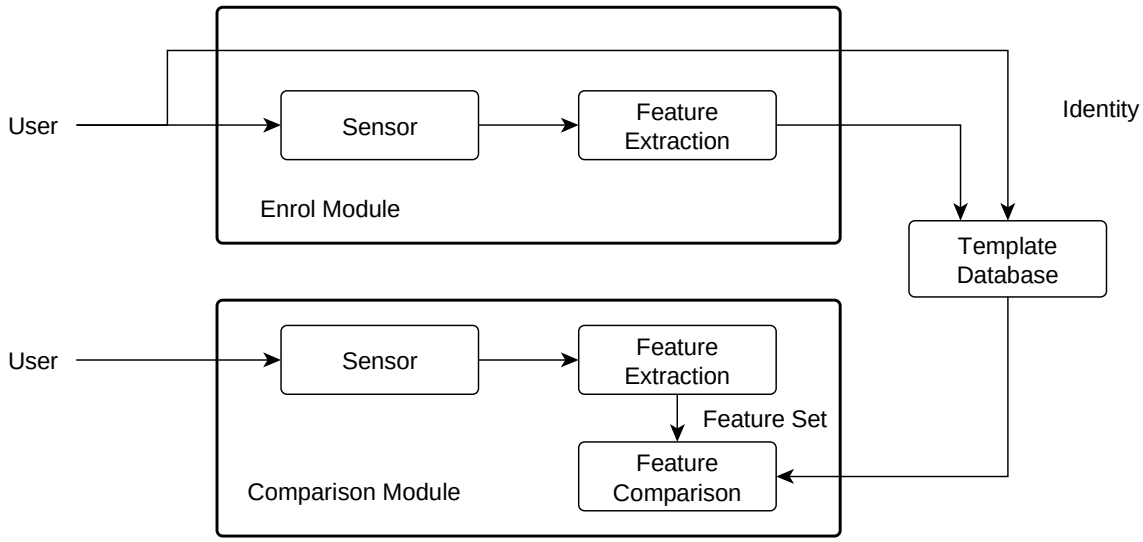


Figure 2.2: Biometric system schema.

Whenever a user is required to interact with the biometric system (e.g. for gaining access to a building), they undergo a similar procedure. Depending on whether the system is set to verify or identify, the user either specifies their identity or not. Then the sensor scans the modality and the module proceeds with feature extraction, which processes the acquired information and extracts the features that can be compared to a template in the database.

In essence, a biometric system thus comprises these blocks [6]:

- sensor
- feature extractor
- template database
- feature comparator and decision maker

The sensor consists of a user interface and a mechanism which scans biometric data. These data are usually unrefined. In many cases (e.g. face recognition, iris recognition), the data are image-based, but they need not to be limited to that. Several modalities are obtained by different means, such as a person's DNA.

Before the collected information is passed to the feature extractor, it is usually pre-processed in order to make the desired features more salient. The common pre-processing methods involve quality assessment, segmentation, and enhancement [6].

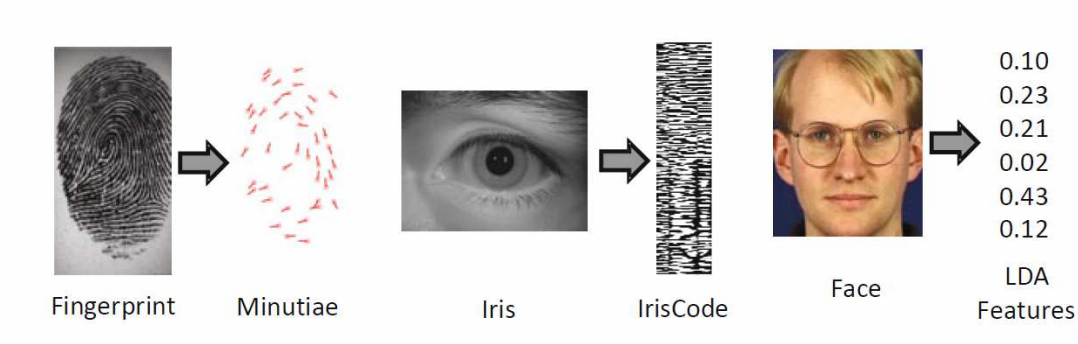


Figure 2.3: Features extracted from chosen modalities [6].

Then the feature extractor processes the data, extracting the features that contain compact and the most prominent details tied to a person's identity.

In the case of enrolment, the feature set is stored as a *template* in the template database. In the case of verification, the corresponding template is obtained from the database and then compared with the recently extracted feature set. The result of this comparison is a score of similarity. The feature comparator is usually programmed to accept scores exceeding a certain threshold while rejecting those below it. In the case of identification, the extracted features are compared to the entire database and the most probable identity surpassing the threshold is selected.

2.2.1 Measuring Performance of a Biometric System

One of the most important aspects of every biometric system is arguably its performance. For the purpose of measuring performance, many metrics exist. Gathered below are the most common [8]:

- **False Acceptance Rate** (FAR) – the rate of comparisons of a particular biometric of two different individuals which lead to acceptance. It is complemented by Genuine Rejection Rate (GRR).
- **False Rejection Rate** (FRR) – the rate of comparisons of a particular biometric of the same individual which lead to rejection. It is complemented by Genuine Acceptance Rate (GAR).
- **False Non-Match Rate** (FNMR) – the proportion of matching decisions from comparing two biometric captures from the same individual that yield a non-match. Unlike with FRR, attempts that fail before comparison are excluded.
- **False Match Rate** (FMR) – the proportion of matching decisions from comparing two biometric captures from the two different individuals that yield a match. Unlike with FAR, attempts that fail before comparison are excluded.
- **Receiver Operating Characteristic** (ROC) **Curve** – a graph plotting the FMR against 1 - FNMR, summarizing system performance, depicted in Figure 2.4.
- **Equal Error Rate** (EER) – the rate at which FNMR equals FMR.

There are a number of other metrics, such as Failure to Acquire, Failure to Enrol, and Failure to Match, etc.

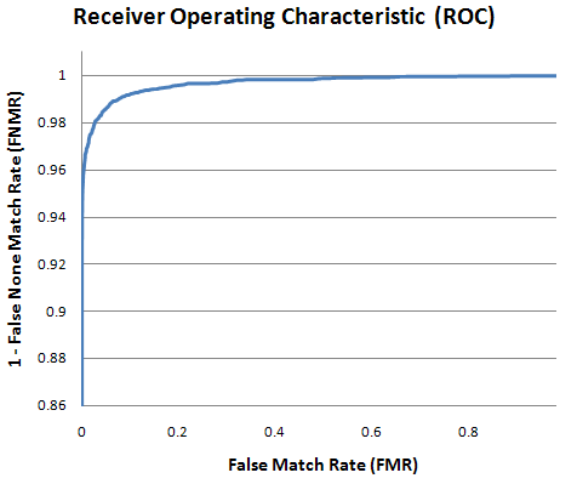


Figure 2.4: ROC Curve [9]

2.2.2 Advantages and Issues of Multibiometric Systems

The generalized model of a biometric system mentioned before could be considered a unibiometric system, i.e. one that uses a single source of evidence for identifying a person or verifying their identity. In contrast to this model stands the multibiometric system. As the name suggests, a multibiometric system uses more than one source of evidence [1].

If designed correctly, these systems can be expected to be more accurate than their unibiometric counterparts [10]. Also, security counts amongst the most prominent benefits of multibiometric systems. By using more sources of evidence, the security of the system can be vastly increased.

In the case of a unibiometric system, specifically one that scans fingerprints, it might be easy to spoof the sensor with a latent fingerprint and beat even very intricate liveness detection algorithms [11]. The deployment of a multibiometric system can effectively prevent this by requiring another modality. One biometric trait might be hard enough for a malefactor to obtain, but any surplus traits will present them with an additional problem that will greatly increase the difficulty of their endeavour.

Multibiometric systems can also help with situations in which unibiometrics systems are considered discriminative. If an individual lacks a particular trait (e.g. has missing fingers in the case of fingerprints), or the trait might be severely deformed that the algorithm cannot acquire it (e.g. cataract in the case of iris recognition), they might be able to provide another biometric and thus the system will allow them to enrol.

Another advantage lies in the fact that some biometrics, such as voice, can be marred by a noisy data signal. Multibiometric systems can remedy this inconvenience by using a supplementary algorithm or a different modality.

Multibiometric systems can operate faster in environments that necessitate a large database. Using more than one biometric trait as search criteria, a database that contains thousands of entries might be traversed more efficiently. For example, one trait would refine the list of potential candidates for an identity match, while another would be then used to determine the identity from this reduced list [12].

That being said, multibiometric systems are not without a disadvantage. There are a number of methods of implementing such a system and some perform poorer than others with certain biometrics while others perform better. It is therefore important to contemplate the aims of the system and design it accordingly.

A multibiometric system usually brings forth the question of additional cost. Not only does the system have to accommodate additional resources such as a sensor or a chip for a surplus algorithm, but the cost of fusion of the acquired data has to be taken into account as well. Also, any new biometric trait required from the user might cause them significant inconvenience. The question that arises from these facts is whether the costs incurred by the aforementioned are outweighed by the overall benefits of the system [1].

While designing a multibiometric system, several factors have to be considered. Besides cost, the following should be mentioned: determining the sources of biometric information, the moment of fusing the sources, and the method of fusing.

2.2.3 Sources of Multibiometric Evidence

Depending on where the multiple sources of biometric information originate, multibiometric systems can be classified into six categories [1]:

1. multi-sensor systems
2. multi-algorithm systems
3. multi-instance systems
4. multi-sample systems
5. multimodal systems
6. hybrid systems

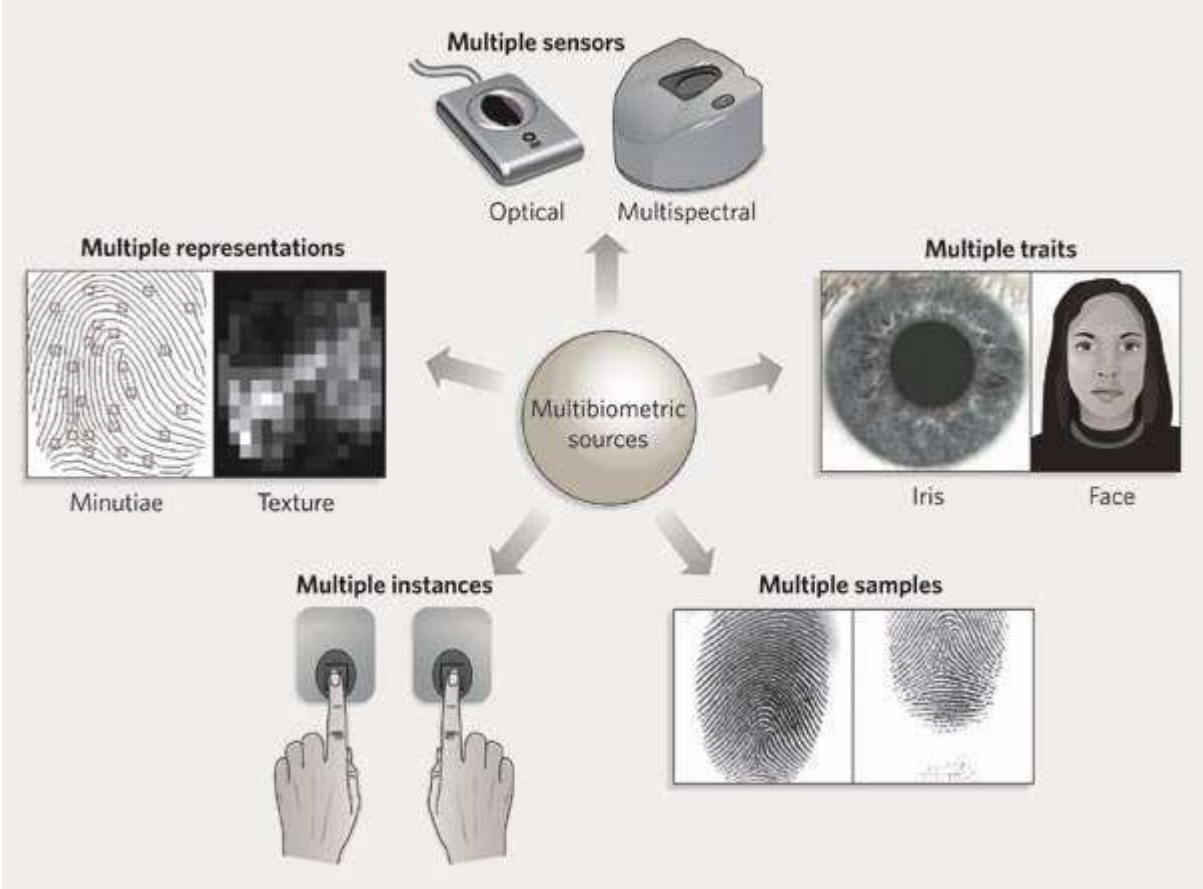


Figure 2.5: Sources of biometric information in multibiometric systems [13].

Multi-sensor systems capture a specific biometric trait with multiple sensors, aiming to acquire distinct information about it. For example, face might be chosen as the desired modality. For capturing, a regular camera might be used together with a thermal infrared one. This particular combination of sensors has been demonstrated to be an improvement over a regular unibiometric face recognition system [14].

Incorporating a new sensor is tied to additional costs, but it also supplies the system with additional data regarding the selected modality, which can assist image pre-processing performance and thus achieve clearer and more noise-free results [15].

Multi-algorithm systems represent another group of multibiometric systems. In this model, the system works with a single piece of biometric information provided by the sensor. Instead of duplicity in the capturing section, there may be two or more algorithms integrated.

An algorithm is developed only once and is deployed with every system. Therefore, additional finances are expended only once. The cost of this system is significantly reduced in comparison to the previous variant, where an additional sensor increases the necessary spending tied to the production of each device. However, it should be mentioned that a new algorithm might mandate surplus computational power.

An instance of a multiple algorithm system is presented in [16], where the authors have implemented three face recognition algorithms, namely Kalman Filtering, Block-Independent Component Analysis, and discrete cosine transform coupled with Fisher's linear discriminant. According to the results of the authors' endeavour, this particular approach outperforms other face recognition approaches.

Multi-instance systems rely on using multiple instances of the same biometric information. Examples of this usage include fingerprints of index fingers of both hands (depicted in Figure 2.5), irises of both eyes, or retinas of both eyes.

Unlike the multi-sensor variant, this normally does not require an additional sensor, and unlike the multi-algorithm variant, this does not require an additional algorithm. Therefore, the cost of designing and implementing this kind of system is even lower in most cases.

However, certain situations demand the introduction of a secondary sensor (or sensors), such as when concurrent acquisition of biometric traits is needed. Obtaining the fingerprints of every finger simultaneously serves as an example.

The multibiometric option is used mainly in systems with large databases. It is also useful in situations where a single instance of the biometric is not sufficient. Two irises can offer better biometric details if the individual's eyes are obscured too much by eyelids. Therefore, a multi-instance system can mitigate the problem of unsatisfactory biometric information.

Multi-sample systems usually employ one sensor that captures a single biometric several times to ameliorate the issues that stem from noise and variation occurring in the biometric. Providing multiple samples to the system can result in a clearer and more precise interpretation of the evaluated trait.

The system might photograph the user from various angles and then proceed to fuse the gathered data, or it might scan the user's fingerprint twice and merge the images into one, possibly negating the deficiencies of the separated images, such as missing or blurred regions of the fingerprint.

Multimodal systems combine two or more biometric traits. For example, the subject of this thesis is a multimodal biometric system that combines iris and retinal recognition. Obviously, the combinations of modalities are abundant, though a system scarcely uses more than three modalities at once due to practical reasons.

While the cost of these systems is higher in comparison to the previously mentioned variants due to the need for more sensors, interfaces, and algorithms, it can be compensated by increased efficiency, as evidenced in a particular face and voice recognition biometric system [17].

Hybrid systems represent a specific category which refers to systems incorporating two or more of the aforementioned groups. In [18], for instance, the authors demonstrate a system that extracts face as well as fingerprint features. Each trait is then analyzed by three distinct algorithms.

This particular variation of multibiometric systems is a subject of research and offers viable possibilities in the foreseeable future [19].

2.2.4 Biometric Fusion

An important aspect of a multibiometric system is the fusion of the gathered information. At certain point during the recognition routine, it is necessary to merge the data to a single entity before proceeding further.

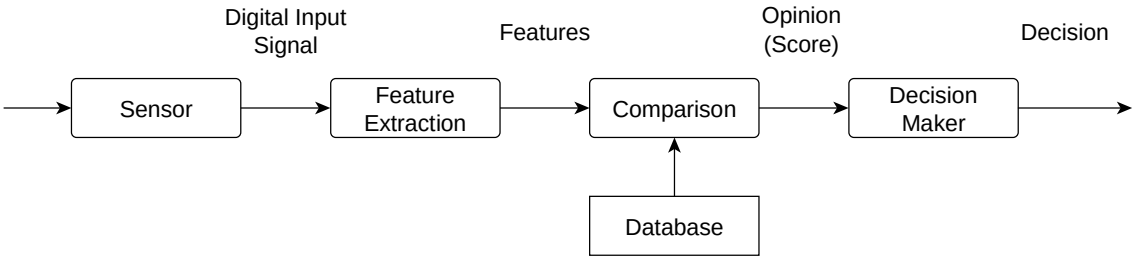


Figure 2.6: Biometric system model.

This in itself poses a significant challenge in the designing phase of multibiometric system development. As shown in Figure 2.6, there are four separate operations that the system performs. At each, fusion can generally be introduced in the system.

It is worth noting that as the data advance through the system, their amount is compressed along the way. However, this does not necessarily imply that the sooner the fusion occurs, the better the results.

While the data at sensor level are arguably of larger quantity than those at feature level, the latter have usually been stripped of superfluous details and noise. On the other hand, it is possible that the feature extraction module could have produced specious results which could have been remedied at sensor level otherwise.

The classification of biometric fusion is depicted below:

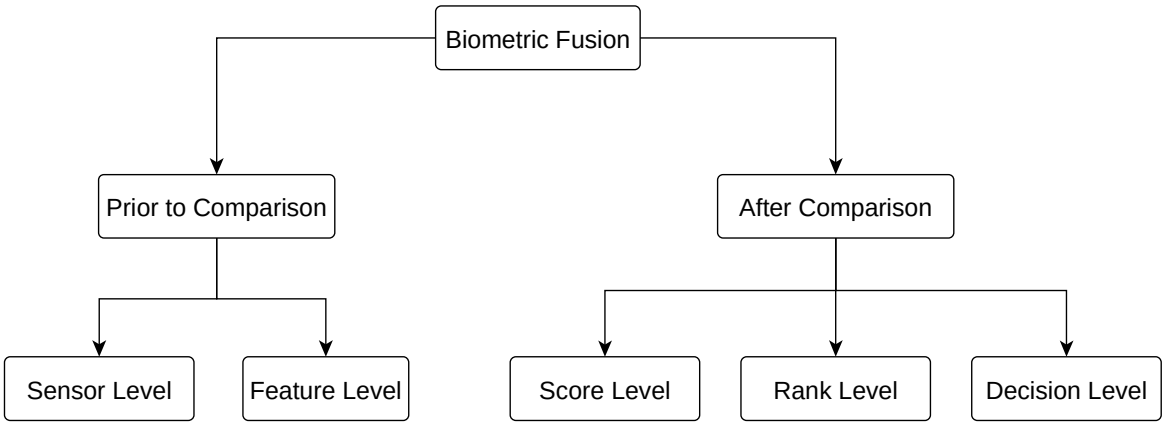


Figure 2.7: Level of biometric fusion.

Biometric fusion is broadly divided into two sections – fusion before comparison and after comparison. The reason for this classification originates from the fact that after comparison, the amount of information available to the system decreases by a significant margin, which is commonly far greater than in the other cases [1].

2.2.5 Sensor-level Fusion

The first possibility of performing biometric fusion is present at sensor level. It involves joining multiple sources of raw evidence prior to extracting features. This can encompass text, images, videos, etc.

At this level, the obtained data contain the most information available. As such, these data are very likely to be marred by noise. However, this can be remedied by a carefully constructed fusion algorithm.

In image processing, a particular method of fusion is employed, often referred to as *mosaicing*. In this process, a composite image is constructed from overlapping component images [21]. This technique is used in biometrics to improve performance of a biometric system.

Mosaicing can enhance multi-sample systems that scan the same biometric several times [1]. For example, many consecutive fingerprint captures might reveal more detail of the ridge structure and present more salient information about the minutiae.

Mosaicing is particularly useful for devices that employ sweeping technique to obtain a person's fingerprint [22]. Each image produced by the sensor represents a segment of the fingerprint, which then needs to be joined with the rest.

The technique of mosaicing can be used in face recognition as evidenced in [23]. First, a number of cameras acquired the photographs of the face from distinct angles. The authors attached markers of different colours to the face in order to help identify the corresponding points in each picture through colour segmentation. Afterwards, linear transformations were carried out to create a mosaic of the face.

