Single-sensor hand and footprint-based multimodal biometric recognition

Magisterarbeit

zur Erlangung des akademischen Grades
eines Diplom-Ingenieurs (Dipl.-Ing.)
an der Naturwissenschaftlichen Fakultät
der Universität Salzburg

eingereicht von

Peter WILD

Betreuer: Ao.Univ.-Prof. Mag. Dr. Andreas UHL

Salzburg, im Januar 2008

Abstract

Biometric systems support the task of reliable automatic authentication, which is a key function for economic transactions in modern society. So far, no universal biometric modality suitable for all applications has been found. This thesis examines the multimodal fusion of different modalities using a single high-resolution scan of the human hand as input and relates existing techniques to a new biometric modality: the human foot. With the target application of wellness areas and spas, this new modality supports privacy interests and still provides satisfying accuracy. After an introduction to basic design principles, related work in palmprint, fingerprint, hand geometry, and footprint-based recognition is discussed. System modules for sensing, preprocessing, feature extraction, matching and decision for both implemented prototype footprint and hand-based biometric systems are described in detail. Necessary modifications due to anatomical differences are proposed and presented. Finally, a performance evaluation comparing overall accuracy and relative performance of individual features concludes this work.

Contents

1 Introduction **1**
1.1 Assignment of tasks 2
1.2 Motivation 2
1.3 Contributions 4
1.4 Outline 4
1.5 Notations used 5

2 Biometric system design **7**
2.1 Identification versus Verification 8
2.2 System properties 10
2.3 System performance 11
2.4 State-of-the-art in Hand biometrics 14
2.4.1 Hand geometry 15
2.4.2 Fingerprint 16
2.4.3 Palmprint 16
2.5 State-of-the-art in Foot biometrics 17
2.6 Multimodal biometric systems 19
2.7 Architecture of the proposed system 20

3 Sensing **21**
3.1 Sensor properties 22
3.1.1 Hand geometry sensors 23
3.1.2 Fingerprint sensors 23
3.1.3 Palmprint sensors 24
3.1.4 Footprint sensors 24
3.2 Proposed sensor 25

4 Preprocessing **29**
4.1 Overview 30
4.1.1 Hand preprocessing 31
4.1.2 Foot preprocessing 33
4.2 Binarisation 34
4.2.1 Fast Otsu Thresholding 36
4.2.2 Canny-edge-detection based binarisation 36
4.3 Rotational Alignment 37
4.3.1 Moment-based alignment 37
4.3.2 Palmprint-boundary based alignment 39

4.4 Silhouette extraction 40
4.5 Salient point detection 41
4.5.1 Salient point refinement for hands 42
4.5.2 Salient point refinement for feet 43

5 Feature extraction **45**
5.1 Overview 46
5.1.1 Geometric features 46
5.1.2 Texture-based features 48
5.2 Silhouette feature 49
5.3 Shape feature 51
5.4 Fingerlength and Toelength feature 53
5.5 Palmprint and Soleprint feature 55
5.6 Eigenpalms, Eigenfingers and Eigenfeet feature 57
5.7 Minutiae feature 60

6 Matching and Decision **65**
6.1 Overview 66
6.2 Minutiae matching 67
6.3 Dynamic time warp matching 68
6.4 Score level fusion 69
6.5 Decision level fusion 71

7 Experiments **73**
7.1 Test setup 74
7.1.1 Test databases 75
7.1.2 Acquisition protocol 75
7.2 Hand matching 76
7.2.1 Verification performance 76
7.2.2 Identification performance 81
7.3 Foot matching 82
7.3.1 Verification performance 83
7.3.2 Identification performance 85
7.4 Discussion 86
7.4.1 A note on statistical significance 87

8 Summary **89**

Bibliography **91**

1 Introduction

According to [10], the task of *biometric* (from Greek *bios*-life, *metron*-measure) systems consists of determining the personal identity based on his or her distinguishing physiological and/or behavioural characteristics (in the majority of cases both properties are addressed [21]). While the term "biometric" may also refer more generally to the application of mathematics and especially statistics to biological sciences, this work explicitly concentrates on the information technologic aspect focusing on authentication tasks. In order to recognise a person, a biometric software extracts a machine-readable representation of physiological and/or behavioural characteristics of that person, called *feature vector* or *template* [52], to facilitate score-based *matching* with stored templates in member databases. From this point of view, biometry is an important pattern recognition research problem [11]. Using hand or footprint-based measurements constitutes one of many different possibilities to realise biometric authentication. The term *multimodal* in the title implies that multiple evidences of the same identity are provided using not only a single biometric indicator but merging the information of multiple matchers in order to improve accuracy [32]. Biometric fusion is a common method to cope with unsatisfactory recognition rates or performance of *unimodal* systems [21]. While most multimodal biometric systems incorporate multiple sensors, this work operates on *optical single-sensor* output and may therefore be classified as a fusion system integrating *multiple representations and matching algorithms for the same biometric*, according to [14]. Finally, the term *recognition* in this work refers to both *verification* ($1:1$ comparison with a claimed identity template) and *identification* ($1:n$ comparison to find a matching template, if existing), as is used in respective literature [1, 21].

The research field of biometrics has seen an enormous growth in the last decade. Thus, numerous biometric identifiers emerged exhibiting certain strengths and weaknesses with respect to specific biometric applications. Whilst biometric patterns (fingerprint carvings) are present on archeological artifacts of the Neolithic age ($8000 - 7500$ B.C.) [21], modern biometric literature has its roots in criminal investigations with Alphonse Bertillon being the first to use physiological measurements to identify humans in the late 19th century [11]. In fact many of the techniques employed in biometric systems today, like *fingerprint classification* or the well-known *minutiae* features in fingerprints by Francis Galton, date back to the industrial revolution period of the late 18th and early 19th centuries [21]. Finally, another milestone for biometrics was constituted by the development of the automatic fingerprint identification system (AFIS) in the early 1960s [21]. Today, criminal applications are just one example where biometric measurements are employed. The market of civilian and governmental applications is constantly growing, as can be seen from the Biometric Market Report conducted by the International Biometric Group [47]: the industrial revenue of 3012.6 million US-Dollars in 2007 is expected to double in the next four years. The most common modalities ordered by market share according to this study are: fingerprint (25.3% and 33.6% AFIS), face (12.9%), iris recognition (5.1%), hand geometry (4.7%) and voice (3.1%). This picture has not changed drastically in the last five years, but numerous other modalities (vein and ear, for example) have been increasingly applied and multibiometric systems have emerged.

1.1 Assignment of tasks

The primary aim of this work is to relate traditional biometric characteristics (called *biometrics* or - when considered as a type of biometric systems - *modalities* [52]) targeting the human hand, or parts thereof, to a new biometric modality: the human foot. This is pioneering work, as footprint-related measurements have largely been neglected in existing literature up till now, at least for image-based approaches targeting *authentication*. In order to meet the required accuracy of 0.1% *False Match Rate* (*FMR*, the rate of falsely as genuine classified imposters) at 0.1% *False Non-Match Rate* (*FNMR*, the rate of falsely as imposter classified genuine users) for the authentication task (according to [11]), a set of different matchers has to be employed relying on foot shape and texture.

Another aspect of this work is to examine fusion improvements combining palmprint, hand geometry and fingerprint identifiers in a single-sensor *multibiometric* environment. While fusion may also be used to combine hand and footprint images (*multiple biometrics*) to achieve even better results, the work concentrates on *loosely coupled* (i.e. biometric classifiers are not combined at feature extraction level, but at the confidence or abstract level [21]), distinct hand and footprint *multiple matcher* scenarios (see [14]).

In order to evaluate system properties, including accuracy and performance, prototype systems for both hand- and footprints have been implemented. Some parts of this thesis focusing on the employed techniques for footprint-based recognition have already been published [38, 39]. This thesis presents all design aspects and also addresses target applications.

1.2 Motivation

Considering biometrics in general, its role in society is becoming more and more important. It is interesting to see that, despite the possibility to classify and identify animals or plants as well, the biometrics community almost exclusively concentrates on personal recognition of humans. This is mainly caused by its primary applications: in Austria and many more countries, every human has legal capacity (§16 ABGB). For the execution of economic transactions, crossing of international borders and many more civilian applications, a reliable personal identification is inevitable. Traditional *token-based* methods via passport, credit cards, etc. and *knowledge-based* identification using personal identification numbers or passwords, for example, have proven their worth in many applications [1]. However, a major drawback of these methods is constituted by the fact that holding tokens or passwords does not necessarily imply legal ownership. Both theft and fraudulently passed tokens (such as entry tickets) or knowledge may be undesired. *Negative recognition* (i.e. *screening* people against a database of members) relies on methods eliminating a person's ability to claim multiple identities and thus only works with biometrics [21], or at least demands fraud resistant tokens. Biometry facilitates personal identification in several ways [21]:

- **Permanence** and independence of tokens and knowledge: a biometric feature as a part of an individual may neither easily be lost without intention, nor forgotten like passwords;
- **Singularity**: for many features, like the *iris* or *fingerprints*, there is evidence that each person has unique characteristics;
- **Efficiency and user convenience**: biometric systems enable large throughput for access control and high accuracy at low cost and are thus being increasingly adapted for civilian and governmental applications.

However, its first property also implies a significant security threat, when, for example, sensitive biometric data may be compromised. This is especially true for biometric features which may be generated out of publicly available data. In contrast to traditional security mechanisms, biometrics do inherently not support *key replacement* [21], meaning that a biometric can not easily be switched as in the case for keys in cryptographic environments, for example. Therefore, it is in each user's interest that private biometric features (such as retina or even fingerprints) are not compromised. There is currently research going on in the field of *cancelable biometrics* (see [31]) separating immutable biometric features from the identifiers used for authentication. Taking the diversity of existing biometrics into account, another approach to alleviate this problem is to apply not the most secure biometric for each application but keeping in mind each user's privacy interests. Less secure biometrics become viable alternatives, if the risk of imposter attempts is expected to be low, or when the number of enroled members is guaranteed not to exceed a small amount.

No biometric is considered *optimal* in a sense that it meets requirements of all applications [14]. For this reason, this work examines *footprint-based* biometrics as a new emerging alternative as access control in wellness domains, spas or thermal baths, for example, and compares its performance to state-of-the-art hand geometry, palmprint and fingerprint techniques. Since footprints are not intended to support large-scale high security applications, such as electronic banking, the storage of features does not necessarily imply security threats. On the other hand, due to the practice of wearing shoes, it is difficult for imposters to obtain footprints for forgery attacks. Thus, footprint-based recognition may also be an alternative for highest-security applications.

Finally, *single-sensor multimodal hand biometrics* is expected to provide very high matching accuracy without the need of additional hardware for each employed modality. This alleviates user instructions for biometric sample capture, saves image acquisition time

[34], and, using commercially available flatbed scanners as input sensors, provides a cost-effective solution for large-scale decentralised biometric applications.

1.3 Contributions

This thesis examines personal single-sensor hand and footprint-based multimodal biometric verification and identification consolidating multiple sources of information.

The main contribution of this thesis to existing work is a newly developed *footprint*-based authentication system with a target application in wellness areas and spas. In contrast to existing pressure-based approaches, an image-based solution is proposed using high-resolution image-data of the human sole, which permits even higher recognition accuracy. Its evaluation uses a database of 32 volunteers, which is at least twice as large as existing approaches. The same features are also applied to the human hand and evaluated using a unique database of 86 people collected at the Department of Computer Sciences, University of Salzburg, Austria.

Second, due to uniform recording conditions (both hand- and footprint images were collected from the same user background using the same sensor and similar recording conditions), it conducts a *fair* comparison between both modalities. Finally, this thesis employs commercially available flatbed scanners for multimodal biometric recognition, thus enabling a wide use in existing systems (e.g. for criminal prosecution) without further hardware requirements.

1.4 Outline

Since many of the techniques to design and implement a biometric system are quite similar in hand and footprint-based biometrics, this thesis is not split into two parts for each of the different modalities. Instead, it is structured according to the different layers in a biometric system and points out the differences between hand- and footprint-based systems and necessary adaption for each of the employed processing steps.

The first part of this work, starting with Chapter 1, motivates biometric systems in general and hand- and footprint-based biometrics in particular. The goals of this work are identified and contributions are marked out. Mathematical notations are clarified and an outline of the thesis' structure is given.

In order to get in touch with the challenges of biometric system design, Chapter 2 presents architectural design, biometric system properties and performance evaluation measures of the implemented multimodal authentication systems. Related work in different hand-specific modalities, i.e. hand geometry, fingerprint and palmprint techniques, as well as related work in foot biometrics concentrating on pressure-sensing devices is introduced.

An overview of multimodal systems and the proposed systems' architecture concludes this chapter.

Chapter 3 examines existing sensors' properties for each of the modalities and legitimates sensor choice for this work.

Preprocessing steps for hands and feet are quite different and at the heart of the problem of normalisation of input samples. These steps are essential for the reliable extraction of biometric features and presented in Chapter 4.

The technical mapping between (normalised) input images and the compressed representation as a *template* is described in Chapter 5, introducing the set of implemented features for hands and footprints. All necessary modifications to employ each of the measures to both modalities are discussed.

Matching and decision functions for biometric templates are presented in Chapter 6. This also includes score normalisation and fusion mechanisms.

Chapter 7 introduces experimental setup and results concerning accuracy and performance of the introduced systems in *verification* and *identification* mode. Furthermore, it compares hand- and footprint-based recognition performance and analyses biometric system properties.

After a discussion of test results in Chapter 8, this thesis concludes with a summary of the main ideas and an outlook on potential future work.

1.5 Notations used

For the easy reading of mathematical structures in this work, different fonts are used in order to avoid ambiguities:

- Letters in italics or Greek symbols symbolise scalars, e.g. a, b, c or α, β, γ;
- Points in 2D are denoted by bold capital letters, e.g. $\mathbf{A}, \mathbf{B}, \mathbf{C}$;
- Sets are denoted in calligraphic font: $\mathcal{A}, \mathcal{B}, \mathcal{C}$. The set of natural numbers is denoted by $\mathbb{N}$, reals are referred to as $\mathbb{R}$;
- The Gothic type is used to refer to matrices for capital letters and vectors for lower case letters, e.g. $\mathfrak{A}, \mathfrak{B}, \mathfrak{C}$ and $\mathfrak{a}, \mathfrak{b}, \mathfrak{c}$. Indices are used to identify vector coordinates: $\mathfrak{a} = (\mathfrak{a}_1, \ldots, \mathfrak{a}_n)$;
- Functions are denoted by capital letters in italics, e.g. $F : \mathbb{R} \to \mathbb{R}$ or also by lower case letters in italics, if no misunderstanding is possible: f, g, h. Sequences as special forms of functions may also be referred to as $\langle a_n \rangle$ for the sequence $\langle a_1, a_2, a_3, \ldots a_n, \ldots \rangle$.

2 Biometric system design

When designing biometric systems, a number of questions need to be addressed, such as *comparison mode* (verification vs. identification), *operational mode* (automatic vs. semi-automatic), the selection of one or many *modalities* according to the needs of the application and *hardware* for sensing these modalities [21]. The awareness of the target application is crucial for the design of biometric systems. In the case of hand- and footprint-based systems, it is important to notice that evidently their target applications may be quite different. While multimodal hand-based biometrics perfectly fits into the mainstream of authentication applications, foot biometrics is too time-consuming in environments where users have to take off shoes first. Thus, the explicit definition of a target application for foot biometrics is needed. A good way to identify target applications is to formulate conditions on application domains. When the domain is found, it can be classified according to characteristics introduced by Wayman [40] (namely *cooperative vs. non-cooperative*, *overt vs. covert*, *habituated vs. non-habituated*, *attended vs. non-attended*, *standard vs. non-standard*, *public vs. private*, and *open vs. closed*), that further help to justify the selection of specific features.

The first question arising naturally when dealing with new modalities, such as footprints, is *"Why do we need a new modality?"*. There are many accurate techniques and foot biometry has largely been neglected so far, so the answer to this question is not obvious. However, when privacy interests of users are considered, footprint-based authentication might be the right choice, since it is not employed in high-security applications. Second, foot biometrics is closely related to hand geometry, palmprint and fingerprints. Many of the biometric techniques using the human hand or parts thereof can be translated to this new domain. In contrast to hand geometry, a footprint-based identification system may be implemented in *covert* mode (i.e. without awareness of the user that the biometric is taken [40]), if certain prerequisites are met (such as in environments where users walk on bare feet). It is, in fact, the only known covert biometric without being captured at a distance. To identify target application domains for footprint-based authentication, the following prerequisites should be met:

- **low throughput should be avoided** by choosing environments, where users already walk on bare feet;
- **unhygienic recording conditions should be avoided**;
- **large space for the sensor** has to be provided, i.e. foot biometric sensors can not be integrated into small devices (such as PDAs, notebooks);
- **privacy interests** should justify the selection of footprints as the identifier of choice.

Wellness areas, public baths, spas - areas where users do not wear shoes or socks - are ideal applications that can be derived from the restrictions above. For systems preventing unpaid access to fee-paid areas, footprint capture can be executed in front of barriers - possibly underwater - supporting both prepaid and pay-per-entry schemes. In contrast to the mainstream of *cooperative, overt, habituated, attended enrolment and non-attended recognition, standard environment, closed and private* applications [21], footprint-applications are generally *non-habituated* and may be *covert*.

2.1 Identification versus Verification

In the following section a more formal treatment of biometric authentication systems will be given, see also [1, 21] for extensive overviews. All biometric systems make use of a database:

Definition 2.1.1 (System database). *Let $\mathcal{M}$ denote the database of enroled members, $\mathcal{M} := \{\mathfrak{m}_1, \mathfrak{m}_2, \ldots \mathfrak{m}_m\}$. Each member $\mathfrak{m}_i \in \mathcal{M}$ is characterised as a tuple $\mathfrak{m}_i := (i, \mathfrak{i}_i)$ for $i = 1, \ldots, m$ where i is the member's label (e.g. user ID) and $\mathfrak{i}_i$ the stored feature vector (template).*

Within a biometric system, a separate module exists, which extracts feature vectors out of biometric samples:

Definition 2.1.2 (Feature extractor). *Let X be a biometric sample within a universe of samples $\mathcal{X}$, a feature extractor is a function $E : \mathcal{X} \to \mathcal{F}$, which maps each sample X to its feature vector representation $\mathfrak{x} \in \mathcal{F}$ within the feature space $\mathcal{F}$. Let $E_1, E_2, \ldots, E_i, \ldots$ denote different feature extractors.*

Two feature vectors are matched by the corresponding Matcher:

Definition 2.1.3 (Matcher). *Given two feature vectors $\mathfrak{x}, \mathfrak{y}$, a matcher is a function $S : \mathcal{F} \times \mathcal{F} \to \mathbb{R}$ returning a similarity score $S(\mathfrak{x}, \mathfrak{y})$. Different matchers are denoted as $S_1, S_2, \ldots, S_i, \ldots$.*

According to [1], such a system may be run in two different *modes*:

1. **Verification**: Users utilise IDs, cards or other external knowledge- or token-based unique identifiers to present their identity to the system and prove their legal usage through an instantaneously acquired biometric feature. Therefore the acquired biometric sample is compared to the stored reference template associated with the referred identity in the user database (one-to-one comparison) using some predefined threshold η, see also [14].

Definition 2.1.4 (Verification system)**.** *Given a biometric sample $X \in \mathcal{X}$ and a claimed identity $i \in \{1, \ldots, m\}$, a verification system is a function $V : \mathcal{X} \times \mathcal{M} \to \{genuine, imposter\}$ determining whether the claim is true by returning the class genuine or whether the claim is false by returning imposter, based on a threshold η:*

$$V(X, \mathfrak{m}_i) := \begin{cases} genuine, & if\ S(E(X), \mathfrak{i}_i) \geq \eta; \\ imposter, & otherwise. \end{cases}$$

2. **Identification**: Only the acquired biometric sample is used for reliable determination of the identity of the access claiming person. Thus, screening against all stored reference templates (one-to-many comparison) is needed, see [14].

   Definition 2.1.5 (Identification system)**.** *Given a biometric sample $X \in \mathcal{X}$, an identification system is a function $I : \mathcal{X} \to \mathcal{M} \cup \{reject\}$ determining the identity $\mathfrak{m}_i, i \in \{1, \ldots, m\}$ if existing or the state reject in case of no suitable identity can be determined:*

$$I(X) := \begin{cases} \mathfrak{m}_i, & if\ i = \arg\max_j \{S(E(X), \mathfrak{i}_j)\} \wedge S(E(X), \mathfrak{i}_i) \geq \eta; \\ reject, & otherwise. \end{cases}$$

   Apart from this definition of identification systems, the following variants exist [1]: sometimes the whole set of matching identities, i.e. the set $\{\mathfrak{m}_i : S(E(X), \mathfrak{i}_i) \geq \eta\}$, is considered (*threshold-based identification*). Another variant, not necessarily based on scores, is *rank-based identification* where the matcher is expected to return a fixed size k-dimensional ranking vector $(\mathfrak{m}_{\pi(1)}, \mathfrak{m}_{\pi(2)}, \ldots, \mathfrak{m}_{\pi(k)})$ with π a permutation of $\{1, \ldots, m\}$. Finally, there are also *hybrid* approaches returning a variable-sized ranking vector of maximum length k containing only templates exceeding the defined threshold.

It is clear, that running systems in identification mode is more challenging for the following reasons:

- **Security**: System error rates largely depend on the number of enroled members. This circumstance is illustrated by the following experiment: assuming a system operates at 0.1% *False Accept Rate* (*FAR*, the rate of falsely accepted imposter authentication attempts) with a member database size of $n = 250$, a single imposter trying to cheat the system will be lucky in approximately $1 - 0.999^{250} \approx 22.13\%$ of the cases (assuming a simplified model with equal *FAR* for each identity in the system database [1]). However, using a database size of $n = 2500$ the probability of being falsely accepted increases to 91.8%;

- **Performance**: The number of necessary comparisons increases with the number of enroled members. Large databases, such as the FBI Integrated Automated Fingerprint Identification System [48] with fingerprints of over 47 million subjects as of July 2007, need fast matching algorithms in order to be able to respond in reasonable time.

2.2 System properties

Depending on application, desired security and fault tolerance, a variety of different biometric features may be recorded. Accuracy is only one out of many quantifiable measurements used for the selection of a particular technique. As most biometric applications demand real-time capabilities, processing performance is also an important issue [13]. The following section gives an overview of common biometric system properties by means of a comparison between *footprint-based* biometrics and traditional approaches which can be accomplished.

For biometric system assessment a number of *properties* can be identified [14]:

- **Universality**: the availability of characteristics for each person in order to minimise the *Failure To Acquire* (*FTA*) rate, i.e. the rate a biometric system is unable to acquire or extract biometric templates from a person of the target population [52].
- **Distinctiveness**: singularity of biometric characteristics caused by genetic diversity as well as random processes affecting the development of the embryo [12] largely affects inter-personal variability. For distinctiveness performance assessment often twin studies like [12] are conducted.
- **Permanence**: biometric characteristics are permanent, if they are not subject to change over long periods of time. Changes in weight, for example, may influence pressure-based footprint features. Ridge-based fingerprint features are known to remain stable according to empirical observations [21].
- **Collectability**: biometrics are expected to alleviate authentication and thus demand easily measurable and quantifiable characteristics. Usually, biometrics with high collectability (face, signature or hand geometry) also show high acceptability [10];
- **Performance**: such as computational or resource demands of the system. Inspecting biometric system assessment in [10], a tradeoff between performance and acceptability can be identified legitimating the diversity of biometrics;
- **Acceptability**: negative associations using a particular biometric are usually undesired. However, sometimes high uniqueness comes at the cost of lower acceptability, like for *iris, retinal scan* or *fingerprint* biometrics [10] due to more complex acquisition, association with criminal investigation and privacy issues;
- **Circumvention**: it should be impossible for intruders to defraud the biometric system. For some biometrics, like *signature* or *face*, this is hard to achieve.

Another differentiating factor, according to [21], is whether the biometric is *dynamic* (a behavioural characteristic is recorded, e.g. *gait, voice print*) or *static* (physiological features such as *iris, face, etc.* are recorded). For some biometrics, such as *signature*, both dynamic and static features may be analysed. This thesis concentrates on static image-based systems examining hand- and footprint-based biometrics. An assessment of these properties for hand- and footprint-based biometrics is found in Table 2.1.

Modality	Universality	Distinctiveness	Permanence	Collectability	Performance	Acceptability	Circumvention
Hand geometry	medium	medium	medium	high	medium	medium	medium
Palmprint	medium	high	high	medium	high	medium	medium
Fingerprint	medium	high	high	medium	high	medium	medium
Soleprint	medium	medium	medium	low	medium	medium	medium
Foot geometry	medium	medium	medium	low	low	medium	medium
Ballprint	low	high	medium	low	medium	medium	medium

Table 2.1: Comparison of hand-based (in: Jain et al. [14]) and footprint-based biometric technologies (according to the author's perception).

In the early stages of biometric research these properties characterised existing systems very well. Today, due to the diversity of existing algorithms, they can rarely be used to predict or classify the performance of a specific system. Using common *fusion techniques* it is possible to incorporate multiple matchers into a single biometric system yielding higher performance than each of the single biometrics [32]. Furthermore, techniques like *cancelable biometrics* influence acceptability.

2.3 System performance

Concrete biometric systems are assessed using statistical error analysis. There are a number of errors made by biometric systems, which need to be understood and estimated before a particular biometric is selected for application. Since various error types depend on the formulation of hypotheses, I clearly state the examination of *positive authentication* in both *verification* and *identification* mode.

A *verification* system has to classify the input X into one of two classes, i.e. given a claimed (enroled) identity M it decides which of the following (null and alternate) hypotheses is true [21]:

$$H_0 : X \equiv M, \textit{i.e. both templates refer to the same person}; \tag{2.1}$$

$$H_a : X \not\equiv M, \textit{i.e. the templates do not refer to the same person.} \tag{2.2}$$

Consequently there are two errors this type of a verification system can make [1], namely false accepts and false rejects.

Definition 2.3.1 (False Accept)**.** *False accepts refer to deciding H_0 is true, while in reality H_a is true (an imposter is accepted by the system). The relative frequency of this error is called False Accept Rate (FAR).*

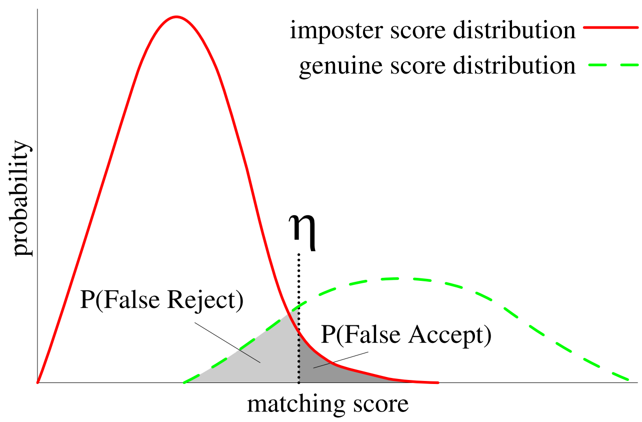


Figure 2.1: Sample genuine and imposter score distributions.

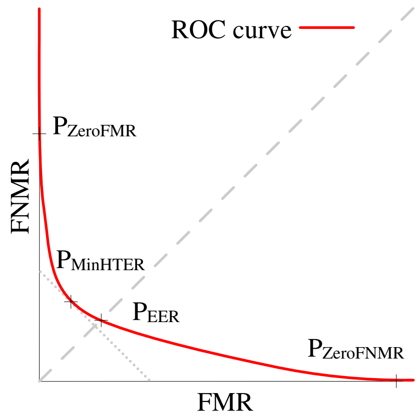


Figure 2.2: Sample Receiver Operating Characteristics.

Definition 2.3.2 (False Reject)**.** *False rejects refer to deciding H_a is true, while in reality H_0 is true (a genuine user is rejected). Again, the rate at which a false reject occurs is called False Reject Rate (FRR).*

Occasionally in literature, the expressions *False Match, False Non Match* with the corresponding rates *False Match Rate* (*FMR*) and *False Non-Match Rate* (*FNMR*) are used as a synonym for *False Accept, False Reject.* In a strict sense, there exists a difference between *FMR*/*FAR* and *FNMR*/*FRR*, namely: *FAR*, *FRR* (a) refer to *pattern recognition* terminology [1], frequently used in *positive authentication* scenarios; (b) also include errors specific to biometrics, such as the *FTA* [22] and; (c) refer to hypotheses, not subjects [1]. Unless otherwise specified the rates in this work will not include the *Failure to Acquire Rate*, i.e. the *FMR*, *FNMR* rates are employed.

In should be pointed out, that both *FMR* and *FNMR* depend on the current system threshold η. In fact, it is possible to achieve an arbitrarily low *FNMR* at the cost of higher *FMR* and vice versa. A good selection of η ideally separates the score distributions of genuine and imposter authentication attempts. However, this yields a tradeoff between *security* (demanding low *FMR*) versus *convenience* (requiring low *FNMR*) [21]. Suitable choices for η are visualised in genuine and imposter score distributions (see Figure 2.1). Inspecting the genuine score distribution $P(s|H_0)$ and imposter score distribution $P(s|H_a)$ of a biometric system, *FMR* and *FNMR* for any fixed system threshold η can be estimated as follows (see [21]):

$$FMR = \int_{\eta}^{\infty} P(s|H_a)\, ds; \tag{2.3}$$

$$FNMR = \int_{-\infty}^{\eta} P(s|H_0)\, ds. \tag{2.4}$$

The selection of an arbitrary operating point, i.e. tuple $(FMR, FNMR)_\eta$ of FMR and $FNMR$ values for some fixed threshold η, is typically insufficient for comparison. Instead, different algorithms are compared using the set of all possible operating points depicted in the form of a Receiver Operating Characteristics (ROC) curve [1] (see Figure 2.2). Alternatively, for performance comparison the following indicators are frequently adopted (see also the Fingerprint Verification Competition 2006 [50]):

- **Equal Error Rate** (*EER*): the value such that $FMR = FNMR$;
- **Zero False Match Rate** (*ZeroFMR*): the lowest $FNMR$ for $FMR = 0\%$;
- **Zero False Non-Match Rate** (*ZeroFNMR*): the lowest FMR for $FNMR = 0\%$;
- **Minimum Half Total Error Rate** (*MinHTER*): the global minimum value of the following function [30]:

$$HTER(t) := \frac{FMR(t) + FNMR(t)}{2}. \tag{2.5}$$

As can be seen in Figure 2.2, each of these performance indicators refers to a single point in the ROC curve: $\mathbf{P}_{EER}$ is the intersection of the ROC curve with the first median and corresponds to the *EER* indicator. Analogously $\mathbf{P}_{ZeroFNMR}$ and $\mathbf{P}_{ZeroFMR}$ refer to *ZeroFNMR* and *ZeroFMR* and can be identified as intersections of the ROC curve with the y-axis (and x-axis respectively) with closest distance to the origin. Finally, $\mathbf{P}_{MinHTER}$ corresponds to the operating point where the half of the sum of coordinates is a minimum.