Another usage of sensor-level fusion can also be found in face recognition, in which several two-dimensional images captured at various angles can produce a three-dimensional model of the face.

Sensor-level fusion does not limit itself to needing sensors of the same type or sensors scanning the same biometric. An example is displayed in Figure 2.8. In this arrangement, there are two sensors – one capturing the user's face, another localizing the user by sound. The information extracted from the sensors is then fused to provide a two-dimensional image.

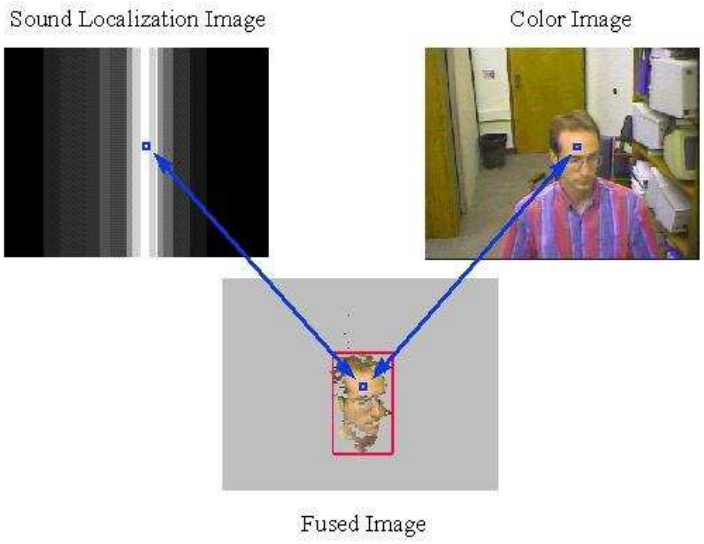


Figure 2.8: Fusion based on sound localization and face capture [24].

This implies that the user is required to be speaking in order for the biometric system to work correctly. Also, both sensors are required to be centred on a common area of interest in order to avoid attributing biometric information falsely, especially if there are several people present in range.

Besides one-time authorization, this system could be deployed for speaker recognition or continuous scanning useful in surveillance efforts.

Voice recognition can benefit from sensor-level fusion as well. The quality of captured voice samples of an individual depends not only on the quality of the sensor device but also on the sound generated in the background. Most superfluous sounds are deemed sources of noise and therefore, their reduction is a desirable.

This can be achieved by the deployment of two or more microphone sensors. The simultaneous input signals can be used to cancel the noise of each component and help with blind source separation [25].

2.2.6 Feature-level Fusion

In feature-level fusion, sources of evidence are consolidated after features have been extracted from their respective samples. Following this, fused feature data are then passed to feature comparator module and the system proceeds as if dealing with a single source of biometric evidence.

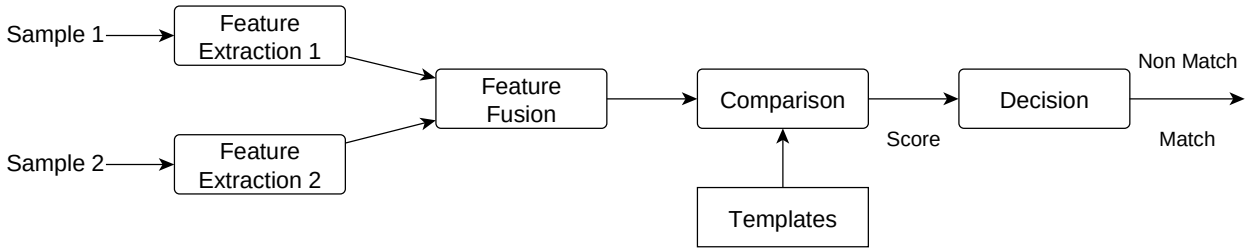


Figure 2.9: Schema of a biometric system using feature-level fusion.

Feature sets of distinct modalities, or feature sets of identical modalities that have been extracted by different algorithms pose a challenge for numerous reasons [1]:

1. It may be a problem to fuse two chosen modalities if the basis on which they should be fused is not known. In these cases, it may be difficult to produce a fused set of features that would satisfy the demands on improvement over a unibiometric system.
2. The previous might be exacerbated by the situation in which feature sets of different modalities are not compatible. This occurs in the case of fingerprint minutiae and eigen-coefficients of face. The former are varied in length, while the latter are a fixed-length set of features.
3. Considering the usage of fixed-length sets of features, a superset of the separate sets might be created by concatenation. However, this may produce an undesirable effect of curse-of-dimensionality, which states that an increasing number of dimensions does not necessarily ensure higher performance of a system, but rather degrades it [26].

In the case of concatenation, the process of feature-level fusion of distinct modalities or algorithms can undergo two stages – *feature normalization* and *feature selection* – in order to fuse the multiple sources of biometric evidence [27]. It is necessary to denote feature vectors *X* and *Y* to be fused:

$$\begin{aligned} X &= \{x_1, x_2, ..., x_m\}, x \in R_m \\ Y &= \{y_1, y_2, ..., y_n\}, y \in R_n \end{aligned} \tag{2.1}$$

Separately, these feature vectors would be utilized in a unibiometric system and compared against a template. Combined, they need to be merged to a final feature vector *Z* before such a comparison. In order to achieve this, the discrepancies of vectors *X* and *Y* in dimension and distribution have to be managed first. This stage of the fusing algorithm is called feature normalization.

The objective of feature normalization is to alter the location and the scale of a feature vector so that the contribution of each component to the comparison process is comparable [28]. It also ameliorates the issue of outlying values.

There are several techniques of feature normalization, for example the *min-max technique*. It defines *x'* as follows:

$$x' = \frac{x - \min(F_x)}{\max(F_x) - \min(F_x)} \tag{2.2}$$

where F_x is a function which generates *x*, $min(F_x)$ is the minimum of all *x* values, and $max(F_x)$ is the maximum value of all *x* values. The min-max technique is efficient when the minimum and maximum are available before the calculation itself. If that is not the case, however, it may be possible to acquire this information from outlying values.

Another technique, which is also relatively indifferent to noise, is based on median. It defines *x'* accordingly:

$$x' = \frac{x - median(F_x)}{median(|\, x - median(F_x)\,|)} \tag{2.3}$$

The denominator is called the *Median Absolute Deviation*. While this function is mostly insensitive to outliers due to the nature of median, its effectiveness might not be adequate in situations where the distribution of the score is not Gaussian, because of the poor estimation of location and scale by median.

By applying either technique (or any other not mentioned), the altered feature vectors *X'* and *Y'* will be obtained:

$$\begin{aligned} X' &= \{x'_1, x'_2, ..., x'_m\} \\ Y' &= \{y'_1, y'_2, ..., y'_n\} \end{aligned} \tag{2.4}$$

Then, the next step towards fusing *X'* and *Y'* vectors is feature selection. Taking the curse-of-dimensionality issue into account, a simple concatenation may not yield the most viable outcome. Additionally, some values acquired can be damaged by noise. Therefore, a fruitful approach might lie in selecting a minimized feature set of size *k*, where $k < (m + n)$, in order to produce satisfactory results.

Chosen solutions of this problem could be [29]:

- *Sequential forward selection,* in which we start with an empty set and the most significant features are added iteratively.
- *Sequential backward selection,* in which we start with the full set and the least significant features are deleted iteratively.
- *Sequential forward floating search,* which combines the previous methods.

Every approach mentioned requires criteria that define the significance of each feature value. There are several criteria, many of which are based on the metrics in Chapter 2.2.1. For example, the Equal Error Rate can be used, although it does not summarize the performance of all thresholds. One option of defining an objective function could be the average of GAR corresponding to four different FARs [27].

The final vector *Z*, which represents the fused vectors *X* and *Y*, is the result of feature selection.

There are many studies which elaborate on the topic of feature-level fusion. For example, there exists a biometric system fusing face recognition, based on Scale Invariant Feature Transform, and fingerprint verification, based on minutiae [30]. Invariance to rotation and translation of the SIFT features and keypoint descriptors around minutia position ensure the compatibility of the two sets.

The feature sets are subsequently concatenated to provide a single fused set. Further data reduction is achieved with three techniques (K-means clustering, neighbourhood elimination, and specific regions selection).

Another system which incorporates feature-level fusion is proposed in [31]. The authors combined fingerprints and finger veins biometric. As feature extraction technique of both biometric traits, they chose Gabor filter. To fuse these traits, the authors utilized canonical correlation analysis (proposed by Hotelling as a statistical tool for identifying linear relationships between two sets of variables [32]). According to the results of the research, it is a robust method of personal identification and could be further expanded upon with other modalities of the finger.

A third example of feature-level fusion is demonstrated with merging face and palmprint traits for small sample biometric recognition [33]. The authors used Gabor filter to extract features as well. In order to fuse the biometrics, they joined the feature sets vertically and normalized the values by using mean values. Afterwards, with nonlinear discrimination analysis in Kernel space they created the resulting feature vector. Following experiments with the system proved this to be an efficient solution to small sample biometric recognition.

2.2.7 Score-level Fusion

At score level, the data are fused after the features have been examined and a score has been produced. It is relatively undemanding to combine the scores generated at this level. Additionally, the score data contain salient information about the input pattern, thus fusion at score level is also the most frequently employed approach in multibiometric systems.

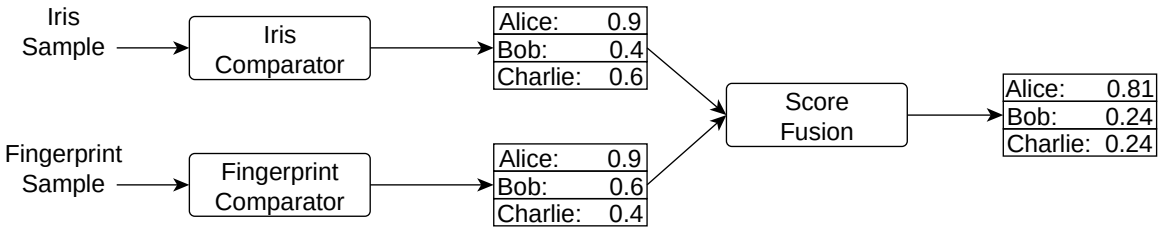


Figure 2.10: Example of biometric fusion at score level.

This type of fusion could be divided into three categories [1]:

- density-based
- transformation-based
- classifier-based

2.2.7.1 Normalization

Individual comparison units might not produce homogenous data. For example, the data may differ in their range, representation, and probability distribution. Therefore, it is necessary to address this issue and normalize the comparison outputs.

One of the methods of score normalization is called *decimal scaling.* This is usually applied when the scores are on a logarithmic scale:

$$ns_j^t = \frac{s_j^t}{10^{n_j}} \tag{2.5}$$

where $n_j = \log_{10} \max_{i=1}^{N} s_j^i$ and s_j^i is the i^{th} score produced by the j^{th} comparator.

In case of problems with outlying values or uneven tail distribution, the median and median absolute deviation can be used. Unfortunately, this method does not perform well when score distribution is not Gaussian.

$$ns_j^t = \frac{s_j^t - median_{i=1}^{N} s_j^i}{median_{i=1}^{N} \mid s_j^i - median_{i=1}^{N} s_j^i \mid} \tag{2.6}$$

Tanh-estimators [34] present a robust and highly efficient way of score normalization. The function is defined as

$$ns_j^t = \frac{1}{2}\left(\tanh\left(0.01\left(\frac{s_j^t - \mu_{GH}}{\sigma_{GH}} \right) \right) + 1 \right) \tag{2.7}$$

where μ_{GH} is the mean and σ_{GH} is the standard deviation estimate given by Hampel estimators and their parameters *a*, *b*, *c*. The estimators are defined by the following function:

$$\psi(u)=\begin{cases} u & 0\le|u|<a \\ a*sign(u) & a\le|u|\le b \\ a*sign(u)*\left(\frac{c-|u|}{c-b}\right) & b\le|u|\le c \\ 0 & c\le|u| \end{cases} \tag{2.8}$$

where

$$sign(u)=\begin{cases} +1 & if\ \ u\ge 0 \\ -1 & otherwise \end{cases} \tag{2.9}$$

Tanh-estimators significantly reduce the influence of the scores at the tail section of distribution and are not sensitive to outlying values. However, if there are many values in the tail, the method might not prove efficient. In order to achieve the best results, the parameters have to be selected carefully.

2.2.7.2 Theoretical Framework

Score-level fusion was discussed in length in [35], in which the authors define a theoretical framework for combining scores of distinct comparisons.

Given the problem of classifying an input pattern X into one of M possible classes $\{\omega_1, \omega_2,\ldots, \omega_M\}$, let there be R classifiers, each representing a given biometric pattern by a specific measurement vector. Denote the vector used by the i^{th} classifier by x_i. Each class ω_k is represented by the probability density function $p(x_i \mid \omega_k)$ and its probability is $P(\omega_k)$.

According to Bayesian theory, given measurements x_i, $i = 1,\ldots, R$, and the pattern X, should be assigned to class ω_i so that the a posteriori probability of that interpretation is maximum:

$$\begin{gathered} assign\ \ X\to\omega_i\ \ if \\ P(\omega_i\mid x_1,\ldots,x_R)\ge\max_{k=1}^{M}P(\omega_k\mid x_1,\ldots,x_R) \end{gathered} \tag{2.10}$$

where $k = 1\ldots M$. The a posteriori probabilities $P(\omega_k \mid x_1,\ldots, x_R)$ can be expressed using Bayes theorem as follows:

$$P(\omega_k\mid x_1,\ldots,x_R)=\frac{p(x_1,\ldots,x_R\mid\omega_k)P(\omega_k)}{P(x_1,\ldots,x_R)} \tag{2.11}$$

where $p(x_1,\ldots x_R)$ is the unconditional measurement joint probability density. It is defined in terms of conditional measurement distributions as

$$p(x_1,\ldots,x_R)=\sum_{i=1}^{m}p(x_1,\ldots,x_R\mid\omega_i)P(\omega_i) \tag{2.12}$$

2.2.7.3 Transformation-based Fusion

The previous definitions can be used to deduce rules for score-level fusion, where the number of classes *M* can be reduced to two, because there are only two types of classes – genuine users and impostors. Namely:

- product rule
- sum rule
- max rule
- min rule
- median rule

The *product rule* is based on the assumption that the feature representations $x_1,\ldots,x_R$ are statistically independent. It can be denoted accordingly as

$$assign \quad X \rightarrow \omega_r \quad if$$
$$\prod_{i=1}^{R} P(\omega_r \mid x_i) \geq \prod_{i=1}^{R} P(\omega_k \mid x_i), \quad k = 1,\ldots,M \tag{2.13}$$

The issue with the product rule is that it is sensitive to errors. If merely one classifier yields a probability close to zero, the resulting product is then also lowered to a near-zero value and this may lead to an incorrect classification decision.

The *sum rule* is based on the assumption that the a posteriori probabilities do not deviate notably from a priori probabilities, although it is worth noting that this assumption might not always be true. The rule is resistant to errors in the estimation of a posteriori probabilities.

It can be defined as

$$assign \quad X \rightarrow \omega_r \quad if$$
$$\sum_{i=1}^{R} P(\omega_r \mid x_i) \geq \sum_{i=1}^{R} P(\omega_k \mid x_i), \quad k = 1,\ldots,M \tag{2.14}$$

The *max rule* estimates the mean of the a posteriori probabilities by their maximum value as follows:

$$assign \quad X \rightarrow \omega_r \quad if$$
$$\max_{i=1}^{R} P(\omega_r \mid x_i) \geq \max_{i=1}^{R} P(\omega_k \mid x_i), \quad k = 1,\ldots,M \tag{2.15}$$

The *min rule* is the exact opposite of the max rule and can be defined accordingly:

$$assign \quad X \rightarrow \omega_r \quad if$$
$$\min_{i=1}^{R} P(\omega_r \mid x_i) \geq \min_{i=1}^{R} P(\omega_k \mid x_i), \quad k = 1,\ldots,M \tag{2.16}$$

If a priori probabilities are assumed to be equal, the sum rule can be regarded as the average of the a posteriori probabilities. However, if one of the classifiers produces an outlying value, the average will be affected and this may result in incorrect decision. This can be remedied by using the median, leading to the *median rule*:

$$assign \quad X \rightarrow \omega_r \quad if$$
$$\underset{i=1}{\overset{R}{median}} P(\omega_r \mid x_i) \geq \underset{i=1}{\overset{R}{median}} P(\omega_k \mid x_i), \quad k = 1,\ldots,M \tag{2.17}$$

2.2.7.4 Density-based Fusion

The next technique is *density-based*. In this case, it is necessary to define S_{gen} which denotes genuine comparison scores and S_{imp} which denotes impostor comparison scores. Also, let $F_{gen}(s)$ be the distribution function of S_{gen} and $f_{gen}(s)$ be the corresponding density [1]:

$$P(S_{gen} \leq s) = F_{gen}(s) = \int_{-\infty}^{s} f_{gen}(v)dv \tag{2.18}$$

Accordingly, $F_{imp}(s)$ is the distribution function of S_{imp} and $f_{imp}(s)$ is the corresponding density function:

$$P(S_{imp} \leq s) = F_{imp}(s) = \int_{-\infty}^{s} f_{imp}(v)dv \tag{2.19}$$

The densities $f_{gen}(s)$ and $f_{imp}(s)$ represent the probability density functions of the comparison score. They are usually not known beforehand and have to be approximated from a training set of genuine and impostor classes. The estimation can be done either by parametric or non-parametric methods [36].

In the case of the former, the form of the function is known, but the parameters have to be estimated from the training data. In comparison, the latter do not assume any shape of the function, but rather rely on the data.

Because of the limited availability of training data, it is imperative to select the density estimation method carefully.

However, in certain situations, the data are so scarce that estimation is not possible. Under such conditions, it is far more viable to combine comparison scores directly without converting them into a posteriori probabilities.

2.2.7.5 Classifier-based Fusion

The last approach is *classifier-based* fusion. In this technique, a pattern classifier is used to learn the relationship between the score vector and the a posteriori probabilities of the genuine and impostor classes [36].

In this method, the vectors are split into two categories based on the aforementioned classes. Given a training set, the pattern classifier derives a boundary between the two classes. Because the classifier is able to learn the decision boundary regardless of the format of the score vectors, they are not required to be homogenous [1]. With this approach, there are several models on which the classifiers can be based, including but not limited to [37]:

- k-nearest neighbour
- decision tree
- logistic regression
- linear discriminant

For example, the authors of [38] implemented a biometric system fusing data on score level with decision tree and linear discriminate classifiers. They combined fingerprint, face, and hand geometry modalities.

2.2.8 Rank-level Fusion

It should be noted that rank-level fusion is only applicable in biometric systems that are set to identify a person, not to verify their identity. However, it is still one of the more frequent levels of fusion.

After processing the feature vector and acquiring the comparison score, the set of probable matching identities can be sorted in descending order and thus create a ranked list of candidate identities.

The aim of this level of fusion is to merge the ranks produced by individual biometric modules in order to gain a consolidated list of ranks for each identity. While ranks provide more information than decisions, they reveal less than scores.

However, there is a significant advantage in rank-level fusion in comparison to score-level fusion. Ranks are comparable directly and therefore do not need to be normalized. As a result, biometric systems using this level of fusion are generally easier to design.

Given M of users enrolled in a database and R of comparators, let $r_{j,k}$ be the rank assigned to user k by the j^{th} comparator, $j = 1,\ldots, R$ and $k = 1,\ldots, M$. Let s_k be a statistic computed for user k such that the user with the lowest value of s is assigned the highest consensus rank. Then, there are three methods to compute the statistic s [39]:

- highest ranking method
- Borda count method
- logistic regression method

In the *highest ranking method*, each user is assigned the best of the ranks computed by different comparators. The statistic for user k can thus be defined as

$$s_k = \min_{j=1}^{R} r_{j,k} \tag{2.20}$$

If the resulting statistics contain ties, they are resolved randomly. The highest ranking method is good for combining a small number of classifiers, each of which specializes on inputs of a particular type. It is especially useful in situations where few classifiers (comparators) are available yet there are several classes (users). In case of a user being ranked high only once, it is still probable that they will be assigned a high rank after fusion. On the other hand, this method is not very practical in situations involving a large number of classifiers.

The *Borda Count method* is a generalization of majority voting. It calculates the sum of the ranks in order to deduce the final rank.

$$s_k = \sum_{j=1}^{R} r_{j,k} \tag{2.21}$$

It is simple to implement the Borda count method and in addition, it does not necessitate training. However, because it considers that the comparators perform equally and the ranks are statistically independent, it does not take into account the differences of individual comparators. Therefore, in situations where certain comparators provide more accurate results than others, this method may not be suitable.

Logistic regression can be obtained by modifying the Borda count method and assigning weights to the ranks produced by each comparator. Therefore, it deals with the issue of difference of comparator quality.

$$s_k = \sum_{j=1}^{R} w_j r_{j,k} \tag{2.22}$$

The weight denoted by w_j is based on logistic regression [40]. Unlike the Borda count method, it requires training before it can efficiently decide the weights.