It should be pointed out that, for *identification* systems, the situation is slightly different. Here, using the notions above, the hypotheses could be reformulated as follows:

$$H_0 : \exists M \in \mathcal{M} : X \equiv M, \textit{i.e. there exists an enroled template with the same identity}; \tag{2.6}$$

$$H_a : \forall M \in \mathcal{M} : X \not\equiv M, \textit{i.e. within } \mathcal{M} \textit{ no template with the same identity exists.} \tag{2.7}$$

Within this context, a *False Accept* occurs, if X is found to be matched with a template of an enroled member $\mathfrak{m}_i \in \mathcal{M}$, while in reality it is an imposter and *False Rejects* denote falsely rejected genuine authentication attempts. Matching with multiple (correct or incorrect) candidates is ignored in this case.

2.4 State-of-the-art in Hand biometrics

Concerning the hand-based recognition part of this work, not only a single biometric modality is of relevance, but three common modalities extracted out of geometric and textural physiological properties of the human hand [14]:

- *hand geometry* extracting local measurements (such as finger lengths, palm width, etc.) out of a (binary) image of the whole hand;
- *palmprint* focusing on position and orientation of principal lines and;
- *fingerprint* extracting the flow of ridge patterns in the tip of the finger.

Additionally, *palm vein patterns* extracted from infrared palmar images were introduced in 2002 [44] as a new distinct type of biometric authentication. All techniques have distinct biometric properties and are aimed at different applications. Different sensing devices for each of the modalities have emerged [21, 44].

The human hand soon appealed to biometricians due to its high accessibility. In the meantime, more sophisticated techniques have emerged yielding extremely low *EER*s (e.g. *iris* recognition systems), however hand-based features are still one of the most common biometrics used today. While a variety of different hand-biometric systems (such as [18, 35, 42]) already exist, whole-hand-based systems, scanning the entire hand to extract multiple features, are not yet common. The fusion of hand geometry, palmprint and fingerprint biometrics in a single sensor environment has been proposed by [19] and an implemented solution has recently been presented by [34]. Nevertheless no implementation is known to the author working with commercially available scanners so far. However, results using randomly paired samples, as is done in [19] seem promising. Using simple fusion techniques, recognition rates of single biometrics with *EER*s of 8.54% (hand shape), 11.06% (palmprint) and 11.4% (fingerprint) could be improved to 3.53% (hand shape + palmprint + fingerprint). Using *multi-spectral* scanning methods [34], even higher accuracy can be achieved. However, unfortunately the lack of standardised testing conditions regarding population size, performance indicators, time lapse between recordings, and even operation modes makes fair comparisons between hand geometry, palmprint and fingerprint systems difficult. Typical state-of-the-art error rates associated with the introduced techniques can be found in Table 2.2. It should be clear that error rates also heavily depend on the type (and supported area) of input sensor used (e.g. *thermal*, *optical* or *capacitative* in case of fingerprints) and whether the algorithms performance tests have been executed by third parties. For example, in the FVC2006 [50, 3] the best reported *EER* results in the *open* category are 0.021% *EER* for the optical sensor versus 5.564% for a different database with an electrical field sensor. This implies, that it is even more difficult to compare different biometrics. Therefore, the conducted performance comparison in this work is *fair* in terms of sensor (all feature extractors are provided with the same sensed information and may discard different parts of the hand) and population (all tests refer to the same group of users).

Type	Reference	Description	Samples	*FMR*	*FNMR*	Indicator
Geometry	Kumar et al. [18]	4 finger lengths, 8 finger widths, palm width, palm length, hand area, and hand length	1000	5.29%	8.34%	*MinHTER*
Geometry	Yoruk et al. [43]	independent component features of the hand silhouette images	458	2.77%	2.77%	*EER*
Palmprint	Kumar et al. [18]	standard deviation of grey-levels in 144 overlapping blocks	1000	4.49%	2.04%	*MinHTER*
Palmprint	Zhang [44]	phase information of discrete Gabor Filter convoluted with palmprint sub-image	425	0%	2.5%	*ZeroFMR*
Palmprint	Zhang [44]	projection of 128×128 sub-image onto eigenspace spanned by 100 eigenvectors (Eigenpalm approach)	3056	0.03%	1%	$FMR \approx 0$
Fingerprint	FVC2006 DB2 open winner [50]	anonymous algorithm on 400×560 (569 dpi) optical sensor data (BiometriKa)	1680	0.02%	0.02%	*EER*

Table 2.2: Error rates of recent hand-based biometric systems in verification mode.

All hand-based systems have the following major limitations in common: Firstly, most systems are overt, i.e. acquisition is an evident and cooperative process. Secondly, capture takes place with the subject nearby and often well-defined environmental conditions (such as e.g. lighting) are required. Finally, limitations concerning universality exist. The inability to acquire features may be caused by serious infringement (e.g. land mines), congenital physical anomalies (e.g. *polydactyly* causing supernumerary fingers or *dermatopathia pigmentosa*, a disorder causing a lack of fingerprints), or even inappropriate hardware unable to deal with extreme shape.

2.4.1 Hand geometry

Hand geometry systems have been implemented since the early 1970s [13] and target the extraction of the silhouette shape of the human hand or single fingers, finger lengths and local widths. Measurements can easily be extracted from low quality scans or camera images of the hand with resolutions starting at 45 dpi [43], as no textural information is involved. Shape-based features are invariant under environmental factors such as lighting conditions, sweat, dry skin or small injuries, however they may change over larger time spans, especially during growth [14]. Another difficulty is constituted by the physical anatomy of the human hand. If fingers touch each other, *salient points* [28] needed for reliable finger segmentation can not be extracted reliably. Most hand geometry detection will fail for users having their fingers closed or with physical anomalies, such as *polydactyly*. Although hand geometry measures do not vary significantly across different people [14], they can nevertheless be used for the verification task and to improve system accuracy using fusion. Therefore, three different algorithms from the class of geometric features will be employed in this work for multimodal hand-based biometric recognition: a contour-based approach, an algorithm targeting lengths of fingers and a shape-based feature.

2.4.2 Fingerprint

The modern fingerprint technique dates back to the 16th century, but it was the understanding of the following biological properties by Edward Henry in 1899, which led to the success of fingerprint recognition [21]: firstly, epidermal ridges form individual characteristics (fingerprints are part of an individual's *phenotype*); secondly, fingerprints may be classified according to configuration types and finally, minutiae landmarks and ridges are expected to be permanent (fully formed at about 7 months).

Typically, fingerprint matchers analyse features at a specific level [21]: *Level-1* fingerprint features (also known as *global level* features) track *singular points* (loops and deltas), can be extracted out of greater than 200 dpi resolved input images and allow coarse matching. *Level-2* features are extracted at the *local level* starting at $350 - 500$ dpi and consist of *local ridge characteristics*, so called *minutiae* points (*Bifurcation* and *Termination* points). Finally, *Level-3* features refer to *sweat pores* which can be identified in very highly resolved input images (at greater than 1000 dpi). Most fingerprint systems are *minutiae*-based, but there are also *correlation-based* and *ridge feature-based* (based on ridge orientation, texture, etc.) matchers, which are generally more tolerant to low quality input [21]. They are the feature of choice if the overlapping area between template and sample is relatively small.

In this work, NIST's NFIS [51] *minutiae-based* extractor `mindtct` and `bozorth3` matcher for fingerprint matching of each individual finger is applied. Other prominent algorithms may be found in [50]. Compared to other biometrics, fingerprint recognition exhibits relatively high performance and is shown to be distinctive, even for twins [12, 14]. Drawbacks of fingerprint technology are its high resolution requirements (for fingerprint storage a specific compression technique - Wavelet Scalar Quantization - became the FBI standard [21]), associations with criminal prosecution, and universality issues (so-called *goats* refer to *unusable* fingerprints [21]).

2.4.3 Palmprint

Palmprint recognition refers to the region of the human hand between the fingers and the wrist and exploits many of the same characteristics as for fingerprint recognition. Therefore, it shares many of its permanence and distinctiveness properties. A first application of palmprints was *fortune telling* and *health diagnosis* in China, starting in 1000 B.C. [44].

As an alternative to signatures for people unable to write their name, inked palmprint impressions were used systematically in 1858 by Sir William Herschel in India for workers to sign contracts [53]. After an integration of palmprints into AFIS systems starting in 1994, the largest available database of palmprints (in Australia) contains 4.8 million templates [53]. It is interesting to note that despite the fact that every third latent print is from a palm and not a finger [53], there are far more fingerprints stored (the FBI card archive already contained 200 million prints by 1995 [21]).

The palmprint pattern consist of the following elements [44]:

- thick *flexure lines* (or *principal lines* with its main lines *life line, head line and heart line*), which form individual patterns and close during gripping;
- *tension lines* (or *wrinkles*) providing elasticity to the skin and;
- *papillary lines* (or *ridges*) forming the structural outermost part of the epidermis.

In addition to *delta-point*-based, *minutiae*-based, *correlation-based* and *ridge feature-based* features like in fingerprint systems [53], the following measures may be extracted [44]: *geometrical features* (such as width, length or area), *principal line features* (i.e. location and form of life, head and heart lines), *wrinkles* (location and orientation), and *datum points* (end points of principal lines). This work focuses on a generic correlation-based approach estimating the difference between two aligned palmprint image textures, since these measures can be extracted fast and reliably even in poor quality images and do not require high resolutions. One technique with low error rates is correlation-based filters designed to produce outputs with high amplitude at the origin when correlated with an image belonging to this class and a lower-energy plane without significant peaks when correlated with a different class [7]. However, they demand multiple training images to generate the filter and some target applications require enrolment using a single input image. Therefore, a more generic approach, as in [18], is selected for this work. Another employed palmprint-based technique frequently used in face recognition [44] is the Eigenpalms feature (and Eigenfaces for face images respectively [37]). Based on principal component analysis it exhibits very low error rates (see Table 2.2) and does not need high-resolution input. This feature will be applied for both palms and individual fingers.

2.5 State-of-the-art in Foot biometrics

Foot biometry is still an open research topic. Even though foot biometry bears similar distinctive properties to the human hand, it has so far not been considered as a feature in non-forensic applications, for example in authentication systems. In contrast to hand biometrics, foot biometry is rarely used in commercial applications for a number of reasons: non-habituated environment, non-user-friendly data acquisition and, finally, issues regarding the hygienic capture of footprints. Furthermore, in some countries, it is considered offensive to show someone the sole of your foot. However, with the target environment of spas and thermal baths, footprint-based authentication might be a useful alternative,

Type	Reference	Description	Subjects	Recognition
Pressure	Nakajima et al. [26]	raw image $\mathcal{L}^2$-dist.	10	85%
Pressure/Body posture	Jung et al. [15]	quantised Center of Pressure (COP)	5	97%
Dynamic pressure	Jung et al. [16]	Hidden Markov Models (HMM)	11	80%

Table 2.3: Recognition rates of footprint-based biometric systems in identification mode.

since, as a biometric feature, it does not rely on tokens or knowledge for identification. In order to increase acceptability, proposed image acquisition in this work does not involve pegs or a specific pre-alignment of the foot on the sensor.

The first medical and forensic investigations related to footprint-based recognition were conducted in the late 1980s: Kennedy [17] recorded physiological characteristics from inked barefoot impressions, such as local geometrical features targeting length between heel and tips of toes, widths at various positions, and distances between optical centers of heel and toes. Automated recognition systems were introduced by Nakajima [26] in 2000 operating on pressure distribution data obtained by a pressure sensing mat using Euclidian distance. With this technique, recognition rates of 85% could be obtained. More recent work by Jung et al. [15, 16] focuses on static and dynamic footprint with Hidden Markov Models yielding recognition rates up to 97.8% dependent on feature selection and database size. However, recognition rates in Table 2.3 refer to very small populations: just $5 - 11$ people were tracked in experiments. Commercial applications, however, demand prospective results for larger population databases which are not available right now. Furthermore, an application domain is missing in existing proposals. For this reason more elaborate approaches to foot biometrics are investigated, using optically acquired footprints for feature extraction. Foot biometrics can also be related to *gait recognition* concentrating on behavioural characteristics [1].

In this work, I will apply different techniques from hand biometry to the high-resolution textural image of the sole. Since foot biometry is closely related to hand geometry, fingerprints, and palmprints, many of the techniques can be translated to this domain. Examples comprise, but are not limited to, [18, 19, 42, 43, 34]. Measurements from all of the introduced hand-based modalities will also be applied to footprints, including (a) *geometrical characteristics* focusing on shape, length and area of the silhouette curve, lengths of toes and inter-toe angles, local foot widths and; (b) *textural characteristics*, such as soleprint features analogous to palmprints, minutiae details on the ballprint and *Eigenfeet features* corresponding to *Eigenfaces* in traditional face recognition.

2.6 Multimodal biometric systems

In its most demanding forms of application, namely *screening* (i.e. matching against a database of wanted persons, e.g. terrorists) and *large scale identification* (i.e. identification from a large number of possible subjects, e.g. criminal investigation), biometrics is faced with the problem of having to guarantee extremely low error rates. Jain et al. [11] quantified accuracy requirements for matchers as less than $1 \cdot 10^{-3}\%$ *FNMR* for large scale identification from 1 million members and less than 1% *FNMR* for screening from a watch list of 500 members respectively at $1 \cdot 10^{-4}\%$ *FMR*. The main reason for these requirements is the fact that in *identification mode* the *FMR* is approximately linear dependent on the number of enroled members in the system database [1]. These rates, however, can hardly be accomplished in *unimodal* systems and it is even hard to bridge the gap between current matchers and performance requirements in *multimodal* biometric systems. Design issues of multibiometric systems (as introduced in the last 10 years) are discussed in [32, 21, 33]. Multibiometric techniques were introduced by *multi-classifier* literature (multibiometric systems may be seen as multi-classifiers over a two-class classification problem [21]) and have gained enormous popularity in the last decade due to [33]:

- their ability to improve matching accuracy;
- higher flexibility in case of failure to acquire single biometrics and;
- more difficult biometric system attacks (all individual biometrics have to be attacked at the same time).

Most multibiometric systems today incorporate fusion in multiple unit and multiple biometrics scenarios (see [14]), since these combine completely independent pieces of information and thus result in higher matching improvements [21]. The systems introduced in this work are single-sensor multibiometric systems and are thus, in the sense of [14], “only” *multiple matcher* scenarios (and therefore considered to combine strongly correlated measurements in the opinion of [21]). However, features are expected to be largely independent, when extracted at different resolutions, such as e.g. the global (*singular points*), local (*minutiae*) and very-fine (*sweat pores*) fingerprint levels, and from different parts of the input image. Using the latter, one can see that *multiple unit* scenarios in [14] may be considered as subsets of *multiple matcher* scenarios when the input covers multiple units and is constrained in size for single matchers. This is, in fact, the case in the proposed system, when fingerprint regions of single fingers are extracted as part of the *preprocessing* step and the results of individual units are merged.

Multimodal biometric systems employ *fusion strategies* to consolidate information. According to [32] fusion may be incorporated:

- **at the feature extraction level** consolidating multiple independent biometric feature vectors via concatenation into one single high-dimensional template;
- **at the matching score level** combining the individual scores of multiple matchers into one score indicating the similarity between feature vector and reference template;

- **at the decision level** via simple majority voting schemes.

Since the employed multimodal fusion strategies are applied at matching and decision level, see Chapter 6 for further details on these techniques.

2.7 Architecture of the proposed system

Typical multimodal biometric authentication systems, such as [18, 19, 34], for example, consist of the following five serial modules (which themselves may be further split up into several processing steps), see also [14, 44]. Sometimes, when preprocessing or fusion is not executed, the corresponding modules may be missing in the system's description.

1. **Sensor module**: This module extracts the raw measurements of the hardware sensor, such as flatbed scanners, fingerprint readers, cameras, etc.
2. **Preprocessing module**: Normalising the input signal in order to achieve translational and rotational invariance is a key task of the preprocessing stage and can increase recognition accuracy enormously (see e.g. [26] for footprints or [18] for hand-images). In particular for single sensor multimodal systems, this module is important, as it may provide several versions of the (clipped) input image for different resolutions, see [19].
3. **Feature extraction module**: Within this part of the system the *feature vector* is generated from the (preprocessed) input signal. As feature extraction may also involve further preprocessing tasks, the distinction between preprocessing and feature extraction module is not always clear. Therefore, both tasks are often merged within a single module (such as in [1]).
4. **Matching module**: Having a set of stored templates called *member database* the matching module compares the extracted *feature vector* to one specific template (*verification*) or all stored references (*identification*) and generates a matching score s for each comparison.
5. **Decision module**: The task of granting or denying access based on matching scores is executed by the decision module.

If the system is *multimodal*, an additional *Fusion module* has to be integrated. Its position within the processing chain is dependent on the type of fusion used. The proposed system design illustrates processing in case of *identification* and *verification*, where biometric templates are matched against each other in order to solve authentication problems. For the preceding storage of biometric templates into the member database, the so-called *enrolment*, the processing chain not necessarily involves *matching* or *decision* stages, see [21]. Instead, multiple (one or more) samples are acquired (possibly under guidance of humans supervising this process) for each individual to be enroled and a *system database module* stores the (averaged) feature vector together with identifying information [14].

3 Sensing

The absolute matching performance of a biometric system largely depends on the intrinsic properties of its sensor. An example of the variations caused by different input can be seen in the quite different error rates in the FVC2006 [50] for different sensors, causing a change of an order of magnitude in *EER* performance [21]. When selecting a proper sensor for an *online* system (i.e. no *latent* fingerprints, palmprints, etc. are matched, but *live* impressions of the human hand and foot), the first question which needs to be addressed is: *"What should be captured?"* Generally, concerning 2D impressions of the human hand and foot there are two possible choices: (a) *volar* (or *palmar* for hands and *plantar* in the case of feet, respectively) scans, i.e. images referring to either the palm of the hand or the sole of the foot and (b) *dorsal* scans, i.e. images of the upper part of the hand or foot. There are numerous biometric systems providing personal verification based on hand images which rely on different views of the hand and/or different kinds of sensors.

Another question immediately affected by the choice of an appropriate sensor is: *"Can the sensing device distinguish between genuine and forged sensor input?"* If systems are prone to frequent imposter attacks, it may be necessary to check whether the source of an input signal is alive or not. This is called *liveness detection* and may be supported by the choice of sensor [1]. It is, for example, more difficult to fool *thermal* sensors than *optical* sensors (sometimes even an ink print is sufficient). Also *security abilities* such as available encryption for decentralised data acquisition and *compression* influence the choice of a proper sensor [21].

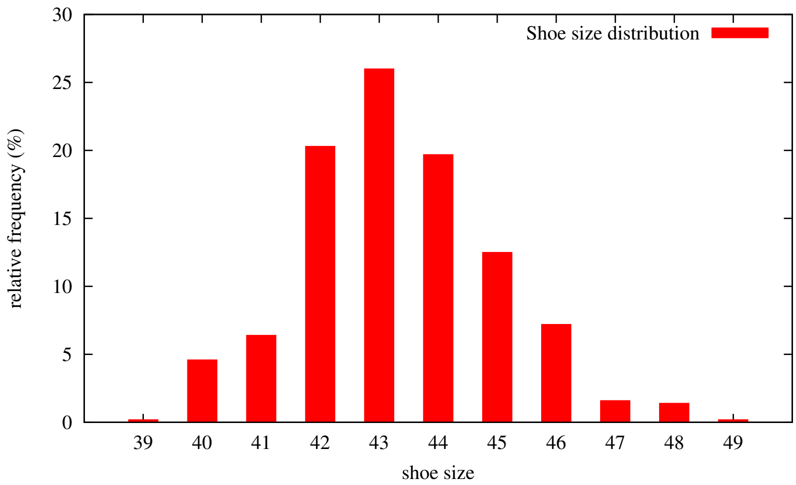
Furthermore, there are a set of user interface and optical system requirements [44]. Capturing devices should be intuitive in a sense that users *know what to do*, even without instructions. This increases throughput and decreases the number of errors due to improper sensor usage when applied in real-world applications. *Real-time capabilities* denote the ability of the biometric system to make the classification decision in an acceptable time. User throughput of single biometric modalities is analysed by Mansfield [23]. According to this study, mean transaction time for optical fingerprint is 9 seconds (minimum 2 seconds) and for hand geometry 10 seconds (minimum 4 seconds). This is on average twice as fast than vein and still 30 percent faster than face. However, when using commercially available scanners at high resolutions, it is clear that these throughput rates are unattainable. Nevertheless, a viable compromise has to be found between accuracy (in terms of resolution) and response time.

Finally, sensor selection also involves the design of the sensor environment. Design goals may be to keep noise, caused by the environment or distortions, to a minimum (e.g. lighting conditions) or to maximise *convenience* (e.g. footprint-based capture in spas using under-water image capture through the use of cameras). For hand and footprint-based measurements these may mean the use of boxes to minimise environmental light (e.g. in [34]) or *pegs* to minimise geometrical distortions.

3.1 Sensor properties

The output of an image-based sensor (sensing hand- or footprints) can be characterised by the following *parameters* (see [21] for details on fingerprint images):

- **resolution**: as measured in dots per inch (dpi). This parameter influences scanning time and storage requirements. While hand geometry can cope with low resolution input, fingerprints typically require resolutions greater than 250 dpi (FBI-compliant scanners are required to operate on resolutions greater than or equal to 500 dpi [21]). Palmprints are captured at a resolution of approximately 125 dpi [44].
- **area**: this refers to the (rectangular) size of the supported area. For most commercially available fingerprint- or palmprint-sensors, there is a tradeoff between area and cost [21]. In case of commercially available flatbed-scanners the situation is different, as the available area usually corresponds to DIN *A*4, a common European format for paperwork introduced by the *Deutsches Institut für Normung*, at a size of 210×297 millimeters. While the majority of human hands are expected to be measurable within this area, footprints deserve further attention. According to a study evaluating foot-related measures conducted by Hofgaertner [8] on 517 industrial workers, 0.2 percent would exceed the available length of 297 mm (a distribution of shoe sizes is given in Figure 3.1). However, since female feet are generally smaller, the part of the users being unable to fit within the available area with respect to the whole population may be even smaller.
- **depth**: typical values are 8-bit or 16-bit grey-scale and 24-bit or 32-bit colour information. Commercially available scanners and cameras usually support both grey-scale and colour image capture. However, this additional information is ignored in most fingerprint, palmprint and hand geometry systems [21, 44, 18], although multi-spectral image sensing (capturing light from different frequencies possibly out of visible light range) can significantly improve input quality and also support hand segmentation [34].
- **geometric accuracy**: this refers to the additional geometric distortion introduced by the sensor due to mirrors or glass plates [44], for example.
- **image quality**: image quality may be affected by dirty sensors, improper (too slight or too intensive) pressure and skin surface (dirt, wet or dry skin). An explicit image quality measure of sensor input may be desired in order to increase matching fidelity [27]. For fingerprint scans several quality measures exist (such as *quality maps* in `mindtct` [51], coherence-based quality [27] or more general *Signal-to-noise-ratio* (SNR) with respect to the original pattern [21]). However, not only textural



shoe (E)	foot (cm)
39	24.77
40	25.40
41	25.72
42	26.04
43	26.34
44	26.98
45	27.62
46	28.26
47	28.90
48.5	29.54
49	29.86

Figure 3.1: Distribution of shoe size, according to a study of 517 male industrial workers aged $18 - 62$ (data from: Hofgaertner [8]) and correspondent foot length (data from: AskNumbers [46]).

quality is essential for the overall performance of a system, but also the ability to extract a stable hand contour, i.e. the sensors ability to support hand segmentation separating grey-scale distributions of fore- and background in the hand image.

3.1.1 Hand geometry sensors

In hand geometry, images are usually captured by flatbed scanners or cameras at a low resolution rate starting at 45 dpi [43]. Hand geometry relies on the reliable extraction of hand contours, thus both *palmar* and *dorsal* hand scans may be applicable. Today, most systems are *palmar*, such as [43, 42, 18, 19, 34, 28]. An example of camera-based dorsal image processing also using the side-view of the hand for geometrical measures is [13].

Typically, hand geometry systems use fixed *pegs* at specific positions between fingers to guarantee a proper alignment [13, 35]. But this has a negative effect on user convenience and *collectability*. More advanced schemes [43, 42] are peg-free and achieve normalisation at the *preprocessing* or *feature extraction* step. As a result of size limitations and immeasurability caused by constrained dexterity, some systems only measure the geometry of a single finger instead of the whole hand [14].

3.1.2 Fingerprint sensors

Fingerprint sensors may be classified according to *sensor type* (such as *optical sensors*, capacitative and thermal *solid-state* and *ultrasound*) [21]. While optical sensors use prisms to sense the difference in light reflectance in order to discriminate between ridges and valleys, solid state sensors use capacitance measures (ridges exhibit higher capacitance than valleys filled with air [53]) or thermal differences. Examples of the generated output by these different types can be found in Maltoni et al. [21]. Fingerprints acquired by flatbed scanners exhibit lower contrast between ridges and valleys than special fingerprint sensors and require additional preprocessing.

Thermal sensors are available as *sweep* sensors (i.e. the user sweeps the finger across the sensor line and the fingerprint is reconstructed from overlapping slices) only [21]. Optical, capacitative and ultrasound sensors are, on the other hand, usually *touch* sensors. Advantages of sweep sensors comprise lower hardware requirements (just a single sensor line is needed) and the absence of latent fingerprints. Drawbacks are additional processing requirements and user training in performing the sweep properly [21]. Available fingerprint databases for performance tests can be obtained from the *Fingerprint Verification Competition* (whilst distribution conditions of the latest FVC2006 [50] database are still being refined, online available data sets of FVC2004 [49] offer 1440 impressions of 12 people for each of the 4 different sensors).

3.1.3 Palmprint sensors

Palmprint sensors use the same techniques as fingerprint sensors [53], thus the same classification principles are applicable. The only difference for palmprint sensors is the larger sensing area covering the 4 different regions of the palm, namely: *upper palm, lower palm, thenar* and *hypothenar*. Since a tradeoff between sensing area and cost exists, they are more expensive (see Bolle at al. [1] for an overview of sensor cost of different modalities). The reliable location of the palm within the hand as part of the preprocessing step involves an extraction of the *region of interest* (ROI) and is frequently achieved using two *fiducial points*, i.e. vertices between fingers as origin points for aligning palmprints (e.g. in [34]). This segmentation process is often supported by acquisition devices using *pegs* (like hand geometry systems, see Zhang's flat platen surface design in [44]). The first online palmprint capturing device was invented at Hong Kong Polytechnic University in 1999 and acquired real-time palmprint images using a white light source and a CCD-camera capturing the reflected image of a hand placed on a glass plate [44]. For online and offline tests, the PolyU database [54] offers more than 7752 grey-scale images corresponding to 386 different palms. The acquired palmprint image obtained by a flatbed scanner is (after registration) quite similar to PolyU sensor data.

3.1.4 Footprint sensors

Kennedy [17] observed, in his forensic research concentrating on barefoot impressions, that feet have many weight bearing areas, which leave an imprint in shoes that may be used for reducing the set of suspects even in the case that no latent ridges are present. This insight can be seen as the birth of footprint-based recognition and led to the development of a hand full of pressure sensing devices (consisting of a matrix of sensels) for footprint-based measures. While some of them are designed as sensing mats, such as the Nitta Big

Mat (used by [26]) or TechStorm Foot Analyzer (US Patent 6331893, used in [16]), the Nitta F-Scan (used by [15]) is designed as a shoe inlet. A drawback of these sensors is the need of segmentation and low resolution. But if texture elements of the sole have to be sensed, such as is done in this work, users are further required to take off socks. Since this may be even more inconvenient and would increase transaction time, I propose a specific application domain, where users already walk on bare feet.

3.2 Proposed sensor

In this work, no specialised biometric sensor is applied, but default commercially available *flatbed* scanning hardware, i.e. an HP 3500c (and HP G3010 for test purposes) scanner. This has several advantages, namely:

- **Availability**: today, a variety of notebooks are already shipped with appropriate *fingerprint sensors* and *web cameras*, which allow the employment of fingerprint and face-biometric applications designed to work on these sensors. Scanners are already widespread - according to a scanner penetration survey by InfoTrends Research Group Inc. [55] in 2001, every third US household is in possession of a document scanner and the scanner market in general was expected to be saturated in 2006. This enables large-scale solutions of biometric measurements via web-based applications, for example.

- **Reproducibility**: since no special hardware is used and no emphasis is placed on the latest technology (a top product has not been selected, but rather a 5-year-old flatbed scanner model designed for a broad commercial market), results are expected to remain stable or even improve when better hardware is employed. Experiments involve untrained users, in order to reflect a realistic capture environment with high reproducibility.

- **Sensor independence**: the goal of this work is to examine hand- and footprint-based biometrics independent of the application of special hardware. Scanners may easily be replaced using *faster* and more accurate devices in order to benefit from technological improvements.

- **Cost**: when biometric systems are applied, special hardware creates a number of costs which are minimised when using hardware designed for large markets.

There are a variety of available scanners with different *interfaces*, namely USB, SCSI, FireWire, and *sensors*, e.g. Charge Coupled Devices (CCD), Contact Image Sensor (CIS) and Complementary Metal Oxide Semiconductor (CMOS). CCDs consist of an array of coupled capacitative sensors for red, green and blue light, and feature the highest depth of field of all three sensor types, whereas CIS scanners do not require mirrors or lenses and therefore minimise geometrical distortions [5]. Even though almost every available scanner provides colour capture, input images are processed as grey-scale bitmaps only.