The aforementioned methods are gathered in Table 2.1. At first, comparators assign the ranks to the enrolled users separately. The acquired ranks are then fused by respective methods, yielding varying results.

	Separate Comparators		Highest Rank		Borda Count		Logistic Regression $w_1 = 0.8$, $w_2 = 0.2$	
User	Comparator 1	Comparator 2	Fused Score	Reordered Rank	Fused Score	Reordered Rank	Fused Score	Reordered Rank
Alice	2	4	2	3	6	3	2.4	2
Bob	3	1	1	1	4	2	2.6	3
Clara	1	2	1	2	3	1	1.2	1
David	4	3	3	4	7	4	3.8	4

Table 2.1: Example of rank-level fusion methods.

2.2.9 Decision-level Fusion

Decision-level fusion is particularly useful in situations where two or more finished biometric systems are available and they need to be fused. More often than not, decision-level fusion is the only option in this case. There are several ways of fusing data at this point [1]:

- AND/OR rules
- majority voting
- weighted majority voting
- Bayesian decision fusion
- Dempster-Shafer theory of evidence
- Behaviour knowledge space

The easiest method in a multibiometric system in verification mode is to implement *AND/OR rules*. The result of an AND rule is a match when all the parts of the system decide that the input and the template match. In the case of an OR rule, only one affirmative output is required to declare the user as genuine.

When AND rule is effected, the FAR of the system is lower than the FAR of individual comparators, while the FRR is higher than the FRR of individual comparators. On the contrary, OR rule produces higher FAR and lower FRR than separate comparators.

One of the main downsides of AND/OR rules method is that if one of the comparators has a significantly higher EER, this method may actually decrease the performance of the multibiometric system [41].

Majority voting is one the most frequent techniques of decision-level fusion [1]. In this method, the input data are proclaimed to match with the identity on which a majority of the comparators agree. If there is no majority consensus on the user's identity, the resulting decision is a rejection.

Because standard majority voting approach does not apply weights, it does not take into account the possible inequality of individual comparators. However, it is a relatively easy and effective method that does not need training and does not require a priori knowledge about the comparators. Although it is a common method of decision-level fusion, there are limits to its accuracy [42].

Weighted majority voting solves the aforementioned issue of imbalanced comparators. In order to facilitate weighting, the labels output by the individual comparators are converted to degrees of support for the *M* classes accordingly [1]:

$$s_{j,k} = \begin{cases} 1 & \text{if output of the } j^{th} \text{ matcher is class } \omega_k \\ 0 & \text{otherwise} \end{cases} \tag{2.23}$$

where $j = 1,\ldots, R$ and $k = 1,\ldots, M$. The discriminant function for class ω_k computed using weighted voting is

$$g_k = \sum_{j=1}^{R} w_j s_{j,k} \tag{2.24}$$

where w_j is the weight assigned to the j^{th} comparator.

Bayesian decision fusion converts the decisions of respective classifiers to probability values. First, a confusion matrix has to be generated for every classifier by using a training set. The Bayes rule for this situation can be defined as follows:

$$P(\omega_k \mid c) = \frac{P(c \mid \omega_k)P(\omega_k)}{P(x)} \tag{2.25}$$

where ω_k denotes classes for $k = 1,\ldots, M$. Because the denominator is independent of the class, it can be omitted, resulting in the discriminant

$$g_k = P(c \mid \omega_k)P(\omega_k) \tag{2.26}$$

This method selects the class with the largest discriminant value. The accuracy of this fusion rule is considered robust [43].

Dempster-Shafer theory of evidence assigns a level of belief to uncertain events [44]. It is more flexible than using probabilities and useful when information concerning the decision problem is incomplete.

To compute the belief functions, a decision profile matrix has to be calculated [1]. It is given by

$$DP = \begin{bmatrix} s_{1,1} & \ldots & s_{1,k} & \ldots & s_{1,M} \\ \ldots & & & & \\ s_{j,1} & \ldots & s_{j,k} & \ldots & s_{j,M} \\ \ldots & & & & \\ s_{R,1} & \ldots & s_{R,k} & \ldots & s_{R,M} \end{bmatrix} \tag{2.27}$$

where $s_{j,k}$ is the degree of support defined in Equation 2.24. From the matrix, this method calculates the belief values and the template with the highest value is matched with the user.

Behaviour knowledge space uses a training data set and a lookup table in order to map the decisions of multiple classifiers to a single one. Also, a vector of classifier decisions $c = [c_1,\ldots,c_R]$ is used. This vector represents a point in the *R*-dimensional space, named behaviour knowledge space. These points are then sorted into bins.

During the training phase, the decision with the highest number of samples is estimated. During the verification phase, a test pattern is passed to classifiers and a decision vector is retrieved. Afterwards, the corresponding bin is identified. If there is more than one possible result in the bin, the decision is chosen at random [1].

3 Eye Biometrics

The eye is a sensory organ, but besides its primary function, it can also serve another particular purpose. Because of its complex structure and rich details of its parts, it is a viable candidate for biometric recognition.

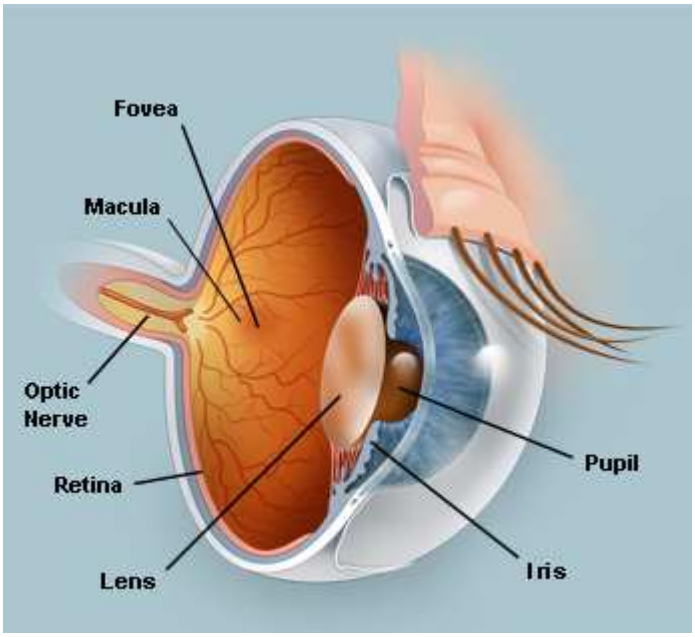


Figure 3.1: Eye anatomy [46].

There are three main sources of biometric information that have been extensively studied and are used in biometric systems [6]:

- iris
- retina
- conjunctival vasculature

While iris and retina are one of the most common biometric modalities and are proved to be efficient, conjunctival vasculature does not reach such levels of security and therefore is not as frequent a modality as the other two [45].

3.1 Iris

Annular in shape, the iris is located between the pupil and the sclera. Its purpose is to limit the amount of light entering the eye. It does so by controlling the size and the diameter of the pupil. In case of too much light, the iris expands and shrinks the pupil, while doing the exact opposite if light is insufficient.

Anatomically, the iris is a multilayered structure. It consists of pigmented fibrovascular tissue, called *stroma*, and pigmented epithelial cells. The stroma is attached to sphincter muscles that are responsible for pupil contraction in circular motion and dilator muscles that are responsible for

pulling the iris radially in order to enlarge the pupil. The back layer contains epithelial cells that are rich in pigment and are impassable by light [6].

The pigment that gives the iris its unique colour is called *melanin*. The stroma, muscles, and border layers comprise the foremost visible portion of the iris. This portion is divided into the central pupilary section and the surrounding ciliary section. These sections are separated by a meandering circular ridgeline called the *collarette*.

In close proximity of the collarette, there are gap-like structures, called *crypts*, which allow easy transmission of fluids in the iris for the purposes of dilation and contraction. In the area delimited by the pigment frill (the boundary between the pupil and the iris) and the collarette, there are spoke-shaped formations called *radial furrows*.

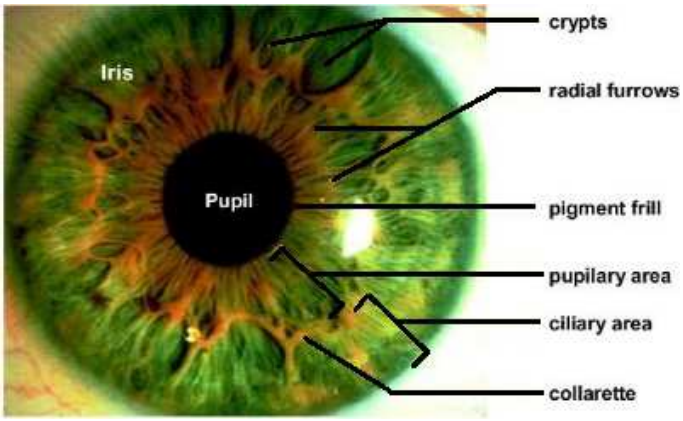


Figure 3.2: Iris detail [47].

The iris provides a detailed texture that is very diverse across human population. While some biometric traits change with age, the iris stops developing around the age of two. In the case of twins, iris recognition has an advantage over face recognition. Monozygotic twins may be nigh identical in terms of facial features, but their irises are highly likely to display several differences in texture.

The colour of the iris depends on the level of pigmentation present. It is defined by the number of melanin granules, which is genetically determined. However, there are other factors that affect the colour, such as the cellular density of the stroma [6]. It is worth noting that the hue of the iris does not bear any significance in iris recognition. The important source of information in this modality is the texture detail.

3.1.1 Advantages and Disadvantages

Iris scanning is considered unintrusive because there is no direct contact between the user and the sensor. It is non-invasive, unlike retinal scanning. People wearing contact lenses or glasses generally do not pose any difficulty to the sensor.

Iris recognition is relatively fast because of its small template. It also offers a broad level of scalability. As such, this modality is frequently used with large-scale applications, for example in ATMs.

Because the iris is relatively small, it is not very suitable for long distance recognition. Additionally, the iris can be obscured by the eyelids and eyelashes, which can complicate the acquisition of the biometric trait.

Iris recognition can be discriminative, as there are a number of illnesses, especially those that come with age (e.g. cataracts), which can severely diminish the ability of a person to be enrolled in the database.

While the intrusiveness of iris scanning is minimal, the user is still required to cooperate with the system in order to enrol and authenticate. Despite the non-invasiveness of iris recognition, some people may be reluctant to cooperate because of their fear of damaging their eye.

Although iris recognition is considered reliable and its performance places it amongst the best biometrics, there are concerns as to its security. For example, it may be possible to spoof some iris sensors by wearing a contact lens designed for this particular purpose.

Also, the costs of developing and maintaining a biometric system based on iris recognition can prove very expensive.

3.1.2 Iris Recognition

Iris recognition follows the general schema of biometric systems that was introduced earlier. For the purposes of simplicity, a system in authentication mode will be taken into account. The goal of such a system is to compare two images of irises in order to provide a final comparison score.

The system can be divided into modules:

- image acquisition
- segmentation
- normalization
- feature extraction
- comparison

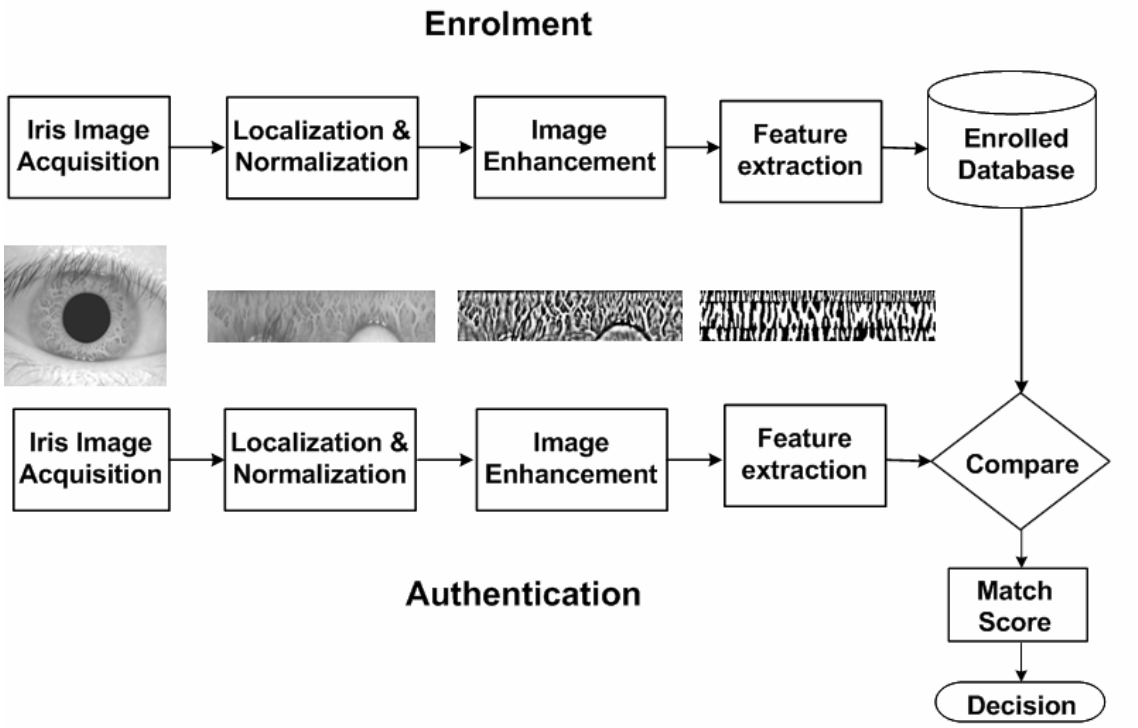


Figure 3.3: Iris recognition system [6].

3.1.2.1 Acquisition

At first, the system proceeds through acquisition phase. Usually, a camera that is sensitive to near-infrared light (700nm – 900nm range of the infra-red spectrum) is used because the iris pigment, melanin, absorbs the majority of this electromagnetic signal. Therefore, the image contains salient information about the iris texture.

While visible light might be used to acquire images in some cases, it might hamper the efficiency of the resulting system due to the compromise in quality. Because the iris can be of distinct colour, a visible-light camera might produce varying results.

Also, near-infrared light has a significant advantage over visible light – it is not deemed intrusive, because human eyes are not equipped to recognize this portion of electromagnetic spectrum. The illumination is thus nigh imperceptible to the user.

To obtain the image, most sensors require a certain level of user cooperation. The user is asked to position themselves in front of the camera in order for it to be able to acquire the best possible images.

The majority of iris recognition systems work with a single acquired image. Although some obtain more and discard everything besides the one with the best quality, it is also possible to acquire two or more images.

Usually, iris recognition algorithms work with images with at least 100 pixels across the iris in order to get enough detail. However, there are several issues that might diminish the salience of the iris texture, such as partially closed eyelids, eyelashes obscuring the iris, inadequate lighting conditions, or excessively constricted or dilated pupil. Fortunately, most of these problems can be avoided with user cooperation [6].

In addition, there are illnesses and conditions which can severely hamper, if not even prevent, the enrolment of a user. Such conditions include glaucoma and cataract.

Glaucoma is the name for a group of ocular disorders that arise from ocular nerve damage primarily caused by abnormal pressure in the eye. While glaucoma is generally connected with vision impairment, not all types of glaucoma damage the structure of the iris.

Cataract is a term for clouding of the lens. It is caused mainly by aging and considerably degrades vision quality. Also, its effects significantly alter the structure of the pupil and iris. In advanced stages, the iris might become almost nonexistent.

Both conditions are displayed in Figure 3.4.

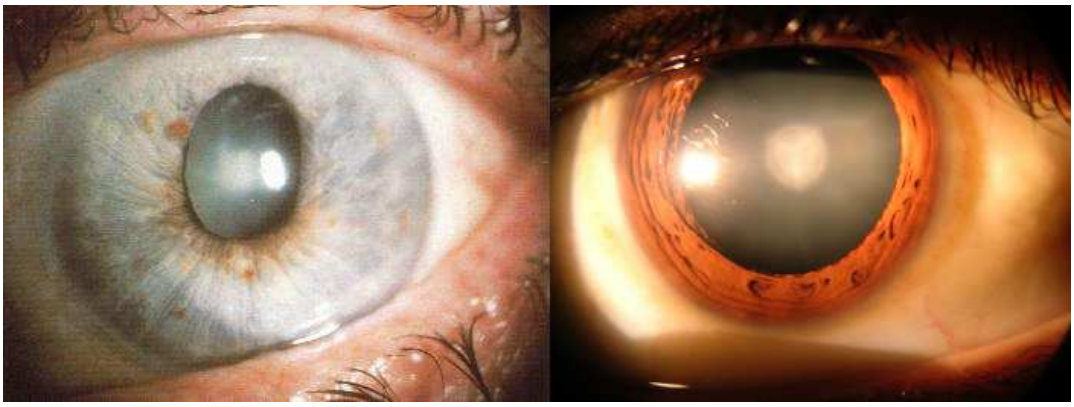


Figure 3.4: Eye diseases (acute glaucoma – left, cataract – right) impairing the possibility of enrolment. [48, 49]

Most cameras used for iris recognition are designed to capture the image from close distance. For this purpose, there are several commercial scanners available. However, systems that scan irises of moving people from the distance exist as well.

This kind of iris biometric usage presents a considerable challenge, as there are several problems that need to be solved. This technology is still under development and requires a certain level of user cooperation.

Unlike in the case of face recognition, which is deemed the primary biometric when moving individuals are to be scanned, the cameras need to be specially equipped. As mentioned above, near-infrared illumination is required. Because the eye is a relatively small object, the camera has to be adjusted accordingly and has to contain a lens with long focal length.

3.1.2.2 Segmentation

Once the image of the eye is acquired, the system can proceed with segmentation. Unfortunately, this part of processing is complicated for numerous reasons [6]. The iris texture exhibits a great degree of irregularity and varies substantially from person to person.

The iris can be considered stochastic texture with edge-like features randomly distributed across its surface. Because of this, it is difficult to model the iris based on its visual appearance for localization.

As mentioned before, the iris is annular in shape, delimited by the pupil on the inner boundary, and the sclera and eyelids on its outer boundary. In some cases, the change of intensity along the boundaries, especially the outer one, might not be very prominent. Wrong estimation of the boundaries can lead to incorrect localization of the iris and thus significantly skew the comparison at the end of the entire recognition process.

Furthermore, eyelids might drastically reduce the visible portion of the iris. They create an additional, irregular border that can diminish the efficiency of a segmentation algorithm. Eyelashes occluding the iris also contribute to this problem, as they can be much harder to detect, depending on the contrast and quality of the acquired image.

Most common approaches to iris segmentation are based on boundary detection. They rely on two facts:

- The iris and the pupil can be estimated with circles.
- The change of intensity along the boundaries is perceptible enough.

There are two main algorithms that are associated with segmentation of the iris [6]:

- integro-differential operator
- geodesic active contours

The *integro-differential operator* is defined as follows

$$\max(r, x_0, y_0)\left|G\sigma(r) * \frac{\partial}{\partial r}\oint_{r,x_0,y_0}\frac{I(x,y)}{2\pi r}ds\right| \tag{3.1}$$

where I is the input image, $I(x, y)$ is the pixel intensity. The image is convoluted with a radial Gaussian filter $G_\sigma(r)$ of scale σ and radius r. The purpose of the filter is to smooth the image in order to blur the crypts, furrows, and other artefacts found in the structure of the iris.

After filtering, the operator calculates the intensity gradient of the pixels located on the circumference of the circle defined by radius r and centre at (x_0, y_0). For each pixel, the gradient is computed along the line connecting it to the centre of the circle (denoted by the differential operator).

Then, the sum of the gradient values, normalized by the factor $2\pi r$, is calculated, which is denoted by the integral operator. The parameters r, x_0, y_0 that result in the maximum sum are chosen as the circular contour of the pupil boundary.

The operator is then used to detect the limbus boundary in the same way. However, the fact that the eyelids can interfere with the circle has to be taken into account, with emphasis on the pixels located on the horizontal diameter. The eyelids can be detected by searching for a parabolic edge within the area defined by the outer boundary. Also, the eyelashes can be detected by seeking near-vertical edges in the iris.

It is worth noting that the iris and the pupil are rarely perfectly concentric. In most cases, the pupil centre is actually situated closer to the nose in comparison to the centre of the iris. Additionally, both the iris and the pupil are not always exactly circular in shape. Therefore, the operator may need to be adjusted to an elliptical variant.