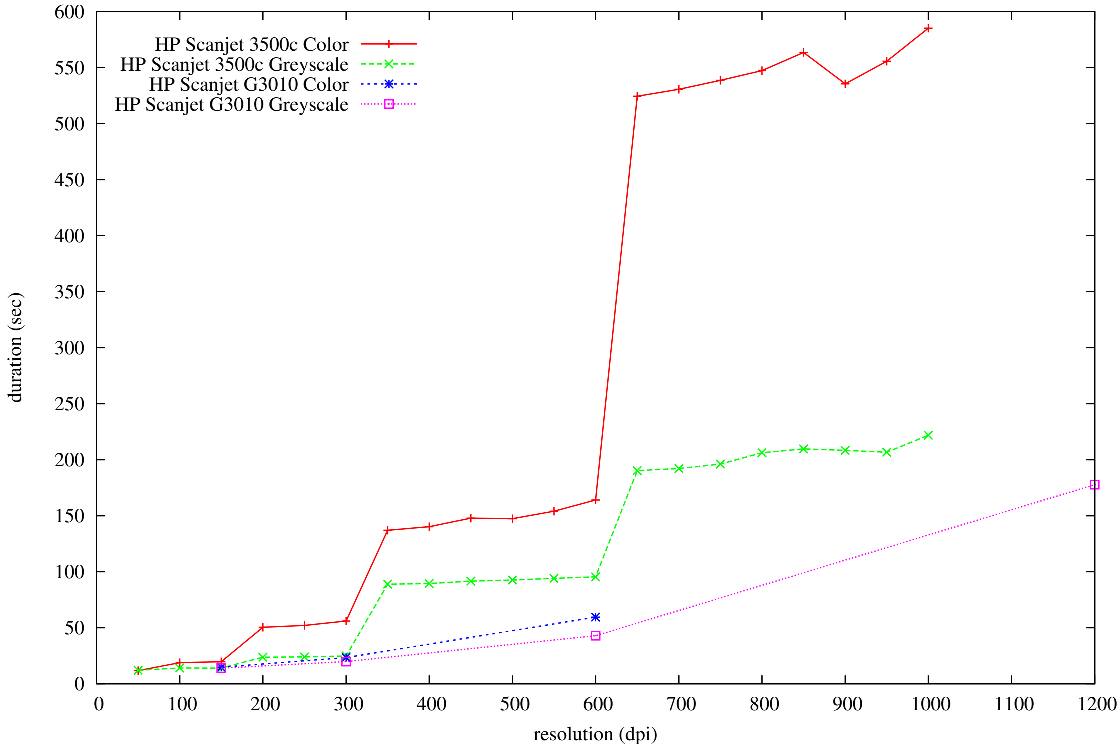
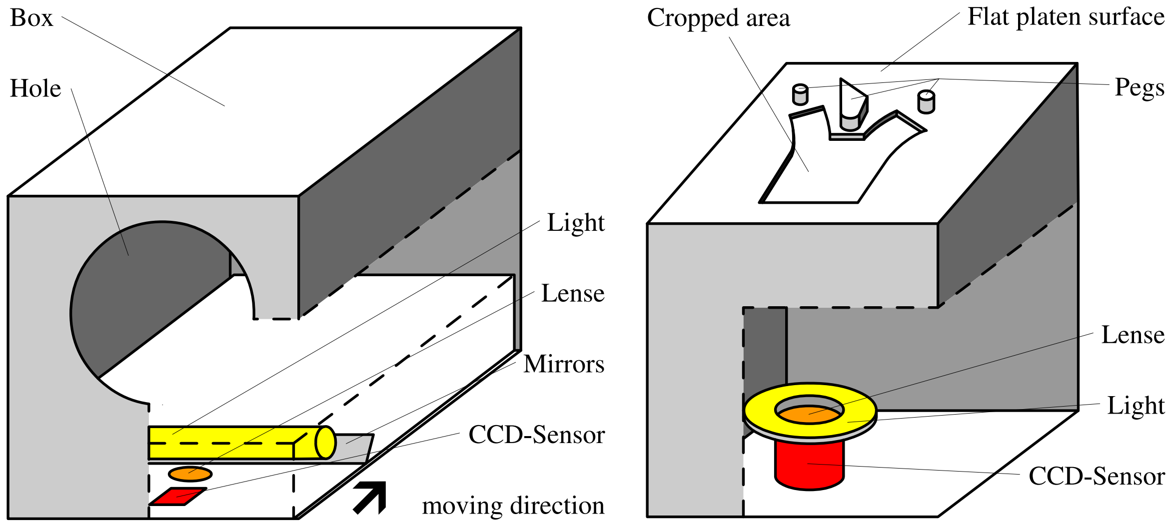


Figure 3.2: Resolution-duration tradeoff for HP Scanjet 3500c and G3010 flatbed scanners.

In order to be able to compare hand and footprint measurements, a single sensor has been selected for both modalities.

HP Scanjet 3500c uses CCD technology, supports the USB interface and represents a 2002-model placed in the low-cost *workgroup* market segment (according to a classification by InfoTrends [5]). It comes with TWAIN driver support and supports an area of 216 × 297 millimeters. An important parameter for image capture is *resolution* since this immediately affects scanning speed, image size and quality. The tradeoff between resolution and duration of a single scan without preview has been analysed for both the employed HP Scanjet 3500c and a new HP Scanjet G3010 model, kindly provided by Hewlett Packard for this project. Figure 3.2 depicts the results of this study indicating points of discontinuity at 300 and 600 dpi for the 3500c model. Scans between these natively supported resolutions are most likely to be down-scaled from the next-highest natively supported resolution. The G3010 driver supported scans at 150, 300, 600 and 1200 dpi only. When comparing colour- and greyscale-mode, it can be seen that colour information further degrades scan speed by roughly $20-40\%$ (G3010) and $70-100\%$ respectively (3500c) within the operational range. For hand-based capture grey-scale acquisition with a resolution of 500 dpi (and 600 dpi for footprints, respectively) has been selected, implying scanning transaction times of roughly 95 seconds for the 3500c model and 40 seconds for G3010. Practical tests have shown that longer capturing times are not only inconvenient, but also may cause unintended displacement of the measured body part.



(a) Employed scanning device.

(b) Zhang's palmprint acquisition device.

Figure 3.3: Comparing (a) employed scanner-based sensor and (b) Zhang's camera-based palmprint capture device (see: Zhang [44]).

Finally, for minimising environmental light, the scanner is situated in a box with a round hole at one end for hand insertion, see Figure 3.3(a). Footprint-based data acquisition may also be supported using a box with a hole at the upper frontal edge, but has been executed in a darkened environment without a box. It is clear to the author, that instant capture devices, like Zhang's palmprint acquisition sensor, for example, depicted in Figure 3.3(b) have certain advantages over the proposed solution with respect to convenience, especially in commercial proprietary solutions (e.g. when installed as an embedded authentication system without needs for standardisation or inter-operability). Nevertheless, the proposed approach will be shown to produce satisfying results at low cost.

4 Preprocessing

While preprocessing steps are usually treated superficially in biometric system design, they are nevertheless one of the most important steps to achieve fault-tolerant and sensor-independent systems. A strong relationship exists between preprocessing and the *FTA* rate. When *alignment* (such as iris segmentation in iris recognition systems) can not be achieved or *image quality checks* indicate bad quality, templates are usually rejected and a reacquisition of the biometric sample is necessary [21]. Preprocessing can drastically increase image quality and thus reduce *FTA* and furthermore *FMR* and *FNMR* values. Examples of preprocessing steps increasing image quality are:

- **Texture enhancement** using, for example, multi-spectral image analysis [34], oriented Gabor-filters (for dry or wet fingerprints) [21], image normalisation in mean and variance [18] or more generic approaches targeting histogram equalisation;
- **Image alignment** by achieving normalisation in direction and position. Using moments, Nakajima et al. [26] could improve their Euclidian-distance-based footprint recognition method on raw images by roughly 55%. Yoruk et al. [42] normalize hands using re-alignment of individual fingers with texture blending. The second approach has not been implemented in this work, since most of the hand-based features operate on local single fingers instead of the entire hand image and for footprints close-fitting toes constitute problems.
- **Image segmentation** and masking areas of low quality, which may be suppressed for feature extraction (e.g. *quality maps* in [51]). Segmentation of the original image may also be helpful, when large amounts of data are processed, such as is the case in hand- and footprint-based recognition. An area of DIN A4 at 500 dpi corresponds to 4252×5846 pixels and takes (uncompressed as 8-bit grey-scale image) approximately 25 MB of memory, whereas the average *bounding box* (the smallest circumscribed axis-aligned rectangle) of 439 normalised hands of 86 people captured for experiments in this work (see Chapter 7) only covers 47 percent of this area. Early localisation of the hand may decrease memory requirements and increase computation speed significantly, especially when filters with high complexity are applied to the entire image.

Sometimes, a clear distinction between preprocessing steps and feature extraction is not feasible, when no clear interface between both processing steps can be defined. In this work, preprocessing unifies common alignment steps executed by most of the applied feature extractors in order to avoid additional overheads. Due to anatomical differences of hands and feet, preprocessing steps differ for rotational alignment and salient point detection.

4.1 Overview

The following information is provided by the preprocessing module in both hand- and footprint-based systems:

- **Normalised hand/foot image**: this image contains the measured body part with the following normalisation: (a) segmented from background, (b) aligned in direction and position and (c) with removed arm and leg parts.
- **Contour information**: in order to identify fingers/toes within input images, preprocessing comes along with hand/foot contour information including landmark indices, such as *salient points* (i.e. intra-finger/intra-toe valleys and finger/toe tips).

This definition implies that each feature extractor is effectively provided with the same information of the whole hand and may discard unused information within a feature-extractor-dependent processing step, before actual extraction takes place. In the strict multibiometric sense of this definition, the employed scenario corresponds to *multiple matcher* [14], whereas a slight modification of the preprocessor providing each modality (fingerprint, palmprint, hand geometry) with already cropped input would classify the scenario as *multiple biometrics* or *multiple units* (in case of fingerprints when each fingertip is processed independently by the same matcher).

Hand- and footprint-based preprocessing consists of three consecutive steps:

1. **Segmentation**: as introduced before, hand and foot only cover parts of the sensor image. Thus, a coarse segmentation, by means of detecting the bounding box of the actual handprint/footprint, is applied to increase computation speed of more complex feature extraction. Segmentation typically also includes *binarisation* for masking the background parts in the resulting image using either Otsu's Thresholding [29] (for hand images) or Canny edge detection [2] and thresholding (for footprints).
2. **Rotational alignment**: when using *peg-free* systems like [43, 18], rotational alignment has yet to be achieved. A frequently applied method is *rigid hand registration using statistical moments* estimating the best-fitting ellipse, a technique from face-detection [36]. While [43] also proposes finger-alignment for binary images, this normalisation is not implemented in the proposed system due to textural distortions caused by different spreadings of fingers. In contrast to more cooperative capture (like in [18]), problems were experienced using ellipse fitting for hands with spread fingers, thus a more elaborate approach was introduced using the palm *borderline*, the least-squares approximated line between wrist and little finger.
3. **Contour extraction and salient point detection**: For hand-images, intra-finger valleys (and peaks) can be detected as minima (and maxima) of the *radial distance function* using a reference point at the wrist [43]. In my experiments, this method is found to be fragile with respect to bad background segmentation, rings (causing contour artifacts) and high resolutions. In this work a new method is proposed, improving intra-finger valley candidates (provided by the radial distance function)

with line-approximation of individual finger boundaries and intersection of the intra-finger angle bisector with the contour. Finger peaks can then be determined by matching each individual finger with the best fitting ellipse and intersecting the major axis with the finger contour. For footprints, intra-toe valley detection is achieved, improving salient point candidates within sectors centered at the center of mass of the footprint.

4.1.1 Hand preprocessing

The following section technically describes processing steps in order to achieve rotational and translational invariance of hand images. Further details on the employed techniques are presented in separate sections. Employed preprocessing steps for hand images are illustrated in Listing 4.1 using Java-like pseudo-code.

Listing 4.1: Pseudo-code representation of the Hand-Preprocessing algorithm.

```
void PreprocessHand(Bitmap hand)
{
   final int max_iterations = 3;
   Bitmap tmp, bin;
   ContourInfo cinfo;
   Point center;
   float rotation;

   tmp = CropInputImage(hand);
   bin = HandMask(tmp);
   hand = Multiply(tmp,bin); // cropped and segmented hand
   tmp = bin; // start to find arm parts in binary image
   for (int i = 1; i < max_iterations; i++){
      center = CalculateCenterOfMass(tmp);
      rotation = CalculateRotation(tmp,center);
      tmp = Rotate(center,rotation,tmp);
      tmp = IterativeArmRemoval(center,rotation,tmp,bin);
   } // within tmp all arm parts are removed
   cinfo = CalculateHandSilhouette(tmp);
   center = GetMiddleRingFingerValley(cinfo);
   rotation = GetPalmBoundaryLineRotation(cinfo,center);
   tmp = Rotate(center,rotation,tmp); // tmp is normalized
   tmp = IterativeArmRemoval(center,rotation,tmp,bin);
   cinfo = CalculateHandSilhouette(tmp);
   AlignHandImage(hand,cinfo,center,rotation);
   AlignHandContour(cinfo); // hand and contour are aligned
   Save(hand);
   Save(cinfo); // normalised image and contour are stored
}
```

Custom functions defined in Listing 4.1 refer to the following execution of *tasks*:

- `CropInputImage(hand)`: this involves (a) image downscale by factor 10, (b) gaussian blur using the parameters $\sigma = 2, s = 3$, (c) binarisation using Otsu's Threshold, (d) region size filtering, (e) bounding box determination and (f) cropping of the initial hand image according to the upscaled bounding box.
- `HandMask(tmp)`: in order to mask background pixels, a downscaled (factor 3) and blurred version ($\sigma = 2, s = 7$) of `tmp` is (a) binarised, (b) morphologically dilated and (c) size filtered.
- `Multiply(tmp,bin)`: This filter aligns both images in size and performs pixel-wise multiplication of both operands.
- `CalculateCenterOfMass(tmp)`: this method refers to inertial moment calculation, in particular to the calculation of the hand's *center of mass*.
- `CalculateRotation(tmp,center)`: this function estimates the inclination of the major axis of the best-fitting ellipse.
- `Rotate(center,rotation,tmp)`: rotates the image `tmp` using the angle `rotation` and origin `center`.
- `IterativeArmRemoval(center,rotation,tmp,bin)`: the rotated `tmp` image is used to estimate the *wrist line* with a top-down scanning algorithm based on thresholding of vertical slices (choosing the slice where the number of hand pixels falls below 50% of the slice with the maximum number of hand pixels). This line is transformed to a half-space mask (masking the arm in the `bin` image), which is multiplied with `bin` and thus clips visible arm parts.
- `CalculateHandSilhouette(tmp)`: the hand-object in `tmp` is a path-connected space (in terms of 8-neighbourhood of pixels). Therefore, the contour silhouette can easily be extracted by a counter-clockwise scan algorithm with a starting point at the center of the wrist.
- `GetMiddleRingFingerValley(cinfo)`: salient points are estimated from the contour using local maxima and minima of the *distance-function* with respect to the wrist reference point. These initial candidates are refined cutting the intra-finger angle bisector with the contour and performing ellipse fitting on individual fingers.
- `GetPalmBoundaryLineRotation(cinfo,center)`: this method extracts the rotation of the line approximating the palm boundary sequence for hand alignment.
- `AlignHandImage(hand,cinfo,center,rotation)`: since all of the employed processing steps have been performed on a sub-sampled binary image, the `hand` image is updated (with respect to arm removal, rotational- and displacement-alignment).
- `AlignHandContour(cinfo)`: finally, the calculated hand contour is aligned and restricted to its bounding box.
- `Save(hand)`, `Save(cinfo)`: stores normalised image `hand` or contour `cinfo` to disk.

4.1.2 Foot preprocessing

Despite the fact that hand and foot have many anatomical properties in common, their preprocessing requirements are quite different. This is caused by:

- **elliptical foot shape**: due to its natural elliptical form and the absence of parts of the leg in the scan (which may have to be removed, like arms in case of hands), best-fitting ellipse-matching for rotational invariance performs well. Also, the problem of *ulnar or radial* abducted *phalanges* causing large variations when matching the footprint using moments is not present in this application domain. Thus, there is no need to further define processing steps with respect to rotational normalisation.
- **close-fitting or dorsal extended toes**: due to common abnormalities (see [8]: e.g. *hammer toes, claw toes, hallux valgus*), unintended *dorsal extension* of toes or close-fitting toes, it is difficult to extract inter-toe-valleys from the foot silhouette. Instead, candidate points on the silhouette curve are refined within the cone centered in the foot's center of mass with a central angle defined by the horizontal distance of the candidate points. Within this sector, the southern-most end-point of the inter-toe edge is selected as the true inter-toe valley.

The employed processing tasks for foot-based preprocessing are presented in pseudo-code notation in Listing 4.2.

Listing 4.2: Pseudo-code representation of the Foot-Preprocessing algorithm.

```
void PreprocessFoot(Bitmap foot)
{
   Bitmap tmp, bin;
   ContourInfo cinfo;
   Point center;
   float rotation;

   tmp = CropInputImage(foot);
   bin = FootMask(tmp);
   foot = Multiply(tmp,bin); // cropped and segmented foot
   tmp = bin;
   center = CalculateCenterOfMass(tmp);
   rotation = CalculateRotation(tmp,center);
   tmp = Rotate(center,rotation,tmp); // tmp is normalized
   cinfo = CalculateFootSilhouette(tmp);
   AlignFootImage(foot,cinfo,center,rotation);
   AlignFootContour(cinfo); // foot and contour are aligned
   Save(foot);
   Save(cinfo); // normalised image and contour are stored
}
```

At this point I will discuss all employed custom functions in detail:

- `CropInputImage(foot)`: this function can be implemented in the same way as the corresponding function for hand images in fully-automated environments. An additional border of tolerance of at least 1 pixel is needed in order to be able to detect contour edges located at the border with the following processing steps. But the function has not been implemented due to a semi-automated acquisition procedure using the *preview*-function of the TWAIN driver and manual selection of the footprint area, in order to save acquisition time (smaller scanning area).