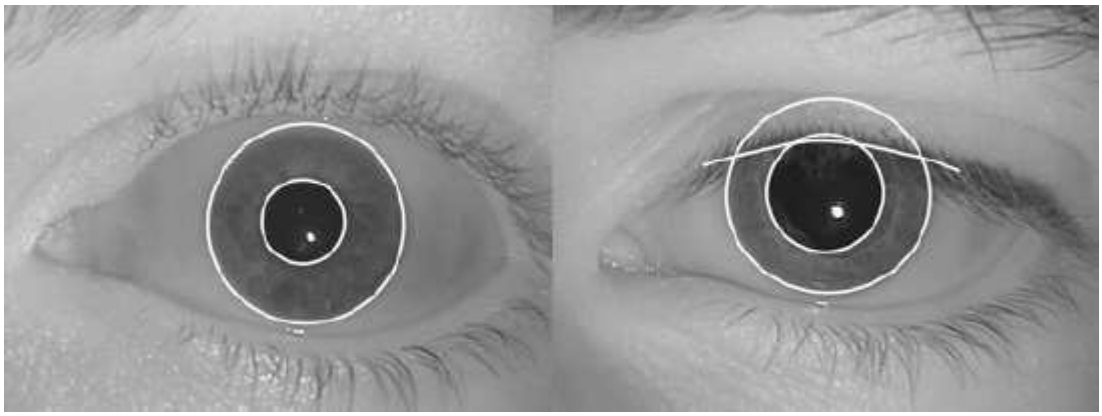


Figure 3.5: Iris segmentation using the integro-differential operator [6].

Another approach of iris segmentation uses *Geodesic Active Contours* [6]. While the integro-differential operator relies on estimating the iris with circles, the GAC approach can deal with irregular edges. The integro-differential operator detects the iris and the eyelids have to be edited out afterwards, but with the GAC, the iris can be demarcated by a single, special contour.

This method is based on the relation between the active contours and the calculation of minimal length curves. Its aim is to evolve an arbitrarily initialized curve from within the iris and perform adjustments according to the geometric properties of the iris boundary.

First, the curve $\gamma(t)$ has to be defined as a curve that gravitates towards the outer edge of the iris at a certain time t that corresponds with the number of iteration. Then, let ψ be the function that measures the distance from the curve $\gamma(t)$.

$$\psi(x,y) = \begin{cases} 0 & if\ (x,y) \text{ is on the curve} \\ < 0 & if\ (x,y) \text{ is inside the curve} \\ > 0 & if\ (x,y) \text{ is outside the curve} \end{cases} \tag{3.2}$$

The function ψ is of the same dimension as the image $I(x, y)$. The curve $\gamma(t)$ is called the level set of the function ψ, which means the set of all points in ψ where ψ is a constant. For instance, $\psi = 1$ is the first level set and $\psi = 2$ is the second level set.

ψ is called the embedding function because it embeds the evolution of $\gamma(t)$. The evolution of the embedding function is influenced by image gradient to ensure it approaches the real boundary of the iris. The initial curve $\gamma(t)$ is generally designed as a circle close to the pupillary edge.

The evolution of the embedding function can be discretized to the following equation:

$$\frac{\psi_{i,j}^{t+1} - \psi_{i,j}^{t}}{\Delta t} = -cK'_{i,j}\left\|\nabla\psi^{t}\right\| - K'_{i,j}\left(\varepsilon K_{i,j}^{t}\left\|\nabla\psi^{t}\right\|\right) + \psi_{i,j}^{t}.\nabla K''_{i,j}^{t} \tag{3.3}$$

where Δt is the time step. The first term is the velocity term that acts as an inflation force. The second term is the curvature. The third term is a discretization of central differences. c and ε are constants. ∇ is the gradient operator. K is the stopping function, given by

$$K(x,y) = \frac{1}{1+\left(\frac{\left\|\nabla(G(x,y)*I(x,y))\right\|}{k}\right)\alpha} \tag{3.4}$$

where $I(x, y)$ is the image and $G(x, y)$ is a Gaussian filter, and α and k are constants. The purpose of the stopping function is to provide a stopping criterion for the evolution of the curvature. It slows down the process of changing shape of the curve of until it settles.

However, in certain cases when the stopping criterion is not sufficient enough, this may lead to over-evolution of the contour. This can be mitigated by calculating the difference in energy between two successive contours and using a threshold. If the difference does not exceed the given threshold, the process is halted and the last result defines the final shape of the contour.

Because of the potential presence of visible edges within the iris region, especially in the form of radial fibres and crypts, there arises the problem of local minima. However, it is possible for this method to overcome the issue by splitting at these minima and merging afterward.

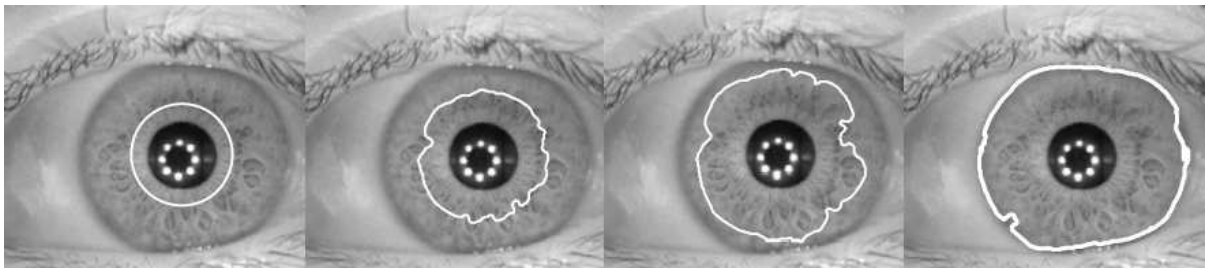


Figure 3.6: Iterative process of geodesic active contours method [6].

3.1.2.3 Normalization

After segmenting the image, the system proceeds to normalize the localized iris. The area of the iris texture is affected by several factors. Key amongst them is pupil contraction and dilation. Another factor is the resolution of the sensor. Furthermore, the size of the pupil and the iris, as well as the amount of occlusion caused by eyelids and eyelashes, varies across individuals.

In order to address these inconsistencies, the iris has to be unwrapped to a stripe, converted from Cartesian coordinate system to a polar coordinate system.

The annular shape is thus unrolled to a rectangular image. The rows correspond to the concentric regions of the iris. The polar coordinates are represented by the dimension r and θ, where $r \in [0, 1]$ and $\theta \in [0, 2\pi]$. The method is called Daugman's rubber sheet and is depicted in the following figure.

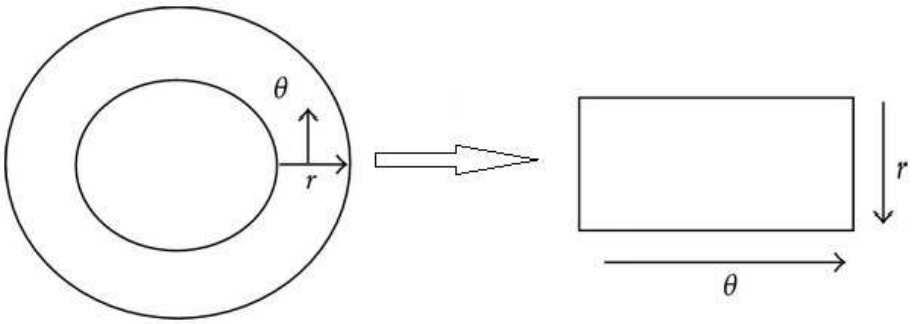


Figure 3.7: Daugman's rubber sheet model.

The conversion can be denoted as [50]:

$$I\left(x(r,\theta), y(r,\theta)\right) \rightarrow I\left((r,\theta)\right) \tag{3.5}$$

where

$$x(r,\theta) = (1-r)\, x_p(\theta) + r x_l(\theta) \tag{3.6}$$

$$y(r,\theta) = (1-r)\, y_p(\theta) + r y_l(\theta) \tag{3.7}$$

where x_p, y_p are the coordinates from the pupillary boundary and x_l, y_l are the coordinates from the limbus boundary. The segmented iris image can contain noise masks that filter out the occlusions caused by eyelids and eyelashes.

Because the GAC method does not usually produce regular shapes and because of the effect of the occlusions on the GAC scheme, only the points located on the boundary of the sclera and the iris are used for centre and radius estimation.

As an example, six points along the edge, close to the horizontal diameter but with different angles, can be selected. Their mean distance from the centre then represents the approximated radius of the iris.

Afterwards, a circle is fitted through the chosen points and the limbus boundary is thus estimated. Then, the system can proceed with the conversion to polar coordinates using Daugman's rubber sheet.

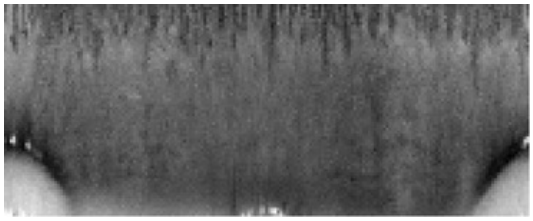


Figure 3.8: Unrolled iris stripe using the Daugman's rubber sheet method (with visible eyelid occlusion in the corners).

3.1.2.4 Encoding and Comparison

In order to acquire a feature set of the iris texture, the acquired sheet has to be encoded accordingly. To achieve this, a two-dimensional Gabor filter is used on the unrolled image. It is defined as follows [50]:

$$G(x,y) = e^{-\pi\left[(x-x_0)^2/\alpha^2+(y-y_0)^2/\beta^2\right]} e^{-2\pi i\left[u_o(x-x_0)\ +v_o(y-y_0)\right]} \tag{3.8}$$

where x_0, y_0 are the coordinates of a point in the image, α, β are the width and length of the image, and u_0, v_0 is the wave direction with a spatial frequency

$$\omega_0 = \sqrt{u_0^2 + v_0^2} \tag{3.9}$$

Separately, the real and imaginary components of the wavelet can be denoted as:

$$\Re\{G(x,y)\} = e^{-\pi\left[(x-x_0)^2/\alpha^2+(y-y_0)^2/\beta^2\right]} \cos\left(-2\pi i\left[u_o(x-x_0)\ +v_o(y-y_0)\right]\right) \tag{3.10}$$

$$\Im\{G(x,y)\} = e^{-\pi\left[(x-x_0)^2/\alpha^2+(y-y_0)^2/\beta^2\right]} \sin\left(-2\pi\left[u_o(x-x_0)\ +v_o(y-y_0)\right]\right) \tag{3.11}$$

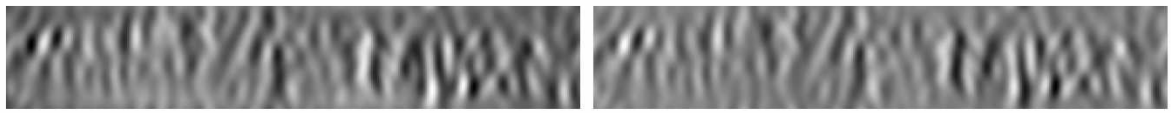


Figure 3.9: Real and imaginary outputs of an image after using a 2D Gabor wavelet [6].

After convoluting the unrolled iris texture with Gabor filter, the results are then demodulated in order to achieve data compression. This can be done by quantizing the phase information into four levels, each representing one quadrant of the complex plane.

Given the normalized iris image $I(\rho, \varphi)$, the demodulation can be denoted accordingly:

$$h_{\mathrm{Re,Im}} = sign_{\mathrm{Re,Im}} \int_\rho \int_\varphi I(\rho,\varphi)\ e^{-i\omega(\theta_0-\varphi)} e^{-(r_0-\rho)^2/\alpha^2} e^{-(\theta_0-\varphi)^2/\beta^2} \rho d\rho d\varphi \tag{3.12}$$

where $h_{Re,Im}$ is a complex value whose components are defined by the sign of the integral, α, β are the width and length of the image, ρ, φ are the polar coordinates, and r_0, θ_0 represent the centre frequency of the wavelet.

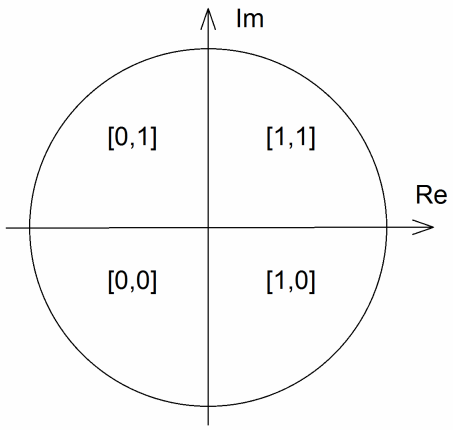


Figure 3.10: Four possible values of quantization.

The resulting size of the iris code depends on the resolution of the r and θ axes. Commonly, a length of 2048 bits is preferred.

In order to compare two feature sets obtained after applying the aforementioned procedure, Hamming distance is used [50]. Taking noise masks into account to filter out noisy regions, the Hamming distance can be calculated accordingly [6]:

$$HD = \frac{\left\| (IrisCodeA \oplus IrisCodeB) \cap MaskA \cap Mask \right\|}{\left\| MaskA \cap Mask \right\|} \tag{3.13}$$

The XOR operator is used to detect dissimilarities, while the AND operator is used to mask the noisy regions. The purpose of the denominator is to normalize the value so that it falls within the interval of <0, 1>. The lower the score, the more similar the two codes are.

Additionally, the two codes might have originated from two images captured at slightly different angles. Therefore, adjustments such as rotating the iris code may be needed in order to achieve the best performance.

Also, it is necessary to define an appropriate threshold for the Hamming distance during the decision process, which may vary across implementations of iris biometric systems. Numerous factors, such as the quality of the original images acquired by the sensor, can influence the optimum threshold.

The aforementioned algorithm is patented by John Daugman and is considered standard in iris biometrics. Although there are other approaches, none of them are as efficient as Daugman's algorithm.

Different methods include:

- zero-crossings of wavelet transformation
- neural networks
- 2D Hilbert transform

3.1.3 Performance

Iris recognition ranks amongst the most popular and robust approaches to biometrics and security. This owes to the fact that the complex structure of the iris provides recognition algorithms with rich detail.

Algorithm	FAR/FRR
Zero-crossings	0.03%/2.08%
Neural networks	0.02%/1.98%
2D Hilbert transform	1.84%/8.79%
Iris code	0.01%/0.09%

Table 3.1: Performance of selected iris recognition algorithms [51].

3.2 Retina

The retina is a layer located at the back of the eye. It is light-sensitive and covers more than half of the interior surface. It is composed of photosensitive cells – rods and cones. The rods, of which there are approximately 120 million, are more sensitive than the cones and are responsible for perception of light intensity. The cones, of which there are approximately 6 million, are responsible for colour perception.

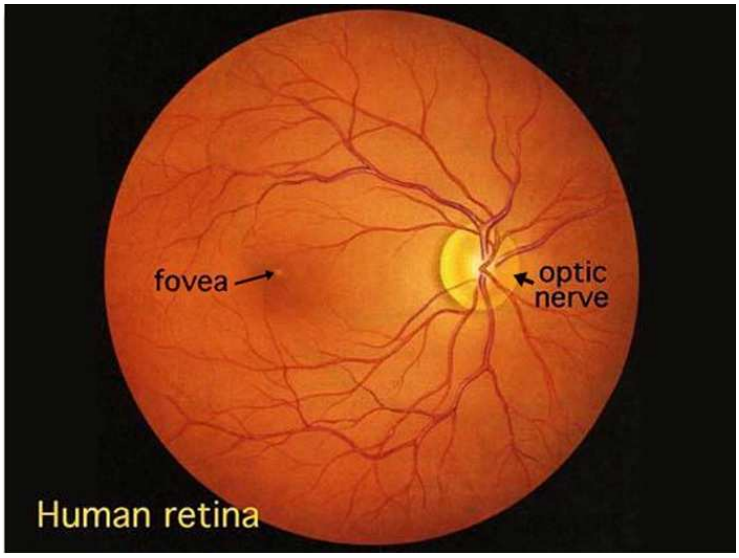


Figure 3.11: Human retina [52].

The retina is 0.5 millimetre thick. As is visible in Figure 3.11, there are two special areas on the retina. The first is the optic disc, where the ending of the optic nerve is located. It connects the eye with the brain and supplies the brain with information about the light and colour entering the eye.

From the optic nerve, arteries and veins radiate to the surface of the retina. The place where the optic nerve leaves the eye is called *blind spot* due to the absence of rods and cones and thus nonexistent sensitivity to light.

The second area of interest is called *fovea*. It contains a high concentration of cones, but there are fewer rods in comparison to the rest of the retina. This means the spot is very sensitive to colour but not light intensity. The fovea is located in the middle of the *macula*, which lacks blood vessels and is relatively dark in comparison to its surroundings.

As can be seen in Figure 3.11, the texture of the retina is nuanced, specifically due to the location of the optic nerve and the network of blood vessels originating from it. The vessels can be considered as a source of high-density, stochastic information, which can be used in biometric recognition.

3.2.1 Advantages and Disadvantages

Like the iris, retina is regarded as a reliable biometric due to its high level of detail and variability. It ranks amongst the most accurate biometrics and the recognition procedure is generally fast. Currently, there is no simple and definitive way of duplicating retinal information and spoofing a robust retinal sensor effectively.

Upon death or eye extraction, the cells of the eye begin to deteriorate rapidly, rendering useless the post-mortem usability of the eye for biometric recognition. Due to this, the possibility of impersonation is diminished and so is the requirement for liveness detection.

Also, because of the complexity of structure of the blood vessels, this biometric can be considered unique for each person. Even identical twins do not share an identical pattern.

Unlike iris recognition, retinal scanning is deemed intrusive. Some users believe the sensor technology to be harmful to the eye and therefore, they can show a certain level of reluctance in cooperation.

In addition, this biometric is not only regarded as invasive because of the aforementioned problem, but also because the retina can reveal the health condition of an individual to a certain degree.

The cost of this biometric system is very expensive. It is arguably one of the most expensive biometric technologies.

3.2.2 Retina Recognition

Although retina recognition follows the general schema of every biometric system, there is no preferred approach (unlike in the case of iris and Daugman's algorithm). The feature sets extracted differ across algorithms.

Possible methods of retina recognition can be based on:

- blood vessel bifurcation
- Fourier and/or wavelet transform

The mentioned approaches can be divided into modules:

- image acquisition
- normalization
- feature extraction
- comparison

3.2.2.1 Acquisition

The part that is common to all algorithms is the acquisition of the input image. The user is required to position their eye close to the camera in order to obtain a precise image of the retina. Occlusive objects such as glasses have to be removed to achieve best results.

For the purpose of retina recognition, infra-red light is utilized. Blood vessels absorb infra-red light, but the retinal background does not. Therefore, the blood vessels in the acquired image are in visible contrast with their surroundings.

Unlike in the case of iris recognition, there are no technologies that would allow scanning retinas from the distance without the knowledge of the scanned user. This is caused by the necessity of user cooperation.

The level of cooperation required of the user is higher than in the case of iris recognition. The user has to focus on the sensor, which then sends a beam of light on the pupil, illuminating the retina and capturing its image.

As was discussed in the earlier chapter concerning the iris, the problem with the inability of the user to enrol is present in retinal recognition as well.

Illnesses can severely affect the structure of the blood vessel network and the hue of the retina. Depending on the subsequently used algorithm, it may or may not be able to extract the features correctly. Thus, it could discriminate the potential user. Furthermore, this problem affects retina recognition more than it does iris recognition.

Conditions that alter the retina visibly include: glaucoma, diabetes, macular degeneration, and retinitis pigmentosa. While macular degeneration is usually caused by age and results in moderate vision loss, retinitis pigmentosa is a hereditary disease that can be responsible for blindness.

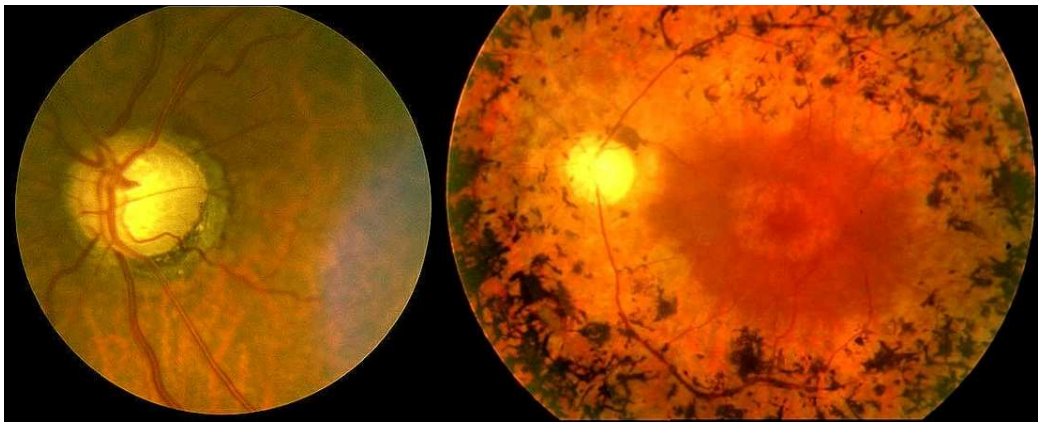


Figure 3.12: Retina affected by glaucoma (left) and by retinitis pigmentosa (right) [52].

Due to the nature of retinal scanning, it is necessary to deal with sources of light noise arising from corneal reflections and ambient illumination. This problem is often solved by sensors during image acquisition, so that the system does not have to filter and pre-process the image during later phases [52].

3.2.2.2 Normalization

Because the image acquired from the previous phase is centred on the retina, there is no need for a segmentation phase. However, algorithms that use the vascular structure usually resort to its segmentation.

First, the image may require normalization in order to deal with issues not resolved by the sensor. Such issues may include inadequate illumination, resolution difference, and rotation. While many sensors try to mitigate the first two problems to a substantial degree, rotation is often a problem that is left to the software side of the system.

The solution to different resolution lies in rescaling the image. However, the method relying on blood vessel bifurcation may not need a correction at this point, because the algorithms used during the comparison stage may anticipate this issue and solve it.

There are various techniques that deal with differences in illumination, depending on the level lighting of the image, such as histogram equalization.

Most methods will require rotation compensation in order to maximize the performance of the system. To do so, reference points have to be defined. For instance, the centre of the fovea and the optic nerve can be used, or the centre of the image and the optic nerve [54].

At first, the contrast is adjusted, given by the equation

$$J(x,y) = \left(J_{\max} - J_{\min}\right)\left(\frac{I(x,y) - \mathrm{I}_{\min}}{I_{\max} - I_{\min}}\right)^{\gamma} + J_{\min} \tag{3.14}$$

where $I(x, y)$ and $J(x, y)$ are the greyscale image representations before and after contrast adjustment. γ is a constant.

After stretching, the fovea can be located using grey level slicing and centroid calculation defined in the following equation [54]:

$$X_c = \frac{\sum M(x,y)x}{\sum M(x,y)} \qquad Y_c = \frac{\sum M(x,y)y}{\sum M(x,y)} \tag{3.15}$$

where

$$M(x,y) = \begin{cases} 1 & \text{if } \mathrm{I}_{\min} < I(x,y), \mathrm{I}_{\max} \wedge x_{\min} \le x \le x_{\max}, y_{\min} \le y \le y_{\max} \\ 0 & \text{otherwise} \end{cases} \tag{3.16}$$

I_{max} and I_{min} are the limits of the grey level. x_{max}, x_{min} are the limits in the x-axis and y_{max}, y_{min} are the limits in the y-axis.

Similarly, the optic nerve centre can be located by using an adjusted Equation 3.15, where the parameter $M(x, y)$ is given as

$$M(x,y) = \begin{cases} 1 & \text{if } \mathrm{I}_{\min} < I(x,y), \mathrm{I}_{\max} \wedge x > x_{fc}, R(x,y) > R_{ref} \\ 0 & \text{otherwise} \end{cases} \tag{3.17}$$

where x_{fc} is the x-axis value of the fovea centre point, $R(x, y)$ is the radial distance from the fovea, and R_{ref} is the reference radius.

Afterwards, the two acquired points can be used to align the images, correcting rotation and translation by measuring their distances and applying corresponding transforms.

3.2.2.3 Feature extraction and Comparison

At this point, the methods of extracting features from the retina image vary. An efficient approach to this is to use Fourier and wavelet transforms [55].

After normalization, Fourier transform is applied to the image, given by the equation

$$F(u,v) = \frac{1}{MN}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(x,y)e^{-j2\pi(ux/M + vy/N)} \tag{3.18}$$

where $f(x, y)$ is a function of image intensity, M and N are the image dimensions, and u and v are frequency variables. The phase angle and the spectrum are given by

$$\left|F(u,v)\right| = \left[R^2(u,v) + I^2(u,v)\right]^{1/2} \tag{3.19}$$

$$\phi(u,v) = tg^{-1}\left[\frac{I(u,v)}{R(u,v)}\right] \tag{3.20}$$

where $R(u, v)$ is the real and $I(u, v)$ is the imaginary part of $F(u, v)$.

Then, a method of partitioning is used to divide the Fourier spectrum to several half circles around the centre that include segments with the same area and degree arc. The pixels that are too distant from the centre do not contain any useful information and the pixels close to the centre contain only low-frequency information.

Due to spectrum symmetry, the partitioning can be reduced in order to avoid extracting redundant data.

After the partitioning process, the energy of each partition is calculated according to the equation

$$E = \sum_{x=l}^{L}\sum_{y=k}^{K}\left(F\left(x,y\right)\right)^2 \tag{3.21}$$

Before being stored as a part of the feature vector, the results are scaled in order ensure uniformity in comparison.

The next part of feature extraction is a k-th wavelet transform with low-pass and high-pass filtering as depicted in Figure 3.13. The input of each iteration, A_{k-1}, is the result of the previous iteration, starting with A_0, which is the original image.

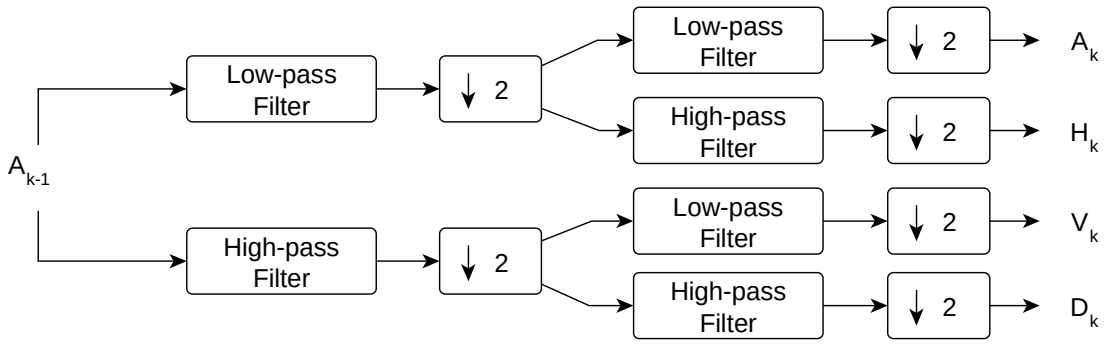


Figure 3.13: Wavelet transform with filters and subsampling.

Using the approximation, horizontal, vertical, and diagonal coefficients A_k H_k, V_k, and D_k k times, the original image is transformed into sub-images, from which the respective energies are calculated according to the following equations:

$$\begin{aligned} E_i^h &= \sum_{x=1}^{M}\sum_{y=1}^{N}\left(H_i\left(x,y\right)\right)^2 \\ E_i^v &= \sum_{x=1}^{M}\sum_{y=1}^{N}\left(V_i\left(x,y\right)\right)^2 \\ E_i^d &= \sum_{x=1}^{M}\sum_{y=1}^{N}\left(D_i\left(x,y\right)\right)^2 \end{aligned} \tag{3.22}$$

The energies obtained are added to the feature vector, resulting in a set that contains both the result of Fourier transform and wavelet transform. To compare two feature vectors, Eucleidan distance can be computed. If the system is working in identification mode, the template with minimum Eucleidan distance is selected on condition it passes a chosen threshold. If the system is working in authentication mode, a threshold is applied during the decision-making process.

Another approach to feature extraction is to use the blood vessels present in the retina. Before the blood vessels are segmented, filters are applied to the image in order to reduce noise and make the blood vessels stand out. Then, an edge-detecting filter can be applied in order to segment the blood vessels. Subsequently, the lines are thinned. In this image, bifurcation points are detected and stored as a feature vector.

There are numerous methods of each step of the aforementioned procedure. Several filters exist which can assist with blood vessel extraction, such as the Kirsch's templates, which convolve a rotating matrix with the image. Another method is to use two wavelet operations, given by the equations [56]:

$$W_{\psi}(b,\theta,a) = C_{\psi}^{-1/2} a^{-1} \int \psi^{*}\left(a^{-1} r_{-\theta}\left(x-b\right)\right) f\left(x\right) d^{2}x \tag{3.23}$$

$$M_{\psi}(b,a) = \max_{\theta} \left| W_{\psi}(b,\theta,a) \right| \tag{3.24}$$

where f is the function representing the finite energy of the image, ψ is the analyzing wavelet, C is the normalizing constant, b is the displacement vector, a is the dilation parameter, and θ is the rotation angle.

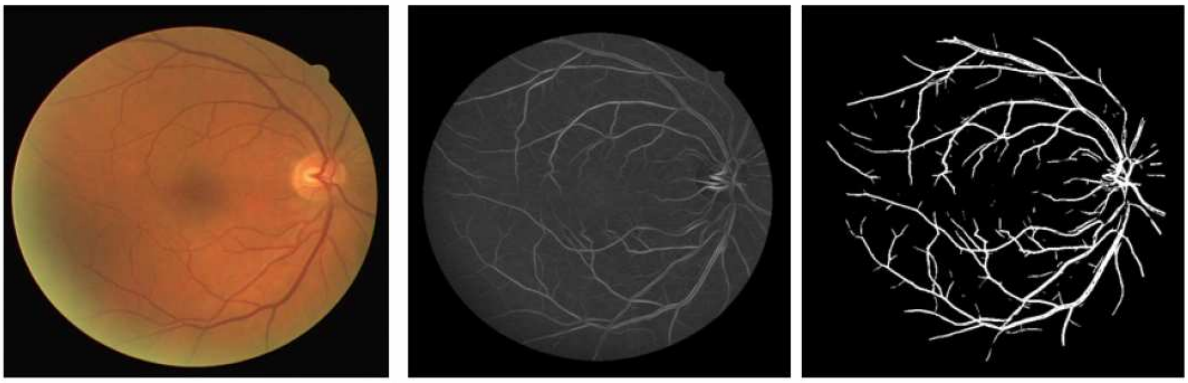


Figure 3.14: Vasculature segmentation using wavelets [56]. From left to right: original image, image enhanced by wavelet transform, image after thresholding.

After this step, the vascular pattern is enhanced for subsequent segmentation. This can be done with adaptive thresholding. Thinning of the vessels is achieved by applying a morphological thinning operator.

Once the blood vessels are thinned to the width of one pixel, vascular bifurcations can be extracted using the crossing number method [56]:

$$C(p) = \frac{1}{2} \sum_{i=1}^{8} \left| I_{thin}\left(p_{i \bmod 8}\right) - I_{thin}\left(p_{i-1}\right) \right| \tag{3.25}$$

where p_0 to p_7 denotes a clockwise sequence of pixels neighbouring the examined pixel p and I_{thin} is the thinned image. $C(p) = 1$ corresponds to vessel endings, while $C(p) = 3$ corresponds to vessel bifurcation.

Once bifurcation points are extracted, they are stored as a feature vector and can be used for comparison. If the bifurcations do not seem to match, they can be aligned by using the iterative closest point algorithm, which follows the sequence [1]:

1. For every point in the first feature set, select the closest point in the second feature set.
2. Compute the corresponding translation and rotation.
3. Perform the correcting translation and rotation.
4. Iterate until convergence.

3.2.3 Performance

Retinal recognition is regarded as secure because of the high level of detail in its texture, uniqueness of the vascular structure, and biological parameters that make it hard to duplicate. In the table shown below, performance results of two implementations of retinal biometric systems are presented.

Algorithm	Total Recognitions	Correct Recognitions	Incorrect Recognitions
Wavelet and Fourier Transform	360	357	3
Bifurcation	354	348	6

Table 3.2: Performance of retina recognition algorithms [55, 56].

4 Multibiometric System Design

The multibiometric system fuses two biometrics discussed at length in the previous chapter – iris and retina. The fusion of the system is performed at score level. The system adheres to the general schema of a multibiometric system, but it lacks sensor modules because the images have been acquired from existing biometric databases.

Since iris and retina biometrics are not biologically dependent, they do not need to belong to the same individual for the purpose of evaluation of this system. Therefore, the fact that each database originates from a different source cannot invalidate the results. However, the chosen pairing of iris and retinal samples is fixed and does not change.

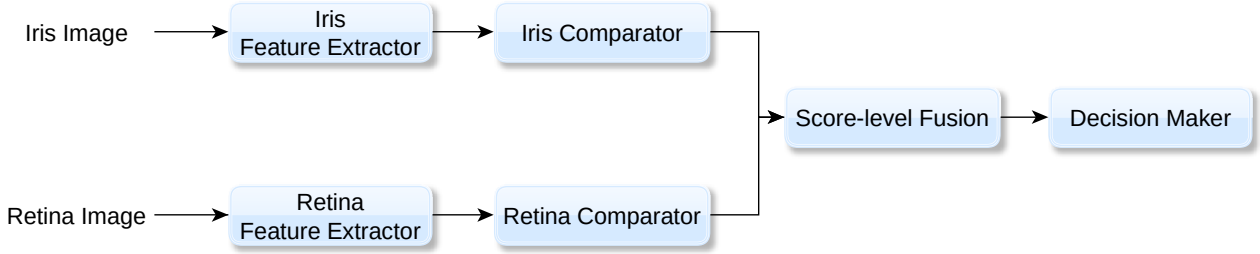


Figure 4.1: Components of the designed multibiometric system.

4.1 Iris

The iris extraction module is divided into sections as depicted in Figure 4.2.

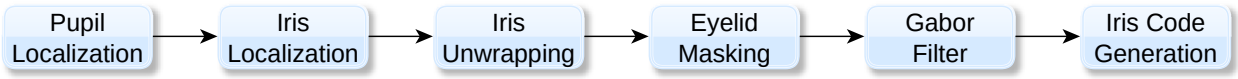


Figure 4.2: Iris feature extraction schema.

The method itself is based on Daugman's algorithm discussed in Chapter 3.1.2 and can be separated into parts:

- **Pupil localization** – First, median filter is applied to the image. The pupil is localized by thresholding a greyscale input image.
- **Iris localization** – The iris is localized by scanning for intensity shift along horizontal lines in the image.
- **Iris unwrapping** – Converting the image from Cartesian to polar coordinates is achieved with Daugman's rubber sheet.
- **Eyelid masking** – Eyelids are detected by scanning for high intensity differences alongside the corresponding border of the unwrapped image.
- **Gabor filter** – A 2D Gabor filter is applied to the image.
- **Iris code** – An iris code of 2048 bits is generated.

The comparison module takes two iris codes. To deal with possible rotational misalignment, one iris code representation is rotated column-wise and slightly adjusted Hamming distance is calculated for each rotation. The rotation with the best score is selected.

4.1.1 Feature Extraction

As the input database, CASIA-IrisV1 [4] was chosen. This database consists of 756 iris images from 108 distinct people. The database contains close-up greyscale pictures of varying quality. Some irises are clearly visible, while others are partially obscured by eyelids and eyelashes.

In this database, the detection of the pupil is relatively straightforward due to its contrast. There are almost no anomalies in the images in this aspect, no dots of light that would require additional attention.

Thus, the pupil can be localized by applying appropriate thresholding technique. First, however, median filter is applied to the image. This smoothes the image and eliminates pixels with outlying values, which further assists the segmentation procedure.

In order to determine the threshold value, the histogram of the image is calculated. In the lower half (the darker part) of the histogram, a pronounced peak can be found. This, together with the surrounding values, mainly denotes the pixels of the pupil.

The desired threshold is therefore to be found around this peak. The chosen threshold is higher than the value of the peak in the histogram to ensure the majority of the pixels of the pupils are included.

After thresholding is applied, the largest black area in the acquired image is bound to denote the pupil. Because it is elliptical in shape, detecting its centre and radius can be determined simply by seeking its widest areas.

The pupil itself is not entirely circular, but it can be estimated with a circle for the sake of simplicity and avoiding computational complexity.

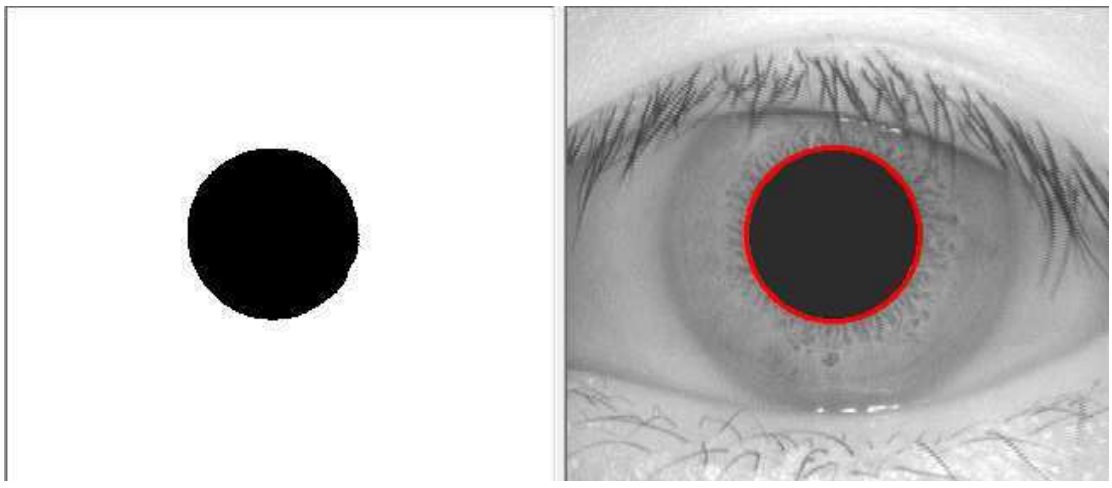


Figure 4.3: Segmented pupil.

Next, the iris must be segmented. While the pupil and its surroundings were separated by a striking shift in intensity, the same does not apply here. Therefore, a different approach must be adopted.

Although the shift in intensity is not as pronounced, it is present nevertheless. To help facilitate its detection, the contrast of the image is adjusted. Together with the median filter, this emphasises the current area of interest, which is the outer edge of the iris.

While not being as sharply defined as in the case of the pupil, it can be detected by searching for places where the pixel intensity transitions markedly over a certain distance. With this database, this achieves satisfying results, but this method is not always possible because of the quality of some pictures (e.g. pictures with the iris heavily concealed by the eyelids). To mitigate this issue, a failsafe method using 1D Haar Wavelets is employed.

Although the iris is not entirely circular as well, it is safe to estimate it by using a circle. Additionally, the iris and the pupil are not concentric, but to make the algorithm faster and simpler, it is assumed they are.

Combined with the detected points where the edge of the iris is located, the radius of the iris can be calculated and thus the segmentation of the iris is completed.

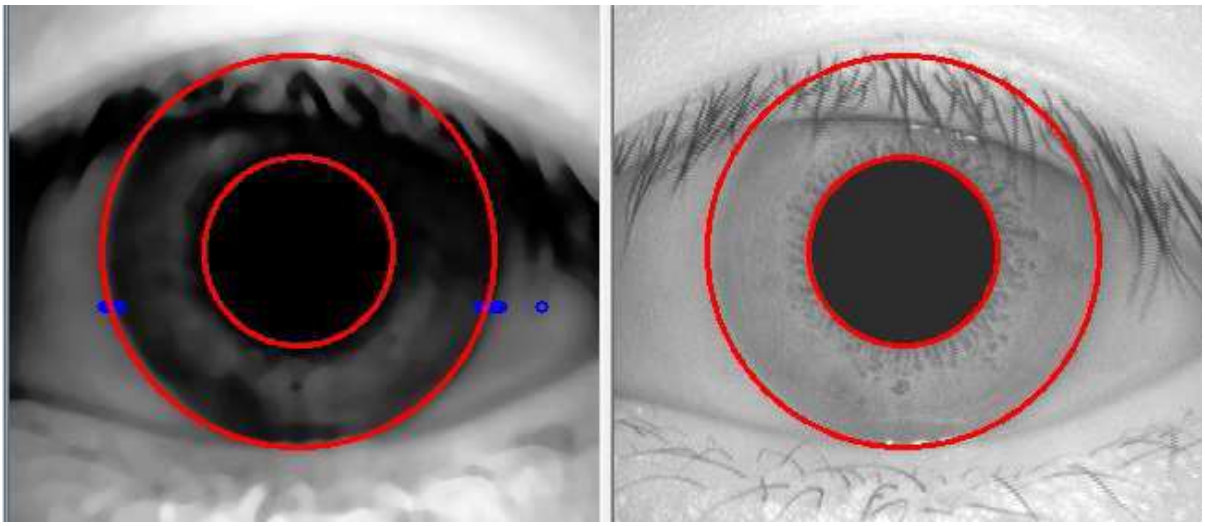


Figure 4.4: Segmented iris. The blue points denote the detected shifts of intensity.

The approach at this point varies, but in this algorithm, unrolling of the iris precedes the segmentation of the eyelids. For this purpose, Daugman's rubber sheet is used. At this point, the rubber sheet is reduced to an image with the width of 360 pixels.

Next, eyelids and eyelashes need to be filtered out. In this algorithm, a simple detector of change in the intensity along the border of the stripe is used. Given the fact that the rough location of the eyelids is predictable, the algorithm defines set boundaries within which the detection is performed.

Once the horizontal borders are determined, the algorithm similarly detects the height of the eyelid. When this is done, two masks in the form of tetragons are generated, which are to be taken into account during the final phase of feature extraction. Of course, this is true only if there are any eyelids.

Figure 4.5: Eyelid detection and mask generation. Yellow points denote the borders of the examined regions.