- `FootMask(tmp)`: in order to preserve edges for accurate shape feature extraction, (a) Canny edge detection is employed, (b) the interior of the foot is filled using binary thresholding, (c) the resulting image is subjected to morphological dilation, (d) region size filtering and (e) morphological erosion.

- `Multiply(tmp,bin)`: unchanged, see Section 4.1.1 for hand preprocessing.

- `CalculateCenterOfMass(tmp)`: unchanged, see Section 4.1.1 for hand preprocessing.

- `CalculateRotation(tmp,center)`: unchanged, see Section 4.1.1 for hand preprocessing.

- `Rotate(center,rotation,tmp)`: unchanged, see Section 4.1.1 for hand preprocessing.

- `CalculateFootSilhouette(tmp)`: foot silhouette extraction is performed in an almost identical way to hand images, except that the starting point is located on the outside part of the foot at 50% of the foot's height.

- `AlignFootImage(foot,cinfo,center,rotation)`: since all of the employed processing steps have been performed on a sub-sampled binary image, the `foot` image is updated (with respect to rotational- and displacement-alignment).

- `AlignFootContour(cinfo)`: the calculated foot contour is aligned and restricted to its bounding box.

- `Save(foot)`, `Save(cinfo)`: stores normalised image `foot` or contour `cinfo` to disk.

4.2 Binarisation

Binarisation is a *thresholding problem* and targets the *segmentation* of the input image, which precedes image *analysis* supporting the extraction of higher-level image information, such as object contours or features. More formally, $m \times n$ images may be treated as $2D$ grey-scale intensity functions of two variables assigning to each *pixel* $\mathbf{P}$ at position (x, y) its grey level $B(x, y)$ within $\{0, \ldots, l\}$ ($l = 255$ for 8-bit grey-scale images).

(a) Detail in original scan.

(b) Detail after gaussian blur.

(c) After Otsu's Thresholding.

Figure 4.1: Result of binary thresholding using Otsu's method on the gaussian blurred input hand.

Definition 4.2.1 (Image). *An $m \times n$ image is a function:*

$$B(x, y) : \{1, \ldots, m\} \times \{1, \ldots, n\} \to \{0, \ldots, l\}.$$

Without loss of generality, an object coordinate system located in the upper left corner is assumed.

The task of *binarisation* can simply be regarded as a reduction of the intensity levels to the set $\{0, 1\}$, i.e. $l = 1$ for a determination of background and foreground (hand- or foot-) pixels.

While many techniques based on the statistics of grey-level *histograms* (i.e. graphical illustrations of tabulated frequencies of intensity values) exist [20], Otsu's method has been around since 1979 and produces satisfying results as illustrated in Figure 4.1. It has proven to be a good alternative for hand geometry [18].

4.2.1 Fast Otsu Thresholding

Let p_i refer to the probability of grey level i within the image B and let μ be the mean intensity of the image. Otsu's Method [29] is a *global thresholding* algorithm, i.e. a global threshold $t \in \{0, \dots l+1\}$ is selected, such that the new binary image B' is given by:

$$B'(x,y) = \begin{cases} 1 & \textit{if } B(x,y) \geq t, \\ 0 & \textit{otherwise.} \end{cases} \quad . \tag{4.1}$$

Since light reflected by the sole of the foot or palm is converted into pixels with higher intensities, this leads to a good separation of background (represented by the class C_1 with grey levels $\{0, \dots, t-1\}$) and sole/palm (represented by the class C_2 with grey levels $\{t, \dots, l\}$). For a given threshold t and class $k \in \{1, 2\}$, let ω_k refer to the class probability and μ_k refer to the mean for C_k [20]:

$$\omega_k := \sum_{i \in C_k} p_i; \tag{4.2}$$

$$\mu_k := \frac{1}{\omega_k} \sum_{i \in C_k} i \; p_i. \tag{4.3}$$

Otsu's method chooses t^*, such that the between-class variance σ_B^2 is maximised [20]:

$$\sigma_B^2(t) := \sum_{k=1}^{2} \omega_k (\mu_k - \mu)^2 = \sum_{k=1}^{2} \omega_k {\mu_k}^2 - \mu^2; \tag{4.4}$$

$$t^* = \underset{1 \leq t \leq l}{\arg\max} \{\sigma_B^2(t)\}. \tag{4.5}$$

Since obtaining ω_k and μ_k for each threshold t involve many recurring operations, an improved fast threshold search can be implemented using lookup-tables of recursively calculated zeroth- and first-order moments of pixel intensities u to v, as illustrated in [20].

4.2.2 Canny-edge-detection based binarisation

In order to preserve foot edges for accurate shape feature extraction within the footprint system, first Canny edge detection [2] with binary thresholding on the original image B is employed. This step keeps the most significant edges only which reliably represent foot contours. Then, within the obtained edge image B_1, the interior of the foot is filled using binary thresholding on B, i.e.

$$B_2(x,y) := \max(B'(x,y), B_1(x,y)); \tag{4.6}$$

where B' is obtained by global thresholding. Then, the binary image B_2 is subjected to morphological dilation using a square structuring element S in order to close the boundary:

$$B_3 := B_2 \oplus S = \{(x,y)|S_{xy} \cap B_2 \neq \emptyset\}; \tag{4.7}$$

where S_{xy} denotes a shift of S by (x,y). Still, the image may contain more than one connected component. Since both hands and feet can be supposed to consist of only one connected space (in terms of 8-neighbourhood), all other (smaller) connected components are removed from B_3 using a filtering technique. Finally, morphological erosion is employed on the obtained image B_4:

$$B_5 := B_4 \otimes S = \{(x,y)|S_{xy} \subseteq B_4\}. \tag{4.8}$$

4.3 Rotational Alignment

Even if the arm is inserted straight into the scanning device, ulnar and radial abduction cause slight rotations. These rotations have to be corrected by a preprocessing algorithm. One method to perform rotational alignment is matching hand- and footprints with the best-fitting ellipse, proposed in [36].

As can be seen in Figure 4.2, results for footprints are satisfying, due to their elliptical form. Hand-prints are more difficult to match reliably, especially because of different spreadings of fingers.

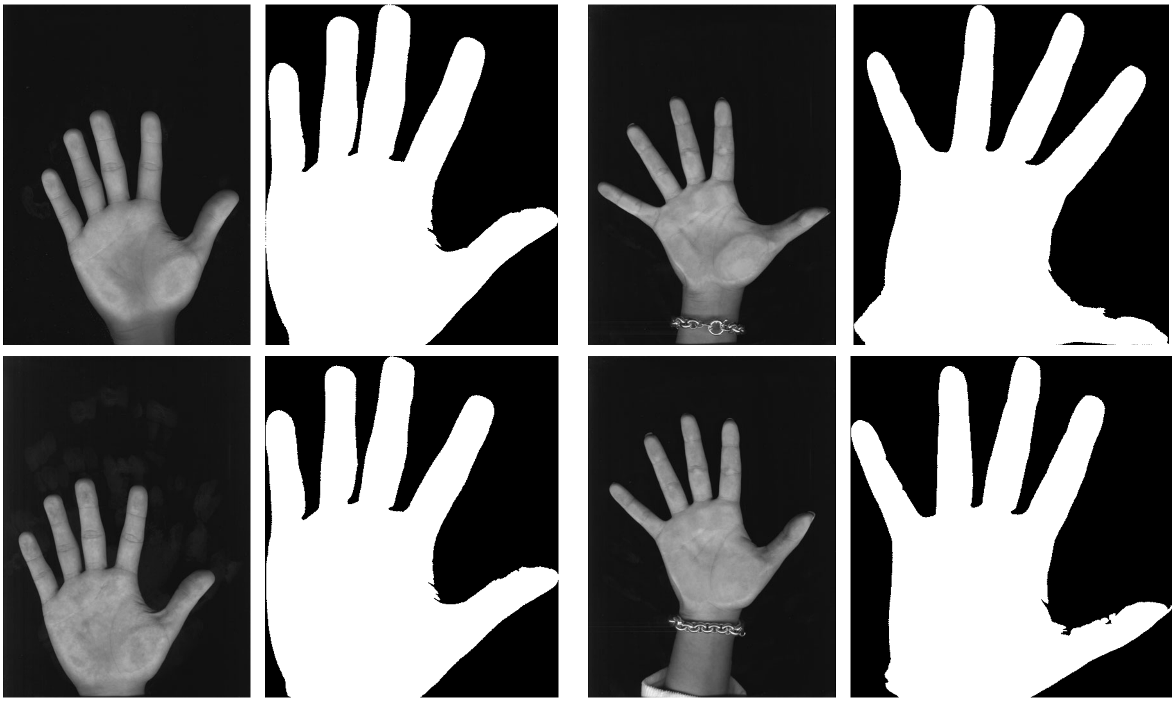
4.3.1 Moment-based alignment

Moment-based alignment has been used in face recognition [6] and hand recognition systems [18] and has also proven to be successful for alignment of footprints [26]. First, the center of the ellipse is calculated (see [6]):

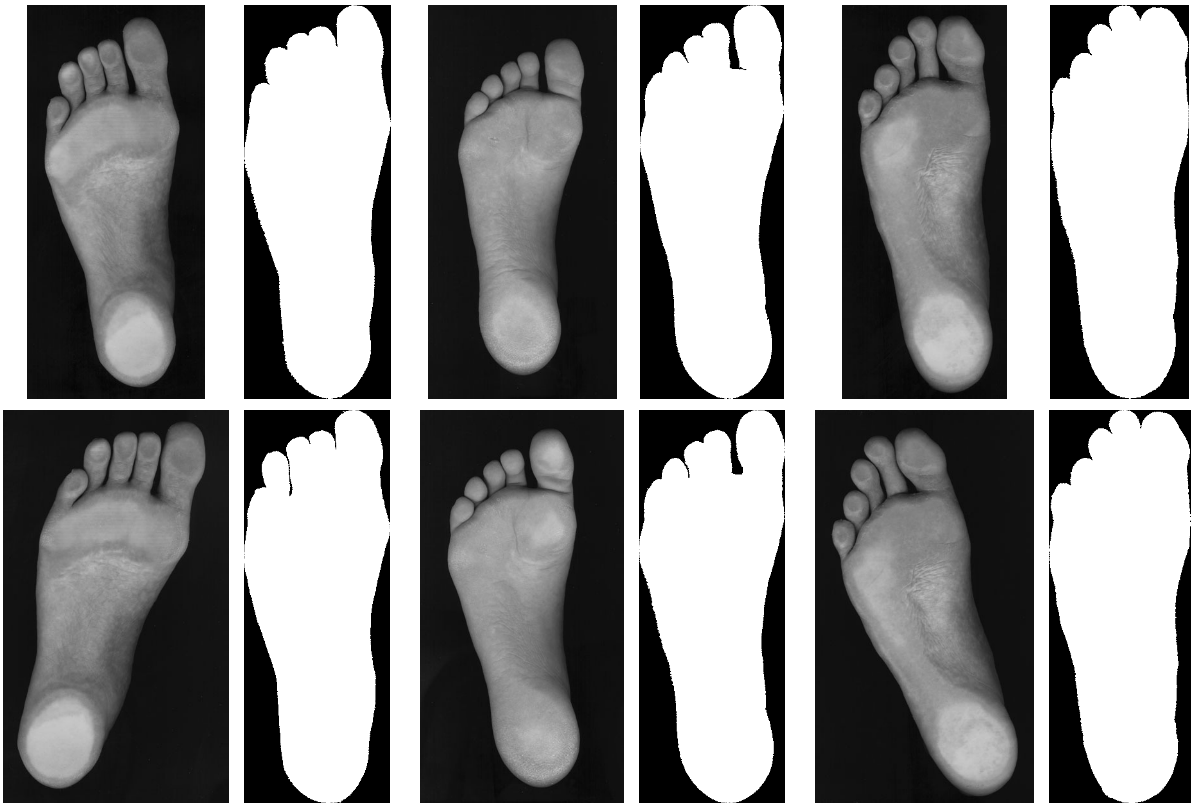
Definition 4.3.1 (Center of mass)**.** *Given the binary $m \times n$ image B and the number a of white pixels representing the object in the image, then the center of mass, $\boldsymbol{C} = (\overline{x}, \overline{y})$, is estimated as follows:*

$$\overline{x} = \frac{1}{a}\sum_{i=1}^{m}\sum_{j=1}^{n} iB(i,j), \qquad \overline{y} = \frac{1}{a}\sum_{i=1}^{m}\sum_{j=1}^{n} jB(i,j).$$

The goal is to calculate the angle Θ between y-axis and the major axis of the best matching ellipse (see [6]):



(a) Examples of rotationally aligned hands using moments.



(b) Examples of rotationally aligned feet using moments.

Figure 4.2: Results of moment-based rotational alignment on hand and foot images.

$$\begin{aligned}
\Theta &= \frac{1}{2}\arctan\left(\frac{2\mu_{1,1}}{\mu_{2,0}-\mu_{0,2}}\right); && (4.9)\\
\mu_{2,0} &= \sum_{i=1}^{m}\sum_{j=1}^{n}(x_{ij}-\overline{x})^2 B(i,j); && (4.10)\\
\mu_{1,1} &= \sum_{i=1}^{m}\sum_{j=1}^{n}(x_{ij}-\overline{x})(y_{ij}-\overline{y}) B(i,j); && (4.11)\\
\mu_{0,2} &= \sum_{i=1}^{m}\sum_{j=1}^{n}(y_{ij}-\overline{y})^2 B(i,j). && (4.12)
\end{aligned}$$

4.3.2 Palmprint-boundary based alignment

Another technique used for aligning inked palmprints is a definition of a palmprint *coordinate system* and rotating and translating the palmprint to a located origin and determined y-axis [44]. This method is adapted to binary hand images in this work and involves the following steps:

1. **Determination of the origin**: while inked palmprints exhibit texture information permitting the definition of the origin as the outer end-point of the heart line [44], binary hand images allow the definition of a similar invariant point using geometrical measures, i.e. the intra-finger valley $\mathbf{V}_2 \in \mathbb{R}^2$ between index and middle finger.

2. **Determination of the y-axis**: in analogous manner to [44], the y-axis is defined as the outer palm's boundary (which can be approximated by a straight line) translated to the origin. Start and end points, however have to be defined carefully in order to get the same portion of the outer boundary for different hand-prints. The *palm boundary sequence* is approximated by a line using the *method of least squares* in order to obtain a normalised orientation.

Definition 4.3.2 (Palm boundary sequence)**.** *Let $\boldsymbol{T}_5 \in \mathbb{R}^2$ denote the finger-tip of the little finger and $\boldsymbol{V}_4 \in \mathbb{R}^2$ the corresponding intra-finger valley. Furthermore, let $\langle \boldsymbol{S}_0, \ldots, \boldsymbol{S}_n \rangle$ denote the (counter-clockwise) hand silhouette sequence. Then, the palm boundary sequence is the subsequence $\langle \boldsymbol{S}_k, \ldots \boldsymbol{S}_l \rangle$ with:*

$$k = \max\{x \in \{0, \ldots, n\} : \| \boldsymbol{S}_x - \boldsymbol{T}_5 \| \leq \| \boldsymbol{V}_4 - \boldsymbol{T}_5 \|\};$$

$$l = \max\{x \in \{0, \ldots, n\} : \| \boldsymbol{S}_x - \boldsymbol{S}_k \| \leq \overline{t}\};$$

where $\overline{t}$ denotes the average length of fingers for this hand.