The proceeding step entails applying Gabor filter, which can be denoted by an alternate equation:

$$g_{\lambda,\theta,\varphi,\sigma,\gamma}(x,y) = e^{-\frac{(x\cos\theta+y\sin\theta)^2+\gamma^2(-x\sin\theta+y\cos\theta)}{2\sigma^2}} \cos\left(2\pi\frac{x\cos\theta+y\sin\theta}{\lambda}+\varphi\right) \quad (4.1)$$

with wavelength of $\lambda = 2$, orientation $\theta = 0°$, phase offset $\varphi = 0$ and aspect ratio $\gamma = 0$.

Before quantization, the question of reducing the amount of data has to be addressed. Because the stripe can be of varying height (depending on the radius of the iris), the answer to this problem also involves solving the issue of potentially different dimensions of the stripe.

In order to resolve this, certain angular and radial slices of the rubber band are selected for quantization, so that the expected feature vector is of the desired size. During this part, it is necessary to take the eyelid mask into account and map it in accordance with the feature vector.

The quantization itself is achieved by using cosine and sine filters. The values obtained are depicted in Figure 3.10. After this procedure, a feature vector of 2048 bits is obtained. This represents a 128x8 pixel encoding of the iris stripe, with one real and one imaginary bit for every pixel.

At this point, the feature vector is complemented by a corresponding mask.

4.1.2 Feature Comparison

Unlike the retinal part, the iris feature comparison is relatively simple. Given two feature vectors, XOR operator is applied to the corresponding bits. In this algorithm, if the values are equal, similarity score is incremented.

The masks of the respective vectors are used to filter out pixels that do not include the iris. This reduces the number of compared pixels. Thus, the resulting score is normalized so that it fits within the interval between 0 and 1.

$$score = \frac{score}{2048 - maskedBits} \quad (4.2)$$

Because of potential differences in rotation of the input image, which was neglected in the feature extraction phase, the score is calculated for differing angles ranging approximately from -16 to 16 degrees. The highest score is then selected as the final score and passed further for the decision making process.

4.2 Retina

The retina extraction module is based on vascular bifurcation explained in Chapter 3.2.2. Its schema is shown below:

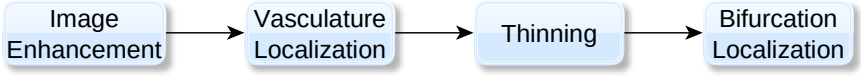


Figure 4.6: Retina feature extraction schema.

The procedure is split into four sections:

- **Image enhancement** – The input image is enhanced by using smoothing filters in order to reduce noise and make the vasculature more visible.
- **Vasculature localization** – Difference between the filtered image and the original, along with adaptive thresholding is used to segment the blood vessels.
- **Thinning** – Zhang-Shuen thinning algorithm is applied to reduce the width of blood vessels to one pixel.
- **Bifurcation localization** – Bifurcation points are localized by analyzing pixel neighbourhood and the resulting points are stored in a feature vector.

The comparison module takes two feature vectors. The position of the optic disc is utilized to align the bifurcations. The adjusted vectors are compared and score is calculated in accordance with level of similarity. The score is normalized so that it falls within the interval of <0, 1>, where higher value indicates a better match.

4.2.1 Feature Extraction

The database used was kindly supplied by STRaDe group at Faculty of Information Technology of Brno University of Technology. It contains 684 images of both retinas from 110 distinct people, totalling 220 distinct samples. The images are of high resolution but varying quality. Unlike the iris images, they are not greyscale.

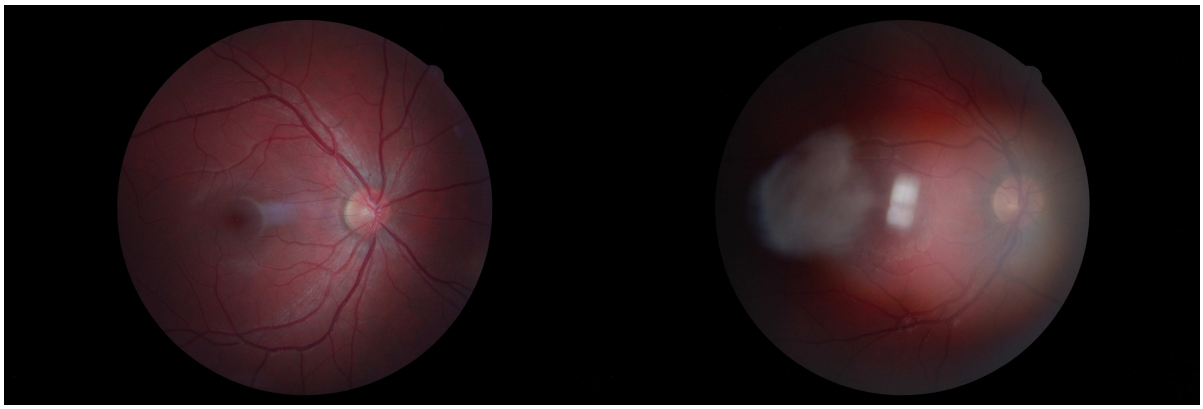


Figure 4.7: Sample images from the retina database.

In order to obtain the most salient information from the image, the green channel is selected for further processing. In this channel, the vascular structure is even more prominent than in a regular greyscale image.

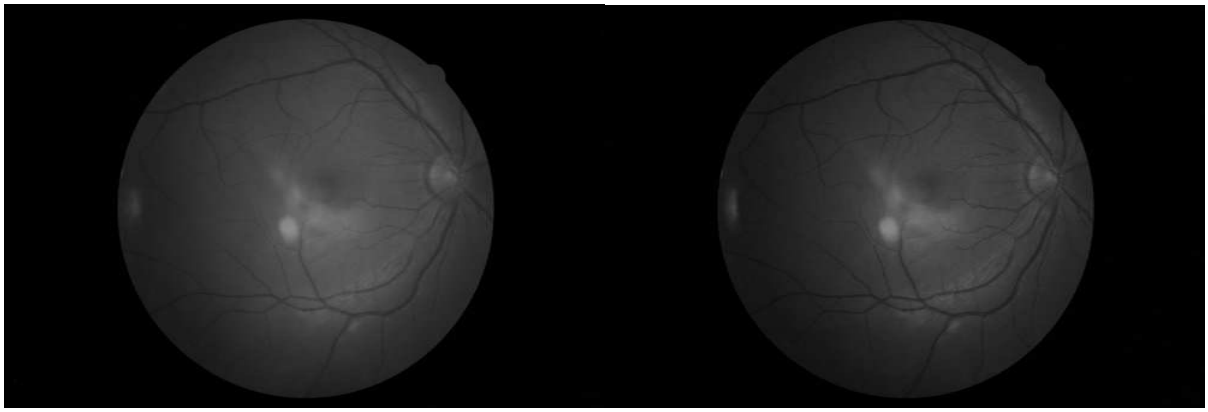


Figure 4.8: Grayscale (left) and green-channel image (right). The green-channel image contains a more pronounced vascular structure.

Next, the contrast of the green-channel image is adjusted by using Contrast limited adaptive histogram equalization algorithm. This algorithm differs from simple histogram equalization by calculating histograms for partitions of the image. The transformation of every pixel is derived from its neighbourhood. Contrast limiting is added in order to reduce the amplification of noise inherent to adaptive histogram equalization [57].

The preceding step to segmenting the veins is the application of median and blurring filters. This produces a relatively smooth image which is then compared to the non-filtered one. The differential image that is the result of this comparison is calculated according to the following equation:

$$diff\,(x,y) = \frac{255}{\max}\left(original\,(x,y) - filtered\,(x,y)\right) \tag{4.3}$$

where *max* is the maximum value of intensity difference of the pixels.

Although the vascular structure is visible at this point, there is a significant level of noise and the veins need to be segmented perfectly. To process the image further, adaptive thresholding is used. This highlights not only the retinal veins but also the noise, which must be removed before bifurcations can be detected. This is achieved by filtering out blobs and by morphological dilatation of the image.

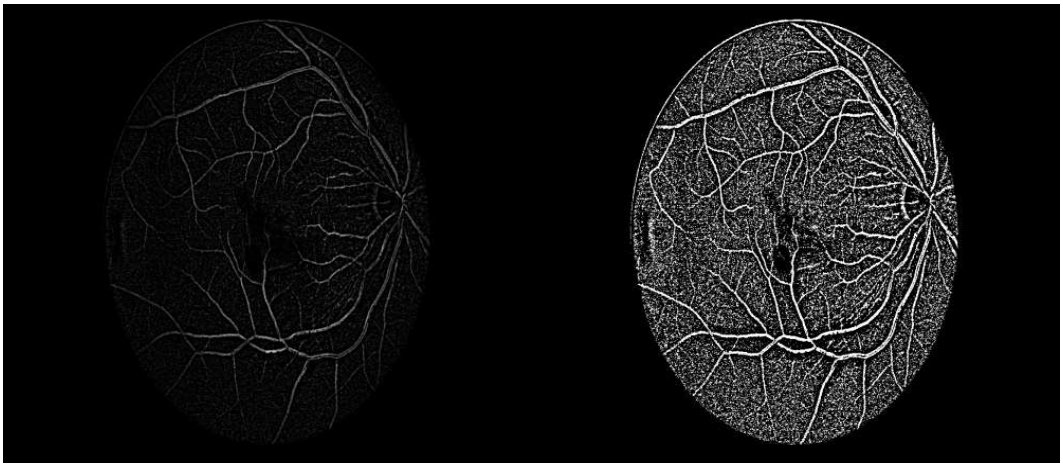


Figure 4.9: Differential image (left) and image after the application of adaptive thresholding (right).

The last step in blood vessel segmentation is thinning. For this purpose, Zhang-Suen algorithm is used. The algorithm is described accordingly [58]:

1. While points are deleted do
   a. For all pixels $p(i,j)$ do:
      i. If $2 \le B(P_1) \le 6$

         $A(P_1) = 1$

         $P_2 \times P_4 \times P_6 = 0$ in odd iterations, $P_2 \times P_4 \times P_8 = 0$ in even iterations

         $P_4 \times P_6 \times P_8 = 0$ in odd iterations, $P_2 \times P_6 \times P_8 = 0$ in even iterations

      ii. Then delete pixel $p(i,j)$

Where $A(P_1)$ is the number of 0 to 1 transitions in a clockwise direction from P_9 back to itself, and $B(P_1)$ is the number of non-zero neighbours of P_1.

Figure 4.10: Blood vessels before and after applying Zhang-Suen thinning algorithm.

The bifurcations are obtained by evaluating every white pixel and its immediate neighbourhood. If a bifurcation is to be marked, there need to be at least three separate paths that converge at a given pixel.

To calculate this, the neighbourhood is analyzed for the number of white pixels and their continuity. If three or more separate white areas are detected, the algorithm regards the pixel as a bifurcation.

This way, the algorithm acquires a list of bifurcations of the blood vessels and stores them as a list of points. In case the thinning yielded an imperfect image with clustered bifurcations, the algorithm filters out these bifurcations.

To be able to align the images during comparison phase, the optic disc must be located. For this purpose, the red channel of the original image is used, because the optic disc is most prominent there. The image is then processed by CLAHE and median filter.

In this image, Canny edge detection is performed and Hough circle transform [59] is used to detect the optic disc. Along with the bifurcations, the optic disc is stored within the feature vector for comparison.

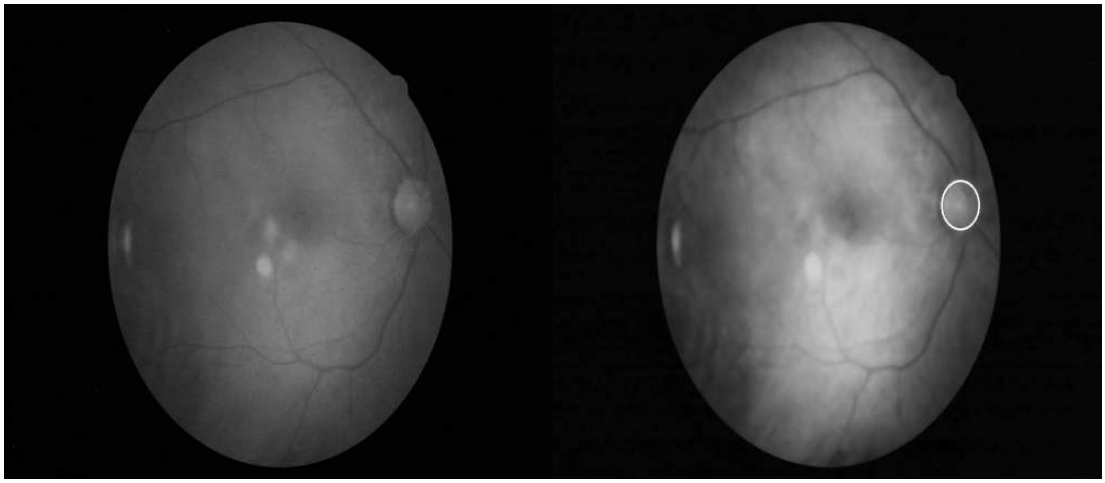


Figure 4.11: Red channel of the retina image (left), filtered image with optic disc detected (right).

4.2.2 Feature Comparison

First, the two vectors have to be aligned before comparison. This is achieved by taking the optic disc centres and translating the bifurcation points of one image. Because the rotation of the images is minimal, only translation is taken into account.

Next, the similarity score has to be calculated. The algorithm is as follows:

1. For every bifurcation point b_1 in the smaller array of bifurcations *B1*
    a. For every bifurcation non-matched point b_2 in the larger array of bifurcations *B2*
        i. If Euclidean distance between b_1 and b_2 is shorter than *threshold* and is currently the shortest, mark b_2 as selected
    b. If there is a match, mark selected b_2 as matched and increment number of matched bifurcations *n*
2. Calculate score

Then the score is obtained accordingly:

$$score = \frac{2n}{|B1| + |B2|} \tag{4.4}$$

Like the iris, this score is also normalized to fit between the interval of 0 and 1 for easier integration during the fusion phase.

4.3 Fusion

Fusion of this biometric system is performed at score level, using transformation-based methods discussed in Chapter 2.2.7. Because iris and retina are statistically independent biometrics (there is not correlation between them), it is justifiable to use different databases for iris and for retina respectively.

However, given the fact that the two databases differ in the number of images and people, they were merged and the people and the images were paired.

The methods used for transformation are:

- product rule
- sum rule
- max rule
- min rule

The median rule is excluded because it would not make sense to calculate the median value of two biometrics. Because of the ease of implementation, AND/OR fusion at decision level is also incorporated, though merely for informative purposes.

Although the scores from the previous blocks of the system were normalized to fit within the interval of 0 and 1, the distribution of the scores can be uneven (eg. threshold value of 0.5 could be too strict for the iris but too lenient for the retina), further normalization is required.

This is achieved by determining the minimum and maximum values of each biometric separately. After acquiring these values, the scores are adjusted accordingly and the fusion is performed.

4.4 Interface

The designed application contains a graphical user interface that lets the user work with the multibiometric system. Before system evaluation is executed, an introductory dialogue window is shown, which allows selecting databases for evaluation or loading a previously saved result file.

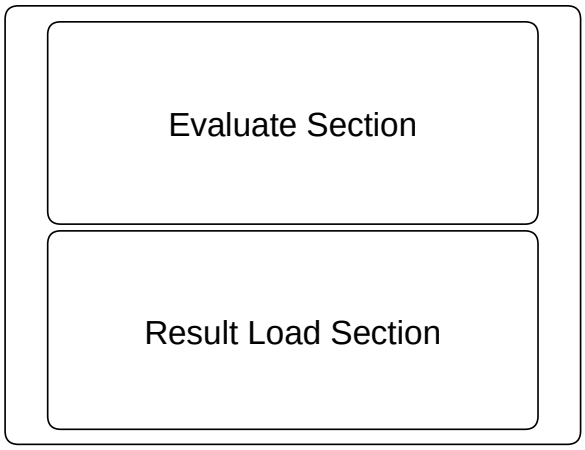


Figure 4.12: Structure of the introductory screen.

In the evaluate section, the user can select the path to the iris and retina databases and can specify the types of image and the way the images are identified. This helps classify the images so that the system knows what images belong to what person.

Because the system has to perform more than ten thousand comparisons and because the user may be forced to wait due to this, the user has the option to load previously stored results. Merely the path to them needs to be specified.

After the system is evaluated, the user is presented with results of the process and can peruse them. The user interface is split into tabs that offer a range of information concerning the efficiency of the system, such as genuine and impostor distribution, performance statistics, and ROC curve. The interface also allows switching views between the methods of fusion.

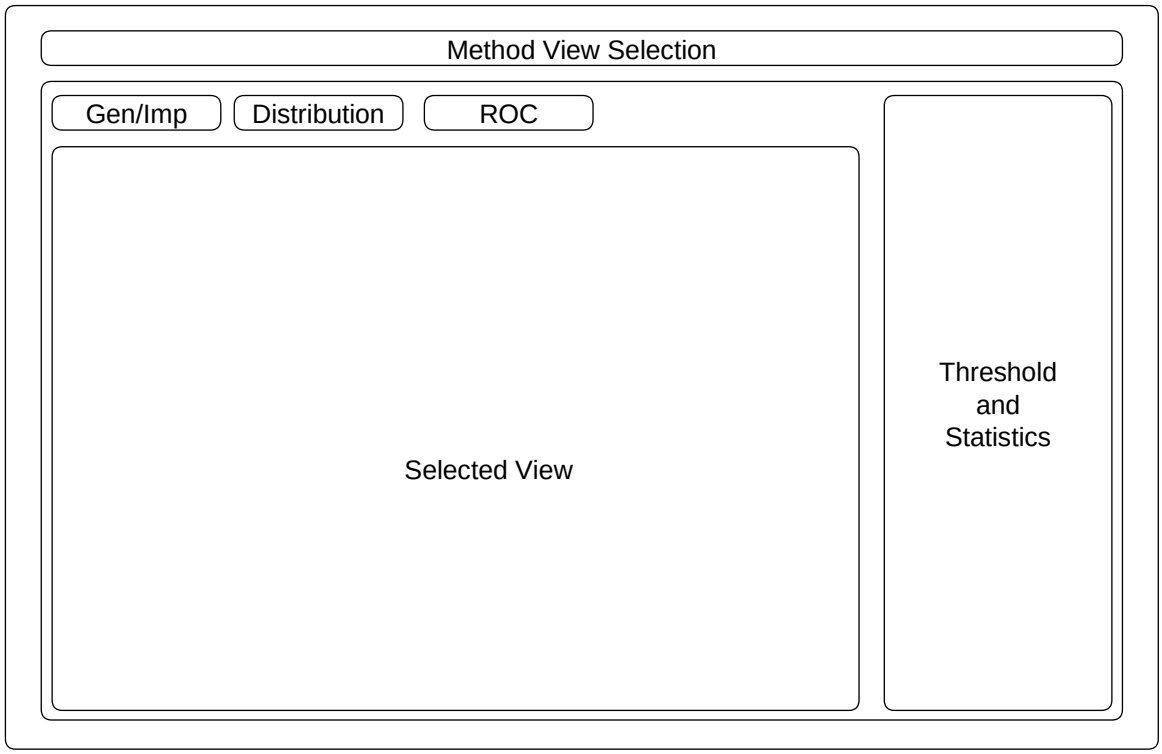


Figure 4.13: Once evaluation is done, a screen with performance results is shown.

The method view selection allows the user to switch between views. This groups together three views:

- separate iris and retina view
- min and max rule view
- product and sum rule view

Each view has its separate graphs that display the evaluation results of the biometric system. Also, the user can set the threshold of the biometrics. The user can see what the results would be before the fusion and after it.

The statistics contain FMR, FNMR, false match count, false non-match count, true match count, and true non-match count.

The form can be closed, at which point the introductory dialogue window is shown again.

5 Multibiometric System Implementation

While the previous chapter detailed the concept of the multibiometric system and its core functions, this chapter lists the technologies that were utilized to develop the system and explains the implementation of said functions.

5.1 Technologies

The biometric system was developed in Microsoft Visual Studio 2010, which is a platform for developing programs for Microsoft Windows, encompassing several programming languages such as C# or Visual Basic.

This platform incorporates .NET technologies shared by its languages. It also offers many tools which made the development of the biometric system easier. Its upside is the form designer, which can be utilized to create forms quickly and efficiently, so that the programmer can focus on the backbone of the application.

The system was programmed in C#, which is an object-oriented programming language. The language was selected because of its abilities to handle the compartmentalization of the biometric system.

The system utilizes a wide range of computer vision and image processing operations and for this purpose, the OpenCV library was chosen. OpenCV is an open-source library for C and C++ containing several toolkits. It is especially optimized for high performance, which is an important aspect in biometric systems.

Because the OpenCV is aimed primarily at C++, the biometric system employs a library for C#, called Emgu CV, which is a cross-platform wrapper for .NET compatible languages. Although a wrapper, it also allows direct invocation of the OpenCV functions.

These libraries were used for most filters, such as median and blur filters, for CLAHE, image handling, etc.

As an auxiliary library for image processing, AForge.NET was used. It is a library dedicated to artificial intelligence and computer vision, originally developed for the .NET Framework. Within the biometric system, this library provided advanced filters and blob detection.

For visualisation of the data, ZedGraph was chosen. This library is designed for plotting graphs in form or web applications in .NET technologies. It offers a broad range of configuration capabilities, but is also aiming for ease of use.

5.2 Program

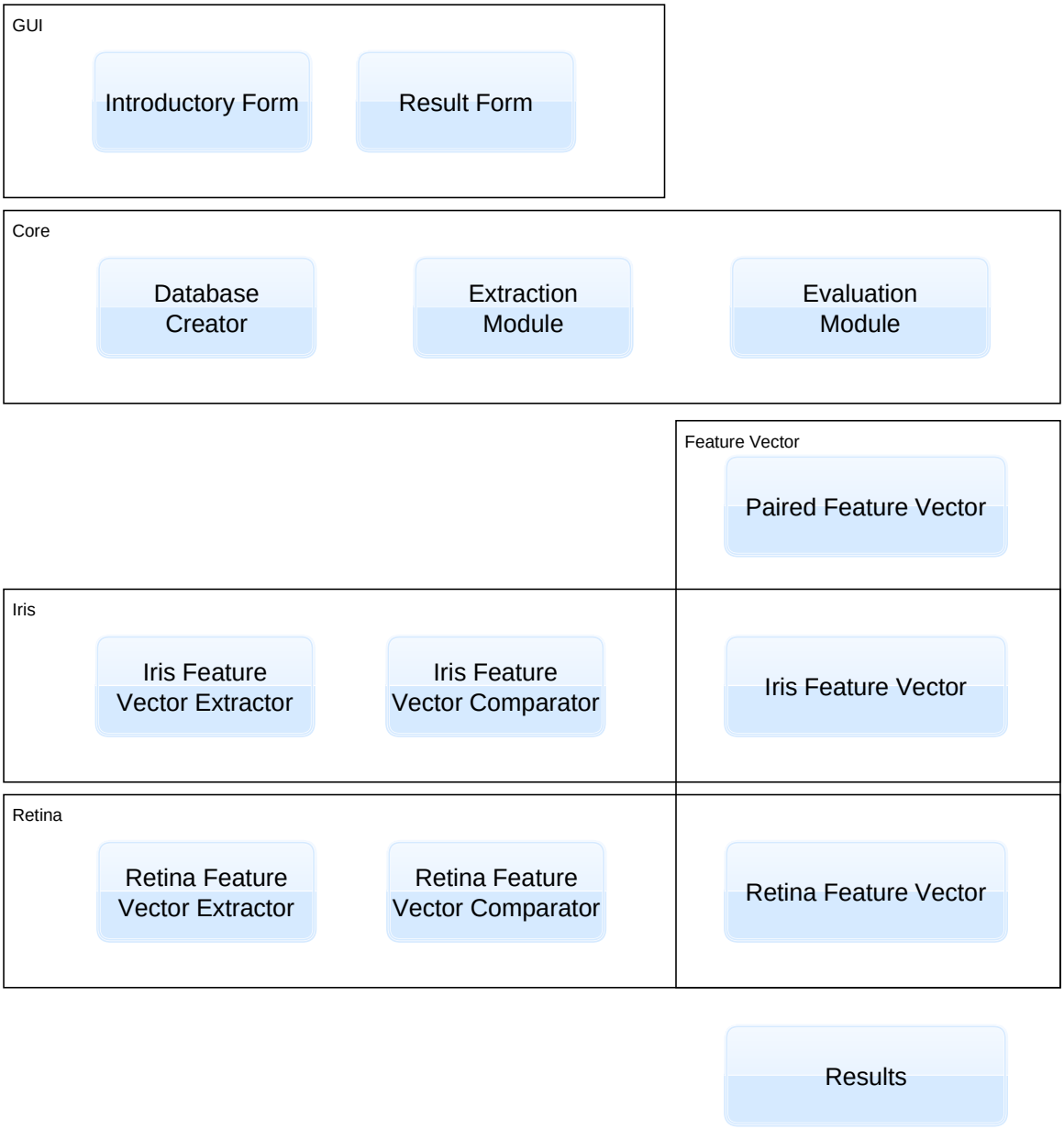


Figure 5.1: Simplified diagram of the classes of the biometric system.

Because the technicalities of the program are many, the presented implementation is simplified and unimportant details are omitted.

The `Introductory Form` class entails the graphical interface and the underlying interfaces of the first screen of the application. It contains the controls that allow the user to select and set the databases for evaluation.

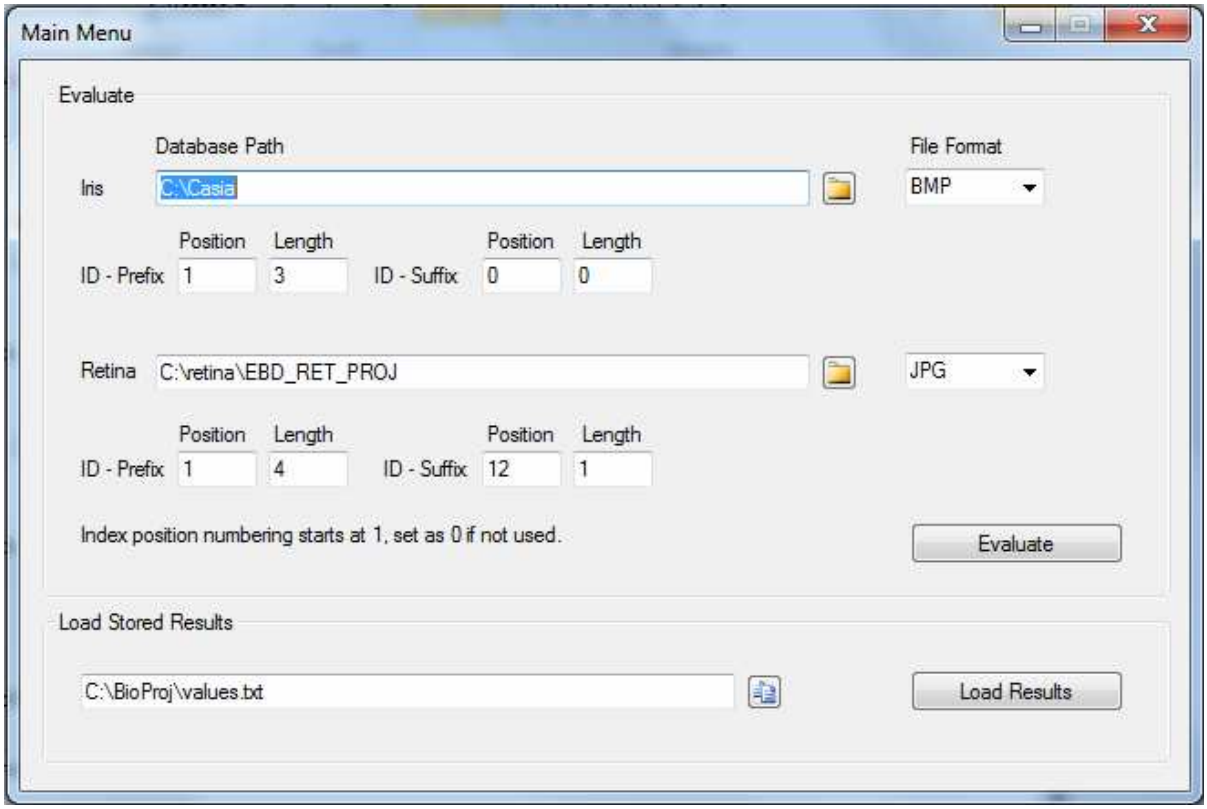


Figure 5.2: Introduction screen.

This class is connected to the `Database Creator`, `Extraction Module` and `Evaluation Module` classes, which form a bridge between this form and the result form.

The `Database Creator` is responsible for handling the databases. It reads the contents of the folders where the databases are located. Then it proceeds to create pairs of iris and retina images, which remain fixed throughout the entire evaluation process.

The results of this operation are passed to the `Extraction Module`, which, as the name suggests, extracts the features from the images and stores them as feature vectors (`Iris Feature Vector` and `Retina Feature Vector`).

For the extraction, `Iris Feature Vector Extractor` and `Retina Feature Vector Extractor` are used. The two classes contain methods for feature vector extractions as explained in the design chapter.

The structure of the feature vectors reflects the design described earlier. The `Iris Feature Vector` class contains a code of 2048 bits and a corresponding mask. The `Retina Feature Vector` class contains an array of bifurcations and the coordinates of the optic disc centre.

Because these vectors belong in pairs, they are saved as such using the `Paired Feature Vector`. The results of this stage are stored in a two arrays of this class, one for templates and one for vectors meant for comparison with the templates.

These arrays are then processed by the `Evaluation Module` class, which is responsible for comparing each vector from one array with each vector from the other array. To obtain the results, `Iris Feature Vector Comparator` and `Retina Feature Vector Comparator` classes are used, functioning as detailed in the design chapter. Iris comparison involves using the binary XOR operation, retina comparison involves matching bifurcations.

After this part, the system obtains the comparison results, which are stored as instances of the `Results` class. The `Evaluation Module` then performs fusion at score level using the rules specified previously and adds the results to their respective `Results`.

Once this is completed, the results are passed to the `Result Form`, which is responsible for displaying them in the form of graphs and statistics. Both the performance of the chosen biometrics separately and the performance of the fused biometrics is shown. The controls on the screens, such as threshold sliders, are present for assessing purposes. Views are available as mentioned in the design chapter as well.

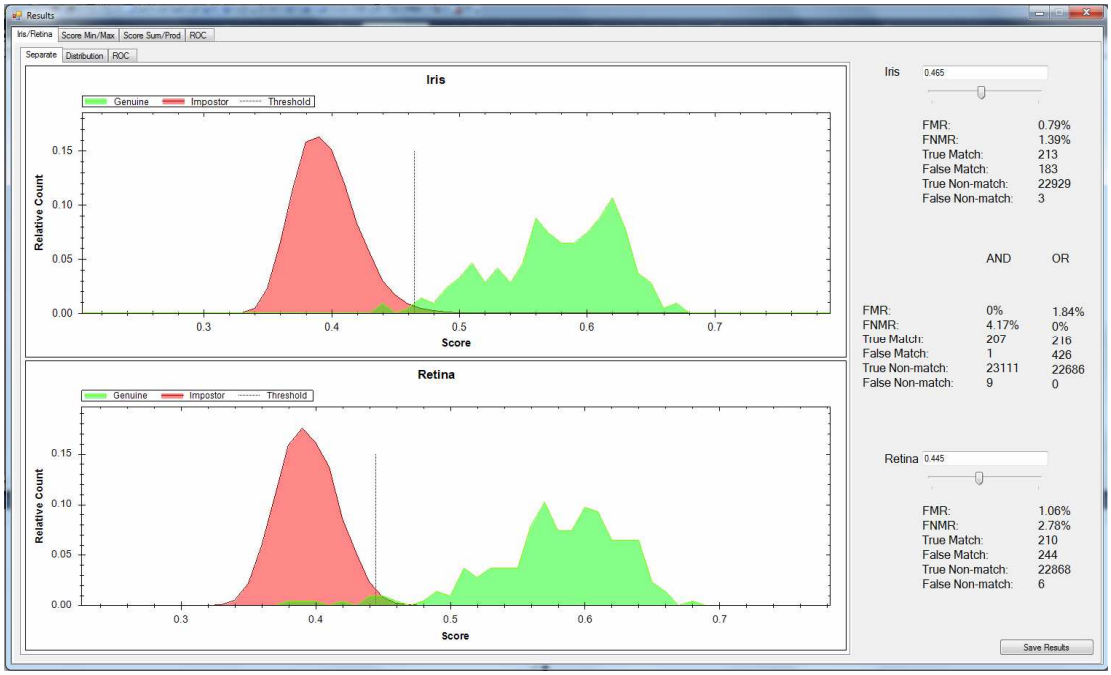


Figure 5.3: View of fusion performances.

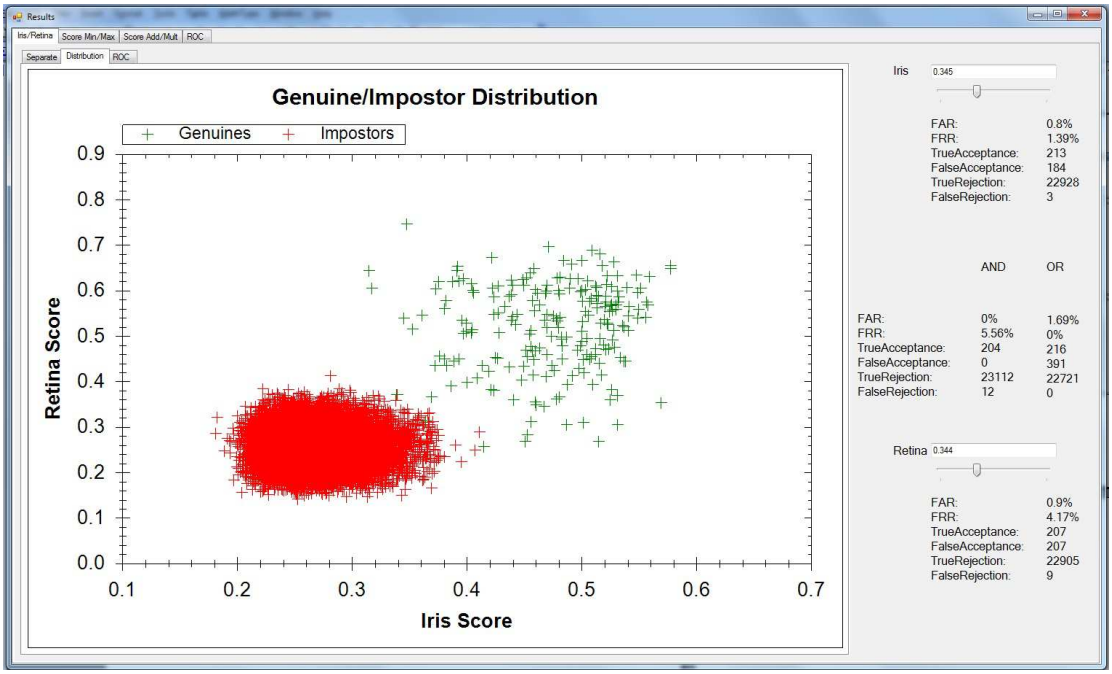


Figure 5.4: Distribution view.

6 Results

The developed biometric system was evaluated on two databases: CASIA-IrisV1 database for iris and database from STRaDe group for retina. The former contains 756 iris images of 108 distinct people and the latter contains 684 images of both retinas from 110 distinct people (220 distinct retina samples). After pairing these databases, the resulting set contains a total of 324 images of 108 different samples.

In order to evaluate the fusion results, the separate biometrics must be evaluated first. During the development of the iris recognition algorithm, parameters of Gabor filter needed to be selected in order to achieve the best performance possible.

In Table 6.1 and Table 6.2, performance of the system using different values of wavelength and orientation of the filter is shown. The performance analysis was tested on the full iris database. It is visible that the best results are achieved by using wavelength $\lambda = 2$ and orientation $\theta = 0°$.

ERR				
Wavelength\Orientation	-45°	0°	45°	90°
2	4.7%	1.8%	4.0%	22.6%
4	4.5%	2.8%	3.8%	23.1%
8	4.1%	2.8%	3.7%	23.9%
16	4.3%	2.9%	3.8%	23.8%

Table 6.1: ERR scores of the biometric system with differing wavelength and orientation used in Gabor filter.

During the development of the retinal recognition algorithm, four methods of obtaining score were considered:

1. average Euclidean distance of all bifurcation pairs
2. average Euclidean distance of bifurcation pairs with Euclidean distance lower than a given threshold
3. doubled number of matched bifurcation pairs / total count of bifurcations
4. previous two methods combined

Surprisingly, the first algorithm, which was originally intended to be used, didn't perform as well as was expected. Although the fourth algorithm is comparable to the third, it's slightly more computationally demanding. Therefore, the third algorithm was selected as the final evaluation algorithm.

The analysis was done on the paired database.

	Algorithm 1.	Algorithm 2.	Algorithm 3.	Algorithm 4.
EER	9.8%	21.0%	2.4%	3.8%

Table 6.2: ERR scores of the four retina evaluation algorithms.

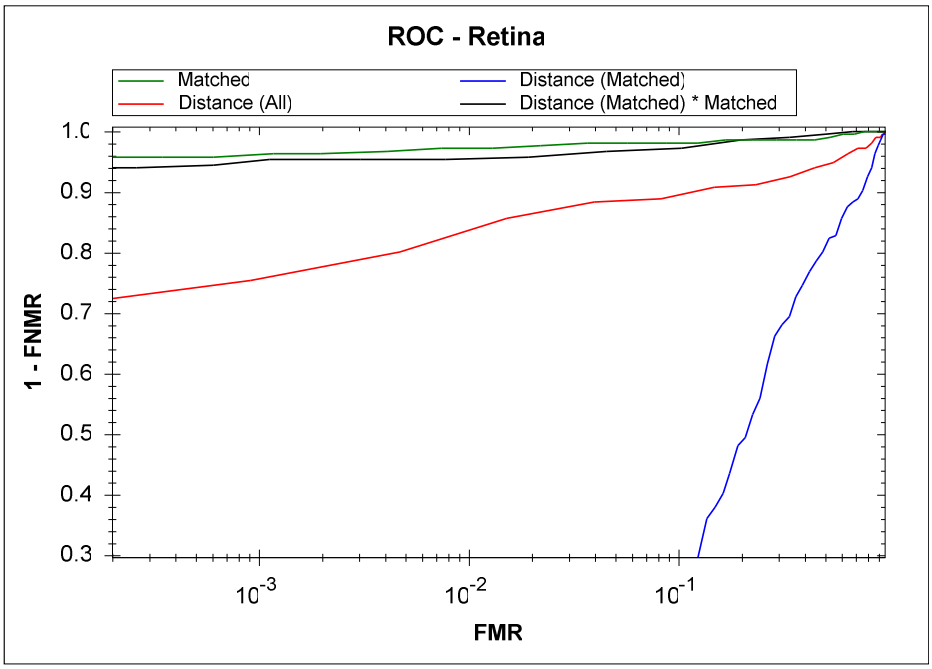


Figure 6.1: ROC curve of the algorithms. Algorithm 1. is depicted by red, 2. by blue, 3. by green, 4. by black.

Before fusion, the performance of both algorithms is summed in Table 6.3, tested on the paired database. The iris recognition algorithm performs slightly better than the retina algorithm. Also, it is recognizably faster due to its small template.

	Iris	Retina
True Acceptance	213	210
False Acceptance	184	244
True Rejection	22928	22868
False Rejection	3	6
FMR	0.80%	1.06%
FNMR	1.39%	2.78%
EER	1.2%	2.4%

Table 6.3: Results of separate biometric evaluation performance. The values of FMR and FNMR are corresponding to the best performance of the respective algorithms.

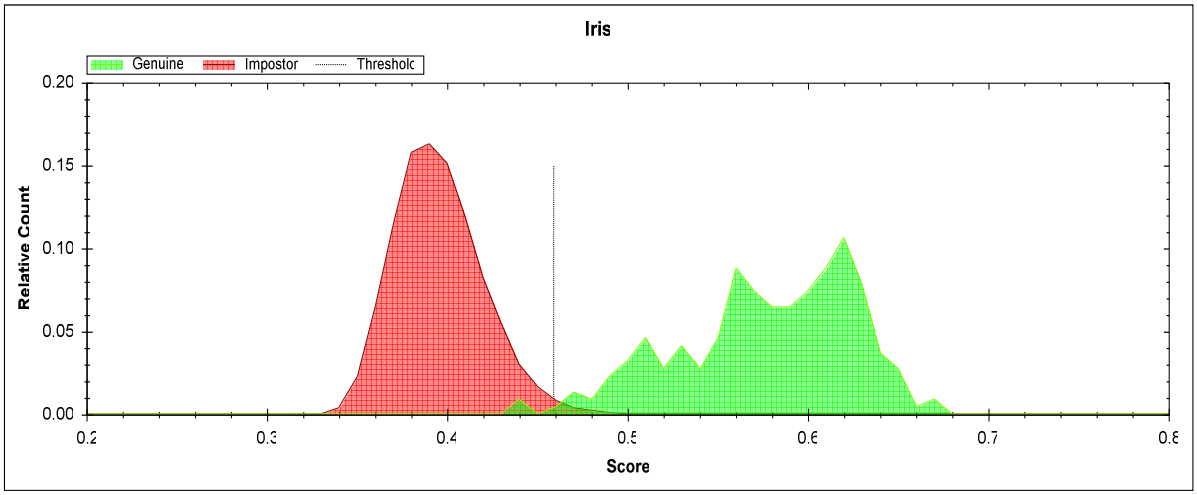


Figure 6.2: Iris recognition performance.