4.4 Silhouette extraction

Having a binary $m \times n$ input image B with only one connected component as the hand or foot object, the counter-clockwise contour polygon $\langle \mathbf{S}_0, \ldots, \mathbf{S}_n \rangle$ is traced as follows:

1. **Estimation of starting point $\mathbf{S}_0$**: depending on whether hands or feet are processed, the starting point is found when intersecting the wrist or outside boundary line of the foot with lines parallel to y-axis (and x-axis, respectively) through the center of the (cropped) image. For hands, one can define:

$$\mathbf{S}_0 = \begin{pmatrix} \frac{m}{2} \\ \max\{y \in \{1, \ldots, n\} : B(\frac{m}{2}, y) = 1 \wedge B(\frac{m}{2}, y+1) = 0\} \end{pmatrix}; \tag{4.13}$$

   For feet, the starting point is selected as:

$$\mathbf{S}_0 = \begin{pmatrix} \min\{x \in \{1, \ldots, m\} : B(x-1, \frac{n}{2}) = 0 \wedge B(x, \frac{n}{2}) = 1\} \\ \frac{n}{2} \end{pmatrix}. \tag{4.14}$$

2. **Contour tracing**: having selected $\mathbf{S}_0$, contour tracing involves a local decision of the next contour pixel $\mathbf{S}_{k+1}$ based on $\mathbf{S}_k$ and the current direction index d_k within the sequence $\langle \mathfrak{d}_m \rangle$ referring to *left-up, left, left-down, down, right-down, right, right-up, up* directions in counter-clockwise order:

$$\langle \mathfrak{d}_m \rangle = \langle \begin{pmatrix} -1 \\ -1 \end{pmatrix}, \begin{pmatrix} -1 \\ 0 \end{pmatrix}, \begin{pmatrix} -1 \\ 1 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 \\ -1 \end{pmatrix}, \begin{pmatrix} 0 \\ -1 \end{pmatrix} \rangle. \tag{4.15}$$

   Now, $\mathbf{S}_{k+1}$ is selected with respect to the next counter-clockwise non-zero pixel within an eight-neighbourhood starting at the pixel in direction of d_k, i.e. the predecessor of $\mathbf{S}_k$. Finally, the resulting found direction is reversed, such that d_{k+1} points towards the new predecessor $\mathbf{S}_k$ of $\mathbf{S}_{k+1}$.

$$d_{k+1} = (d_k + 4 + \min\{i \geq 1 : B(\mathbf{S}_k + \mathfrak{d}_{(d_k+i) \ mod \ 8}) = 1\}) \ mod \ 8; \tag{4.16}$$

$$\mathbf{S}_{k+1} = \mathbf{S}_k - \mathfrak{d}_{d_{k+1}}. \tag{4.17}$$

   The iteration stops, when $\mathbf{S}_n = \mathbf{S}_0$ is reached. Initial directions are $d_0 = 3 = down$ for hands and $d_0 = 1 = left$ for feet.

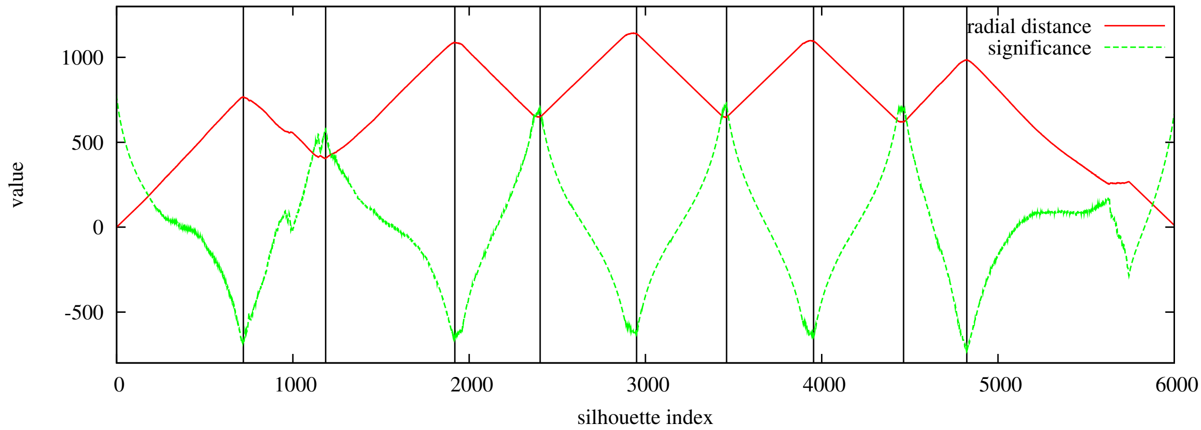


Figure 4.3: Sample radial distance and significance function with vertical lines indicating candidates for finger peaks and valleys.

4.5 Salient point detection

In order to normalise hand images which exhibit more variation than feet after initial rotational alignment using moments, indices of *salient points*, i.e. finger peaks and valleys, have to be extracted from the obtained silhouette polygon [28, 42, 43]. For footprints, detection of toe peaks and valleys is primarily used for geometry feature extraction (including length of toes), thus this processing step may also be migrated to geometry feature extraction in this case (since it is not used for normalisation purposes). Since salient points of footprints are more difficult to extract (due to tightly fitting toes), I will first cover salient point extraction for hand images and then discuss adaption for footprints.

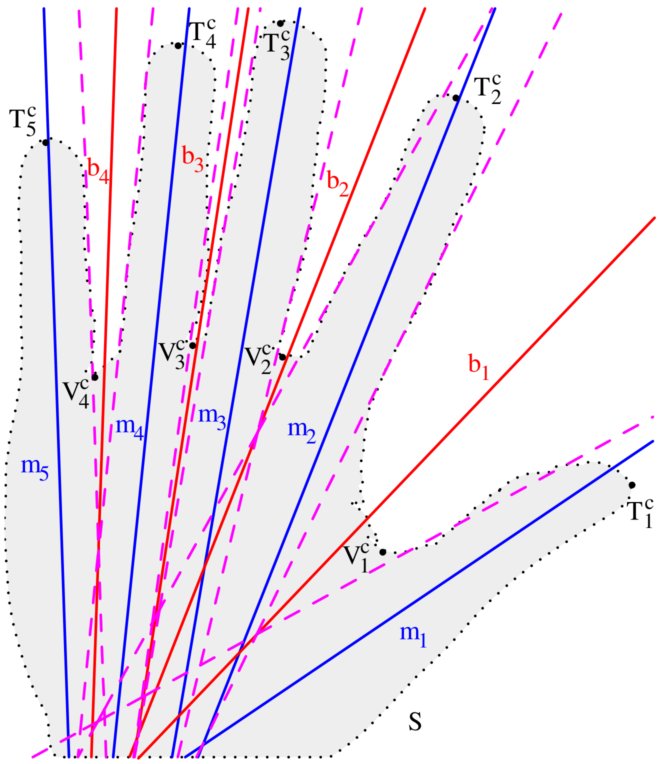
Typically, salient points are extracted from the hand contour $\langle \mathbf{S}_0, \ldots, \mathbf{S}_n \rangle$ by finding local minima and maxima of the corresponding *radial distance function* [43]:

$$R(i) := \| \mathbf{S}_i - \mathbf{P} \| \quad \forall i \in \{0, \ldots, n\}; \tag{4.18}$$

with respect to a reference point $\mathbf{P}$ at the wrist region. The implemented algorithm uses $\mathbf{P} = \mathbf{S}_0$ for hands. In the case of footprints, the center of mass is a good reference point to obtain contour peaks and valleys: $\mathbf{P} = \mathbf{C}$. Instead of the radial distance function, in this work a *significance function* I estimating local bending is defined (see Figure 4.3):

$$I(i) := \sum_{j=1}^{\frac{n}{15}} \frac{R(i - j \ mod \ n) - R(i)}{j} - \frac{R(i) - R(i + j \ mod \ n)}{j}. \tag{4.19}$$

Due to their anatomical form, finger valleys have high positive significance, while peaks can be expected to exhibit negative significance. Local minima and maxima within the interval $I_0 := [1, n]$ are extracted iteratively by selecting the global maximum of I within I_k, $\max(I, I_k)$, and escaping the window with positive values around the maximum, i.e.:



(a) Hand salient point refinement.

(b) Foot salient point refinement.

Figure 4.4: Position of (a) initial peak and valley candidates, finger axes, finger boundaries and inter-finger bisecting lines within a normalised (binary) hand and (b) initial peak and valley candidates and triangular valley-refinement areas within a normalised (binary) foot.

$$I_{k+1} = I_k \backslash \{x \in \{1,\ldots,n\} : I(y) > 0 \, \forall y \in [\min\{x, \max(I, I_k)\}, \max\{x, \max(I, I_k)\}] \cap \mathbb{N}\}. \quad (4.20)$$

Having extracted the five finger peak candidates $\mathbf{T}_k^c \in \mathbb{R}^2$ (ordered according to their relative position within $\langle \mathbf{S}_0, \ldots, \mathbf{S}_n \rangle$) from $\max(I, I_k), k = 0, \ldots 4$, valleys $\mathbf{V}_k^c$ refer to absolute minima between two consecutive peaks.

4.5.1 Salient point refinement for hands

The method introduced before is found to produce reliable candidate points, but due to high resolution and segmentation errors caused by rings and scanner quality, often valleys are displaced. Figure 4.4(a) illustrates refinement methods and originally extracted valley and peak candidates of the corresponding significance function in Figure 4.3. Candidate points for hands are refined as follows:

1. **Finger peak refinement using best-fitting-ellipse matching for each finger**: since fingers have elliptical shape, finger peaks $\mathbf{T}_k \in \mathbb{R}^2$ can be detected more reliably by estimating the major axis m_k of the best-fitting ellipse for each individual finger, using the rotation techniques described earlier in this chapter:

$$\mathbf{T}_k := \mathbf{S}_j, j = \underset{1 \leq i \leq n}{\arg\min}\{d(i)\}, d(i) = \begin{cases} \infty & \textit{if } \textit{dist}(\mathbf{S}_i, m_k) > \textit{thres}, \\ \parallel \mathbf{S}_i - \mathbf{T}_k^c \parallel & \textit{otherwise}. \end{cases} \tag{4.21}$$

2. **Finger valley refinement using least-squares approximation of the finger boundary**: for a given, normalised finger (centered within the center of mass and using the major and minor axes of the best-fitting ellipse as coordinate axes), the left (and right) *finger boundary line* can be defined as the least-squares approximated line of contour members within the third (and fourth, respectively) quadrant, for which the x-coordinate lies within a 90% interval around the mean value for this quadrant (to eliminate outliers). Then, for two finger boundary lines bordering an inter-finger valley, the bisecting line b_k is selected, which divides the smaller angle (or if the enclosed angle α lies in between $[60°, 90°]$, the line separating both adjacent finger tips into two half-spaces). In case of both lines being parallel, the line with equal distance to both boundary lines is selected.

$$\mathbf{V}_k := \mathbf{S}_j, j = \underset{1 \leq i \leq n}{\arg\min}\{d(i)\}, d(i) = \begin{cases} \infty & \textit{if } \textit{dist}(\mathbf{S}_i, b_k) > \textit{thres}, \\ \parallel \mathbf{S}_i - \mathbf{V}_k^c \parallel & \textit{otherwise}. \end{cases} \tag{4.22}$$

4.5.2 Salient point refinement for feet

For footprints, modifications are necessary in order to obtain true inter-toe valleys. Candidate points are selected in a similar manner, i.e. points of high curvature with respect to a significance function are extracted. However, as illustrated in Figure 4.4(b), inter-toe valleys on the silhouette are less distinctive. Since toes may occlude each other, intra-toe valleys are frequently *not* even members of the contour polygon, as is the case in Figure 4.4(b). The following processing steps are executed for footprints:

1. **Estimation of initial valley candidates**: using a significance function based on the average angle of a contour point (for simplicity-reasons, the x-monotone subcontour is selected for candidate extraction) with its neighbourhood, valley candidates $\mathbf{V}_k^c \in \mathbb{R}^2$ are extracted and ordered according to their relative position within $\langle \mathbf{S}_0, \ldots, \mathbf{S}_n \rangle$.

2. **Estimation of toe peaks**: since peaks of toes are typically not occluded and are usually members of the contour, they can be identified as local maxima of the contour between two consecutive valley candidates.

3. **Valley refinement within cones**: true valleys may be off-line, i.e. not members of the contour. Therefore, a triangular search area $\triangle_k$ is defined for each valley candidate $\mathbf{V}_k^c$, which is scanned bottom-up (points with larger y-coordinates first) to find the true inter-toe valley. Each region $\triangle_k$ is formed as the union of two right angle triangles sharing the same cathetus $\overline{\mathbf{V}_k^c\mathbf{C}}$ with right angle at $\mathbf{V}_k^c$ and with inner angles α_k, β_k at $\mathbf{C}$ as follows:

$$\alpha_k := \begin{cases} \frac{\angle(\mathbf{V}_{k+1}^c;\mathbf{C};\mathbf{V}_k^c)}{3} & \textit{if } 0 \leq k < 3; \\ 2\beta_k & \textit{otherwise.} \end{cases} \tag{4.23}$$

$$\beta_k := \begin{cases} \frac{\angle(\mathbf{V}_{k}^c;\mathbf{C};\mathbf{V}_{k-1}^c)}{3} & \textit{if } 1 \leq k < 4; \\ 2\alpha_k & \textit{otherwise.} \end{cases} \tag{4.24}$$

The function $\angle : \mathbb{R}^2 \times \mathbb{R}^2 \times \mathbb{R}^2 \to \mathbb{R}$ estimates the enclosed angle between three (counter-clockwise) points. The inter-toe valley is defined as the point with largest y-coordinate (and smallest x-coordinate, if not unique), that has a non-zero intensity in the binarised foot image (i.e. this method is sensitive to holes within the binarised image referring to inter-toe space).

5 Feature extraction

The designed hand and foot-biometric system is *multimodal* in the sense of [32] combining the result of different matching algorithms in order to improve recognition accuracy. More precisely, according to [33], both proposed systems may be classified as *multi-algorithm* systems applying fusion *after matching*. Therefore, r different feature extractors E_i and matchers S_i for $i \in \{1, \ldots, r\}$ are applied independently of all other extractors and matchers representing different *modalities* (*fingerprint, palmprint, hand geometry* for hands and *ballprint, soleprint, foot geometry* for feet) and *algorithms* within these modalities (e.g. *Eigenpalms + Eigenfingers* or ridge-based *Palmprint* features). Sometimes, single features are applied to individual fingers instead of the entire hand. Then, even a single feature extractor E_i or matcher M_i may use biometric fusion. In this case, given a biometric sample X, feature vectors $\mathfrak{x}_i^u$ of single units u (e.g. fingers or the palm region) are concatenated (or *augmented*, see [33]) to a vector $\mathfrak{x}_i$ representing the union of features for each (virtual) extractor E_i. Unit-based matchers for each algorithm operate on the projected single feature vectors $\mathfrak{x}_i^u$ for each unit, that is, prior to execution of a unit-based matcher, its corresponding projection: $P_i^u : \mathfrak{x}_i \rightarrowtail \mathfrak{x}_i^u$ is applied and a single matching score for each (virtual) matcher M_i is generated. Thus, single feature extractors and matchers may itself (as a single algorithm) be *multimodal*. In this sense, the employed scenario corresponds to *hybrid* biometric systems [33].

Typically, each feature extraction algorithm requires a special resolution (e.g. minutiae extraction requires up to 500 dpi, while hand geometry works on images as low as 45 dpi) provided by the *preprocessing* module directly or by further involving feature extractor dependent preprocessing stages.

According to [42], common hand biometric systems, and therefore foot biometric systems, in an analogous manner can broadly be classified as follows:

- **Geometric features**: schemes relying on silhouette shape and the lengths and widths of fingers, among others;
- **Texture features**: extracting palm curves or minutiae-based texture information (such as the NFIS2 minutiae matching software from NIST [51]);
- **Hybrid approaches**: these systems employ fusion at decision, matching-score or feature extraction level to improve error rates. Examples comprise [18, 19, 34, 28], however, fingerprint matching results are fused with hand geometry in multiple biometric schemes (such as in [32]), rather than combined employing a single sensor (as in [19], for example). A reason for this might be that, usually, fingerprint matching

requires special hardware for image acquisition and does not work on low-resolution input.