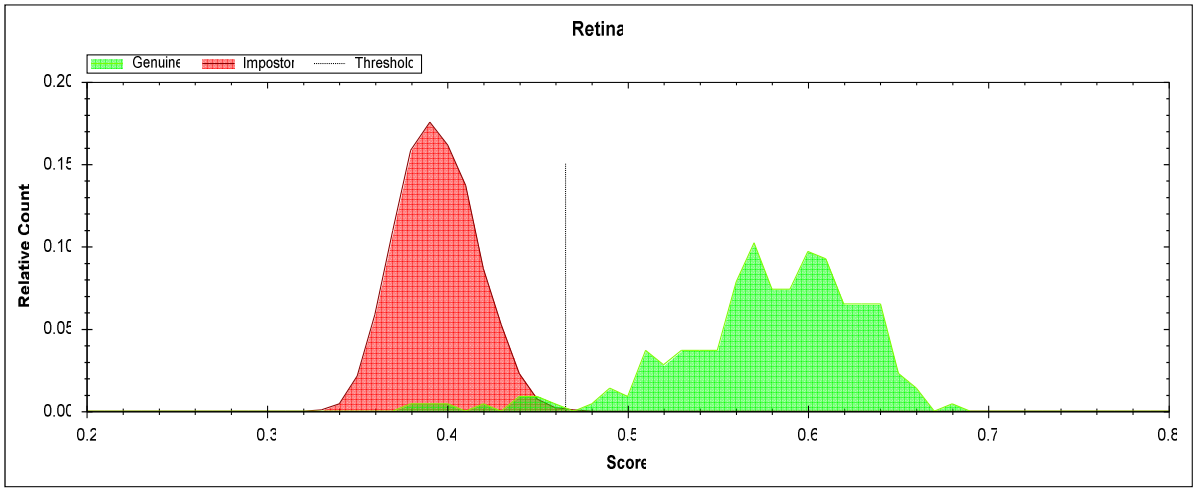


Figure 6.3: Retina recognition performance.

Biometric fusion is done at score-level, using transformation-based rules: minimum, maximum, sum, and product. The results are depicted below. From the following figure, it is visible that the fusion mostly achieved improvement.

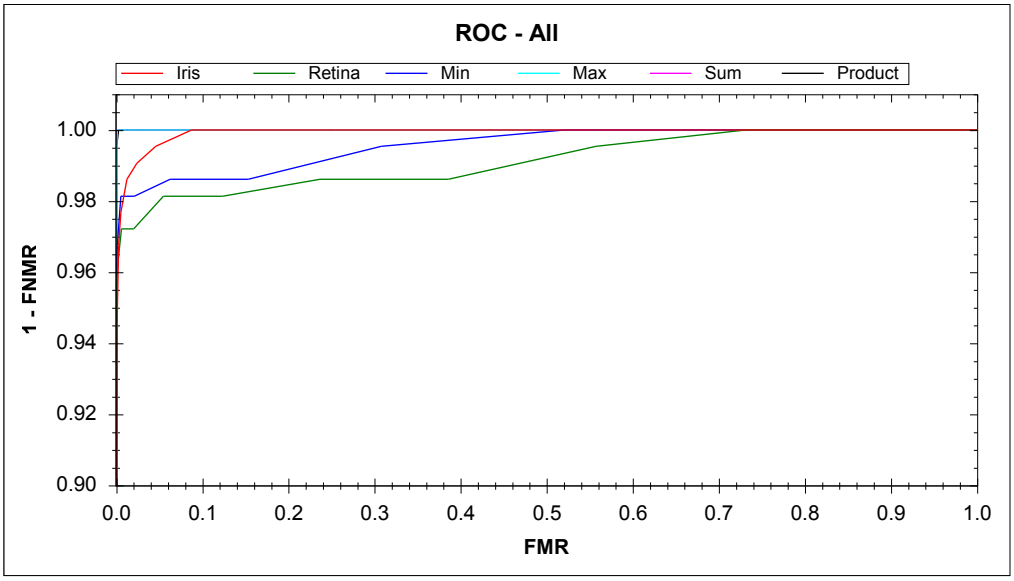


Figure 6.4: Linear ROC curve of the four fusion methods and the iris and retina algorithms.

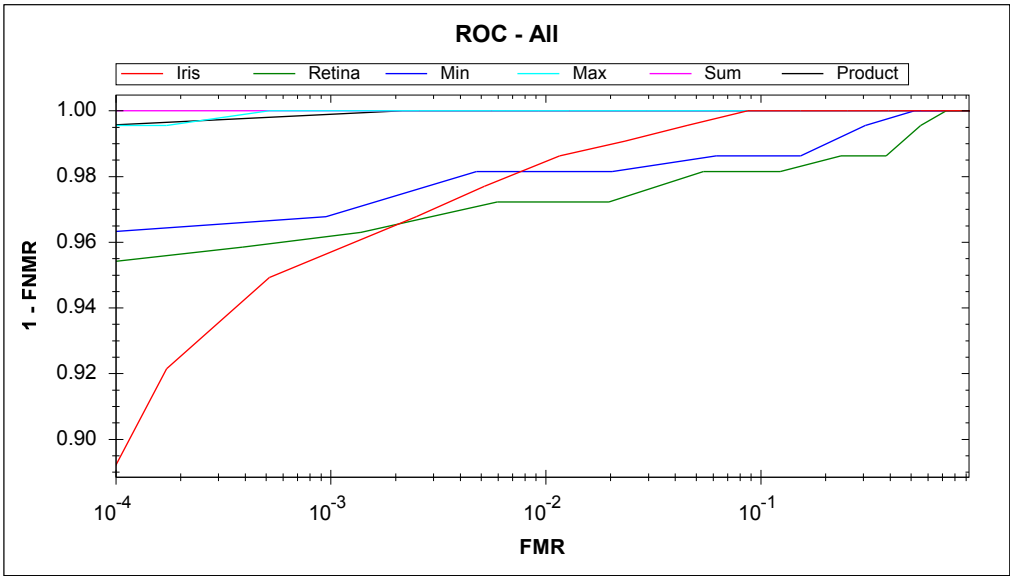


Figure 6.5: Logarithmic ROC curve of all four fusion methods and the iris and retina algorithms.

Curiously, the performance of the minimum rule was generally worse than that of the iris algorithm. However, the other rules enhanced the performance of the biometric system, with all three having almost identical scores (they overlap significantly in Figure 6.4). However, the best results were achieved by using the sum rule, albeit by a narrow margin.

	Iris	Retina	Min	Max	Sum	Product
FMR	0.80%	1.06%	0.13%	0.03%	0.01%	0.06%
FNMR	1.39%	2.78%	2.31%	0.00%	0.00%	0.00%
EER	1.20%	2.40%	1.85%	0.02%	0.01%	0.03%

Table 6.4: Final results. The values of FMR and FNMR are corresponding to the best performance of the respective algorithms.

7 Conclusion

The purpose of this thesis was to develop a multibiometric system combining iris and retina. For this reason, the algorithms for iris and retina recognition were designed. A database containing 324 pairs of images of 108 people was used.

The developed iris algorithm is based on Daugman's iris code, which involves unrolling the iris annulus to a stripe and demodulating it using a 2D Gabor filter. The resulting code is compact and therefore, the iris recognition algorithm is fast. With the specified database, the Equal Error Rate of the system is 1.2%.

The developed retina algorithm is based on matching bifurcations. This entails segmentation of the retinal vascular structure, thinning it to a width of 1 pixel and locating the bifurcation points. Out of four possible evaluation algorithms, the ratio of number of matched bifurcations to the total count of bifurcations proved to be the best, with EER of 2.4%.

The fusion was performed at score level using transformation-based rules, namely min rule, max rule, sum rule, and product rule. Before the fusion, the results of the separate biometrics evaluation were normalized using minimum and maximum values.

The performance results of this system showed that this fusion mostly achieves improvement over the separate biometric evaluation. Min rule achieved the worst results (ERR 1.85%). The remaining three rules achieved much better results: max rule has an EER of 0.02%, sum rule has an EER of 0.01%, and product rule has an EER of 0.03%. Out of the four tested rules, sum rule has achieved the best results.

The developed program displays both fused and non-fused results for comparison. The program also displays statistics and graphs depicting the performance of the system. In the program, databases can be selected for evaluation.

The program itself could be developed further in the future and expanded with more fusion methods for comparison. To make other levels of fusion more interesting for study, new modalities could be introduced to the program.

The conclusion that can be drawn from the results is that score-level biometric fusion is a viable option in the development of biometric systems combining iris and retina.

Literature

[1] A. K. Jain, A. A. Ross and K. Nandakumar. *Handbook of multibiometrics*. New York: Springer, 2006. ISBN 978-038-7331-232.

[2] H. ten Have and B. Gordijn. *Handbook of global bioethics*. ISBN 978-94-007-2513-9.

[3] J. M. Kizza. *Ethical and social issues in the information age*. New York: Springer, 1998. ISBN 03-879-8275-2.

[4] National Laboratory of Pattern Recognition. Biometrics Ideal Test [online]. [Accessed 2015-05-18]. Available from: http://biometrics.idealtest.org/

[5] EPA's IRIS: A Database with Blind Spots. Center for Progressive Reform [online]. [Accessed 2015-05-18]. Available from: http://www.progressivereform.org/iris.cfm

[6] A. K. Jain, A. A. Ross and K. Nandakumar. *Introduction to biometrics*. New York: Springer, 2011. ISBN 978-0-387-77325-4.

[7] W. Xiaomin, X. Taihua and Z. Wenfang. Chaos-based biometrics template protection and secure authentication. In: *State of the art in Biometrics*. InTech. DOI: 10.5772/19599.

[8] M. Schuckers. *Computational methods in biometric authentication statistical methods for performance evaluation*. New York: Springer, 2010. ISBN 18-499-6202-2.

[9] M. O. Derawi. What is Biometrics. Derawi Biometrics [online]. [Accessed 2015-05-18]. Available from: http://biometrics.derawi.com/?page_id=51

[10] L. Hong, Y. Wan and A. K. Jain. Fingerprint Image Enhancement: Algorithms and Performance Evaluation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(8):777-789

[11] T. Matsumoto, H. Matsumoto, K. Yamada, S. Hoshino and R. L. van Renesse. Impact of artificial "gummy" fingers on fingerprint systems. In: *Optical Security and Counterfeit Deterrence Techniques*. 2002, p. 275-289. DOI: 10.1117/12.462719.

[12] Advantages of Multi-biometric Systems over Unibiometric Systems. Techbiometric [online]. [Accessed 2015-05-18]. Available from: http://techbiometric.com/articles/advantages-of-multi-biometric-systems-over-unibiometric-systems

[13] A. K. Jain. Technology: Biometric recognition. In: *Nature*. 2007, p. 38-40. DOI: 10.1038/449038a.

[14] D. A. Socolinsky, and A. Selinger. Thermal face recognition over time. In: *Proceedings of the 17th International Conference on Pattern Recognition*. IEEE, 2004, p. 187-190. DOI: 10.1109/ICPR.2004.1333735.

[15] A. Bendjebbour, Y. Delignon, L. Fouque, V. Samson and W. Pieczynski. Multisensor image segmentation using Dempster-Shafer fusion in Markov fields context. In: *IEEE Transactions on Geoscience and Remote Sensing*. 2001, p. 1789-1798. DOI: 10.1109/36.942557.

[16] V. Vaidehi. Multi-Algorithmic face authentication system. In: *Lecture Notes in Engineering and Computer Science*. 2005, p. 485-490. ISSN 2078-0958.

[17] R. Brunelli, D. Falavigna, A. K. Jain and N. K. Ratha. Person identification using multiple cues. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 2009, p. 955-966. DOI: 10.1109/34.464560.

[18] T. B. Long and L. H. Thai. Hybrid Multi-Biometric Person Authentication System. In: *World Congress on Engineering and Computer Science*. 2012, p. 106-111. ISSN 2078-0958.

[19] M. Monwar, M. L. Gavrilova, A. K. Jain and N. K. Ratha. Enhancing security through a hybrid multibiometric system. In: *2009 IEEE Workshop on Computational Intelligence in Biometrics: Theory, Algorithms, and Applications*. 2009, p. 84-91. DOI: 10.1109/CIB.2009.4925691.

[20] M. Faundez-Zanuy, L. O'Gorman, A. K. Jain and N. K. Ratha. Data fusion in biometrics. In: *IEEE Aerospace and Electronic Systems Magazine*. 2005, p. 34-38. DOI: 10.1109/MAES.2005.1396793.

[21] *Encyclopedia of biometrics*. Editor A. K. Jain. New York: Springer, 2009. ISBN 978-0-387-73002-8.

[22] X. Xia, L. O'Gorman, A. K. Jainand N. K. Ratha. Innovations in fingerprint capture devices. In: *Pattern Recognition*. 2003, p. 361-369. DOI: 10.1016/S0031-3203(02)00036-5.

[23] F. Yan, M. Paindavoine and H. Abdi. Fast panoramic face mosaicing and recognition. In: *ISPA Proceedings of the 4th International Symposium on Image and Signal Processing and Analysis*. 2005, p. 197-202. DOI: 10.1109/ISPA.2005.195409.

[24] S. G. Goodridge. Sound Localization. Industrial Engineering and Systems Engineering | NC State University [online]. [Accessed 2015-05-18]. Available from: http://www.ise.ncsu.edu/kay/msf/fusion.htm

[25] E. Oja. Independent component analysis and blind source separation. [online]. [Accessed 2015-05-18]. Available from: http://research.ics.aalto.fi/bayes/biennial2003-2.pdf

[26] A. K. Jain, B. Chandrasekaran and N. K. Ratha. Dimensionality and sample size considerations in pattern recognition practice. In: *Handbook of Statistics*. 1982, p. 835. DOI: 10.1016/S0169-7161(82)02042-2.

[27] A. A. Ross, R. Govindarajan, A. K. Jain and N. K. Ratha. Feature Level Fusion Using Hand and Face Biometrics. In: *Biometric Technology for Human Identification II*. 2005, p. 196-204. DOI: 10.1117/12.606093.

[28] A. K. Jain, K. Nandakumar, A. A. Ross and M. Tistarelli. Score normalization in multimodal biometric systems: evaluation, application, and small sample performance. In: *Pattern Recognition*. 2005, p. 2270-2285. DOI: 10.1016/j.patcog.2005.01.012.

[29] A. K. Jain, D. Zongker, M. Bicego and M. Tistarelli. Feature selection: evaluation, application, and small sample performance. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 2007, p. 153-158. DOI: 10.1109/34.574797.

[30] A. Rattani, D. R. Kisku, M. Bicego and M. Tistarelli. Feature Level Fusion of Face and Fingerprint Biometrics. In: *2007 First IEEE International Conference on Biometrics: Theory, Applications, and Systems*. 2007, p. 1-6. DOI: 10.1109/BTAS.2007.4401919.

[31] J. Yang and X. Zhang. Feature-level fusion of fingerprint and finger-vein for personal identification. In: *Pattern Recognition Letters*. 2012, p. 623-628. DOI: 10.1016/j.patrec.2011.11.002.

[32] H. Hotelling. Relations Between Two Sets of Variates. In: *Biometrika*. 1936, p. 321-377. DOI: 10.1093/biomet/28.3-4.321.

[33] X. Jing, Y. Yao, D. Zhang, J. Yang and M. Li. Face and palmprint pixel level fusion and Kernel DCV-RBF classifier for small sample biometric recognition. In: *Pattern Recognition*. 2007, p. 3209-3224. DOI: 10.1016/j.patcog.2007.01.034.

[34] R. O. Duda. and F. R. Hampel. *Robust statistics: the approach based on influence functions*. Digital print. New York: Wiley, 2007. ISBN 978-047-1735-779.

[35] J. Kittler., M. Hatef, R. P. W. Duin and J. Matas. On combining classifiers. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 1998, p. 226-239. DOI: 10.1109/34.667881.

[36] R. O. Duda. *Pattern classification*. 2nd ed. New York: J. Wiley, 2001. ISBN 04-710-5669-3.

[37] P. Verlinde and G. Chollet. Comparing Decision Fusion Paradigms using k-NN based Classifiers, Decision Trees and Logistic Regression in a Multi-modal Identity Verification Application. In: *Second International Conference on Audio and Video-based Biometric Person Authentication.* 2003. DOI: 10.1.1.47.8153.

[38] M. Davide, A. K. Jain. *Biometric Authentication*. Berlin: Springer, 2005. ISBN 978-354-0259-763.

[39] K. H. Tim, J. J. Hull and S. N. Srihari. Decision combination in multiple classifier systems. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 1994, p. 66-75. DOI: 10.1109/34.273716.

[40] A. Agresti. *An introduction to categorical data analysis*. Hoboken: Wiley, 2007. ISBN 04-712-2618-1.

[41] J. Daugman. Combining Multiple Biometrics. Computer Laboratory: Faculty of Computer Science and Technology [online]. [Accessed 2015-05-18]. Available from: http://www.cl.cam.ac.uk/~jgd1000/combine/combine.html

[42] L. I. Kuncheva, C. J. Whitaker, C. A. Shipp and R. P. W. Duin. Limits on the majority vote accuracy in classifier fusion. In: *Pattern Analysis*. 2003, p. 22-31. DOI: 10.1007/s10044-002-0173-7.

[43] P. Domingos and M. Pazzani. On the Optimality of the Simple Bayesian Classifier under Zero-One Loss. In: *Machine Learning*. 1997, p. 103-130. DOI: 10.1023/A:1007413511361.

[44] L. Xu, A. Krzyzak, C. Y. Suen and R. P. W. Duin. Methods of combining multiple classifiers and their applications to handwriting recognition. In: *IEEE Transactions on Systems, Man, and Cybernetics*. 2003, p. 418-435. DOI: 10.1109/21.155943.

[45] S. P. Tankasala, P. Doynov, R. R. Derakhshani, A. A. ROSS and S. Crihalmeanu. Biometric recognition of conjunctival vasculature using GLCM features. In: *2011 International Conference on Image Information Processing*. 2011, p. 1-6. DOI: 10.1109/ICIIP.2011.6108974.

[46] The Eyes (Human Anatomy): Diagram, Optic Nerve, Iris, Cornea, Pupil, & More. WebMD [online]. [Accessed 2015-05-18]. Available from: http://www.webmd.com/eye-health/picture-of-the-eyes

[47] Biometrics: iris. MAINGUET, Jean-François. Jean-François Mainguet: page d'accueil [online]. [Accessed 2015-05-18]. Available from: http://fingerchip.pagesperso-orange.fr/biometrics/types/iris.htm

[48] C. Nickson. Acute Glaucoma. Life in the Fast Lane [online]. [Accessed 2015-05-18]. Available from: http://lifeinthefastlane.com/ophthalmology-befuddler-007-2/

[49] C. del Rosario. Cholesterol Lowering Statins Linked To Cataracts. Medical Daily [online]. [Accessed 2015-05-18]. Available from: http://www.medicaldaily.com/cholesterol-lowering-statins-linked-cataracts-257529

[50] J. Daugman. How iris recognition works. In: *Proceedings. International Conference on Image Processing*. 2002. DOI: 10.1109/ICIP.2002.1037952.

[51] M. Vatsa, R. Singh and P. Gupta. Comparison of iris recognition algorithms. In: *International Conference on Intelligent Sensing and Information Processing*. 2004, p. 354-358. DOI: 10.1109/ICISIP.2004.1287682

[52] H. Kolb. Simple Anatomy of the Retina. Webvision [online]. [Accessed 2015-05-18]. Available from: http://webvision.med.utah.edu/book/part-i-foundations/simple-anatomy-of-the-retina/

[53] A. K. Jain, R. M. Bolle and S. Pankanti. *Biometrics: personal identification in networked society*. New York: Springer, 2006. ISBN 978-038-7285-399.

[54] M. N. Islam, A. Siddiqui and S. Paul. An efficient retina pattern recognition algorithm (RPRA) towards human identification. In: *2009 2nd International Conference on Computer, Control and Communication*. 2009, p. 1-6. DOI: 10.1109/IC4.2009.4909204.

[55] M. Sabaghi. Retinal Identification System Based on the Combination of Fourier and Wavelet Transform. In: *Journal of Signal and Information Processing*. 2012, p. 35-38. DOI: 10.4236/jsip.2012.31005.

[56] M. U. Akram, A. Tariq and S. A. Khan. Retinal recognition: Personal identification using blood vessels. In: *6th International Conference for Internet Technology and Secured Transactions*. 2011, p. 180-184. ISBN 978-1-4577-0884-8.

[57] *Graphics gems IV*. Editor P. S. Heckbert. Boston: AP Professional, 1994. Graphic Gems Series. ISBN 0-12-336155-9.

[58] N. Dede. Implementation of thinning algorithm in OpenCV. OpenCV Code [online]. [Accessed 2015-05-18]. Available from: http://opencv-code.com/quick-tips/implementation-of-thinning-algorithm-in-opencv/

[59] R. Fisher. Hough Transform [online]. [Accessed 2015-05-18]. Available from: http://homepages.inf.ed.ac.uk/rbf/HIPR2/hough.htm

Appendix A: CD Content

- folder `src` – source code of the program
- folder `manual` – user manual of the program
- folder `install` – install files of the program
- folder `thesis` – thesis documents