In this work, *hybrid* approaches are observed extending the considerations to fingerprint biometrics since minutiae and singular points, which are typical features of fingerprint recognition, may also be extracted from palmprints [44]. Typical ridge structures are also present on large parts of the foot. When multiple features can successfully be extracted from the captured scans of hands or feet, system performance can be increased using fusion without the cost of additional sensors or further inconvenience caused by multiple-step data acquisition.

While some features may be extracted from both hand and foot images, others require different regions or even modifications regarding composition of the feature vector. One example is minutiae-based feature extraction: while fingers exhibit a large touching surface with the scanning device, problems have occurred when trying to extract minutiae out of toes. In such cases, the employed extractors have been applied to different areas within hands and feet (i.e. hand-based minutiae extraction concentrates on finger tips, while for footprints the ballprint area is examined). This, however, introduces a change, which makes a direct comparison of the same algorithm between the different classes of *hand-* and *footprint*-based biometric systems difficult (especially, when results of multiple instances are combined, as is the case for fingerprint matching within hands).

5.1 Overview

This chapter describes in detail the generation of individual feature vectors $\mathfrak{x}_i$ for each extractor E_i. Since both hand and footprint-based algorithms use the same features (with certain modifications), all features will be introduced in sequential order with all necessary adaption to foot-biometrics. Furthermore, problems regarding anatomical differences between hand and foot will be pointed out and solutions will be discussed. A list of implemented features for both systems can be found in Table 5.1.

5.1.1 Geometric features

The first part of the features comprises geometrical features corresponding to techniques used in hand geometry systems. Geometric measurements are frequently employed in hand biometric systems due to their simplicity, robustness to injuries, high collectability and acceptability. They usually rely on low-resolved binary images [18, 43] and, as the major advantage over fingerprint or palmprint features, they demand low processing power. Image capture is less complex than for biometrics captured at a distance (simple scanners suffice for acquisition [43]) and they can be acquired in a short time. However, this comes at the cost of relatively weak accuracy, as illustrated in Table 2.2. Typical hand geometry-based systems [18, 43, 13, 35] exhibit *EER*s in verification mode of $3 - 10\%$ among databases of about $400 - 1000$ samples. While not at all suitable (if not integrated

Algorithm	Hand	Foot
Silhouette	5 finger contour dist. to finger centroid, length and enclosed area of single finger's silhouette polygon	contour dist. to foot centroid, length and enclosed area of silhouette polygon
Shape	5×3 local finger widths and positions	15 local foot widths and positions
Fingerlength/ Toelength	5×3 length of proximal, intermediate and distal phalanges, 5×2 (left and right) average finger widths	5 toe lengths and 4 inter-toe angles
Palmprint/ Soleprint	variance of 144 overlapping blocks in edge-detected image (similar to [18])	variance of 288 overlapping blocks in edge-detected image (similar to [18])
Eigenpalms and Eigenfingers/ Eigenfeet	projection of sub-sampled fingers and palm onto feature space spanned by 25 most significant principal components	projection of sub-sampled footprint onto feature space spanned by 20 most significant principal components
Minutiae	using NIST [51] `mindtct` minutiae extractor on 5 finger tips	using NIST [51] `mindtct` minutiae extractor on ballprint region under big toe

Table 5.1: Employed geometric and texture-based features using hands and footprints as input.

with other biometrics) for identification problems (since for large scale identification out of m subjects, the approximation $FAR(m) \simeq m \cdot FAR$ holds [1]), nevertheless these features serve excellently for classification purposes in cascaded *hierarchial multibiometric systems* [33]: these systems process biometrics in a tree-like architecture to determine the identity of an individual by reducing the number of matches employing a coarse (fast) matcher to derive the top n matches, which are further investigated by more accurate (but more costly) matchers. Considering the sole of the foot can be prone to injuries, shape-based features seem well suited for the foot verification task. A large number of possible features fall into the category of *geometrical features*, which can broadly be divided into:

- **Global features**: focusing on measures with respect to the whole hand, e.g. palm width, length or hand area and;
- **Local features**: representing local lengths and intersections, such as finger lengths and widths at various positions or finger contours.

In this work, three different geometrical algorithms have been implemented, namely *Silhouette, Shape* and *Fingerlength/Toelength*, for both hands and footprints. *Silhouette* takes into account the contour shape of single fingers and feet using a variable-size feature vector. While for hands, the number of typical measurements lies in between $20 - 30$ [42], the two implemented algorithms *Shape* and *Fingerlength/Toelength* extract $9 - 25$ static measurements of the foot and finger shape, depending on whether hands or feet are processed. While for the application on hands, similar performance to existing systems can be expected, due to close-fitting toes, a rather high intra-personal variability in general is assumed for feet. This may also be caused by the fact that many hand recognition schemes rely on a robust identification of finger tips and finger valleys. When these characteristic landmarks can not be detected reliably, a normalisation, i.e. correct placement of individual fingers, is hard to achieve. The extraction of these salient points is often facilitated by pegs [35, 13]. However, since toes can not be spread easily, peg-based systems are almost unacceptable for footprint systems. Instead, toes are typically aligned to a person-specific position, thus inter-toe angles can be extracted as features. More advanced schemes like [43] employ normalization in the preprocessing stage, but demand high contrast between background and palm.

5.1.2 Texture-based features

The second part of features corresponds to texture-based features. Both fingerprint and palmprint biometrics extract features from textural patterns at different *scales*. For fingerprint, these are [21]:

- **Global level features**: determining *singular points*;
- **Local level features**: extracting *minutiae details*; and
- **Very-fine level features**: estimating positions of *sweat pores*.

Palmprints also exhibit textural features at these levels, e.g. metacarpal minutiae extraction [34] or singular-point determination [44]. Important palmprint-based features, according to [44], also involve:

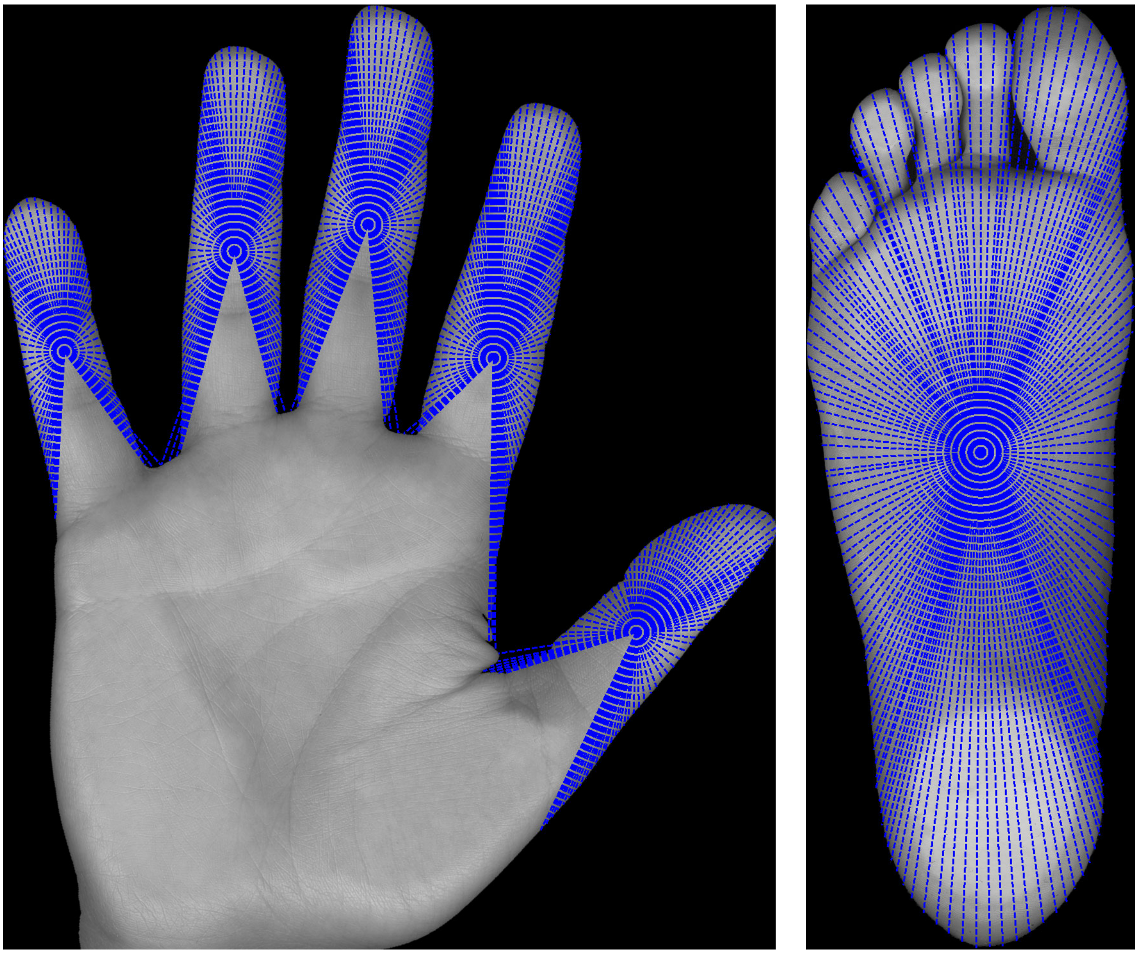
- **2D Gabor-filter-based features**: a set of discrete Gabor filters with discretised orientation, standard deviation and frequency is convoluted with the preprocessed palmprint sub-image yielding a two-bit phase-information;
- **Line features**: first-order derivatives of the grey-level profile are extracted for different directions (e.g. 0, 45, 90 and 135° [18, 44]), from which features are extracted, such as *ChainCode* consisting of directional information for each member of a line;
- **Linear discrimination features**: using algebraic transformations or matrix decomposition, such as the *Fisherpalm* or *Eigenpalm* approach, these extractors consider input images as high-dimensional noisy vectors which need to be projected into a lower-dimensional feature space. A similar situation is assumed for fusion techniques, where dimensionality reduction techniques such as *principal component analysis* (PCA), *independent component analysis* (ICA) and *multidimensional scaling* (MDS) are employed to further reduce augmented feature vectors from multiple possibly correlated features [33].
- **Transform-based features**: By applying Fourier transform to palmprint images and extracting the energy in concentric rings of the frequency image as R-features and within sectors defined by lines through the origin as Θ-features, intensity and direction of principal lines are estimated and encoded.

Using feet instead of hands, new problems arise, such as the difficulty of distinguishing between creases caused by different rotations of feet and permanent line patterns. Typical principal lines, such as *life line, head line* and *heart line*, can not be identified in footprints. Instead a comb-like pattern is present which seems to be sensitive to different pressure distributions. For this reason, simple and generic but robust methods [18] have been favoured in the feature selection process to extract texture-based patterns. In this work, three texture-based features are implemented: *Minutiae* extracts Level-2 fingerprint features on tips of fingers and on the Ballprint area under the big toe using NIST's fingerprint software [51], *Palmprint* or *Soleprint* correspond to line feature-based techniques estimating local principal lines using directional Prewitt edge detection in a generic manner [18] and, finally, *Eigenpalms + Eigenfingers* and *Eigenfeet* is a linear discrimination-based feature using principal component analysis [37]. While in the case of footprints, most algorithms are applied to just one part of the foot, for hands some algorithms may be applied to each finger. In this case, extracted feature vectors are augmented and multimodal fusion techniques are applied at matching stage to obtain a single matching score for each of the algorithms in Table 5.1.

5.2 Silhouette feature

While many hand geometric features record high-level descriptive information about the human hand, e.g. finger lengths or widths, the *Silhouette* feature targets low-level geometric, shape-based information with few assumptions about the measured object. Similar to the shape-based approach in Jain et Duta [9], the objective is to extract a feature using contour information and aligning both hand shapes. As mentioned in the previous chapter, preprocessing already provides feature extraction with a (sub-sampled) contour and various landmarks. However, in contrast to [9], this contour is not pre-aligned for fingers, since it is peg-free. In addition, the alignment procedure is not obtained by finding a transformation (linear or affine, for example) minimising a distance function between the point sets of the contours to be matched, but contour-features are extracted *prior* to alignment and optimally matched using *dynamic time warping* [25]. This approach is more effective in terms of storage requirements (no contour needs to be stored, but the extracted features) and matching performance. Another approach by Yoruk et al. [43] uses principal component analysis to project contour points in 2D onto the eigenspace spanned by the 40, 100, 200 or 400 most significant eigenvectors of covariance matrix $\mathfrak{C}$ obtained by a set of sample hand contours. However, this method requires a proper alignment of single fingers (which is achieved by a normalisation procedure interpreting input data [43]). To be able to apply the same feature to both hands and feet, a simpler approach is favoured, since alignment of toes in a preferred direction is more difficult and prone to normalisation errors. The proposed method is generic in a sense that both hands and feet may be used for feature extraction.

Dynamic time warping-based matching is a natural way to align features when having a time series with possible missing parts in the contour polygon. Since the silhouette



(a) Hand. (b) Foot.

Figure 5.1: Silhouette feature (contour distance to reference point, length and enclosed area of the contour).

$\langle \mathbf{S}_0, \ldots, \mathbf{S}_n \rangle$ typically contains a large number of points ($\approx$ 6000 for hand-prints and $\approx$ 3000 for footprints at the selected resolution respectively), a subsampling of contour points is necessary to reduce target feature vector size. Sub-sampling may be performed by just selecting each $k-$th contour point, thus reducing the number of contour points by a given factor k, or a selection of the next contour point $\mathbf{C}_{i+1}$ based on $\mathbf{C}_i$ by defining a closed disk $\mathcal{D}$ around $\mathbf{C}_i$ with given radius r and selecting the point of $\langle \mathbf{S}_0, \ldots, \mathbf{S}_n \rangle$ with largest index that lies within $\mathcal{D}$, until the starting point intersects with $\mathcal{D}$. In this work the simpler first method with a factor $k = 10$ has been selected for hand-prints. For footprints, $k = 15$ is chosen and an adaption of the first algorithm is applied, which may select additional outlying contour points between $\mathbf{S}_{ik}$ and $\mathbf{S}_{(i+1)k}$ with respect to their distance to the line through these points. This has been done to capture inter-toe valley candidates more accurately. Due to missing finger alignment in case of hands, each finger is extracted prior to sub-sampling and silhouette extraction is performed on each finger separately. At matching stage, a fusion of all matching scores is applied to obtain a single matching score for a hand-print silhouette.

Given a (sub-sampled) contour $\langle \mathbf{C}_0, \ldots, \mathbf{C}_m \rangle$, features are obtained with respect to a reference point $\mathbf{A}$. In the case of footprints, $\mathbf{A}$ corresponds to the center of mass. For separate fingers in hand-prints, the center of mass with respect to the sub-contour of single fingers (prior to sub-sampling) is selected. Note, that the center of mass with respect to a given hand-print is not stable due to different spreadings of fingers. Let $L(\langle \mathbf{C}_i \rangle), A(\langle \mathbf{C}_i \rangle)$ denote length and enclosed area respectively of the selected contour, then the silhouette feature vector $\mathfrak{x}_1$ with respect to the sub-contour $\langle \mathbf{C}_i \rangle$ is constructed as follows:

$$\mathfrak{x}_1 := \begin{pmatrix} s_0 \\ \cdots \\ s_m \\ L(\langle \mathbf{C}_i \rangle) \\ A(\langle \mathbf{C}_i \rangle) \end{pmatrix}, \textit{with } s_k = \parallel \mathbf{C}_k - \mathbf{A} \parallel \forall k \in \{0, \ldots, m\}. \tag{5.1}$$

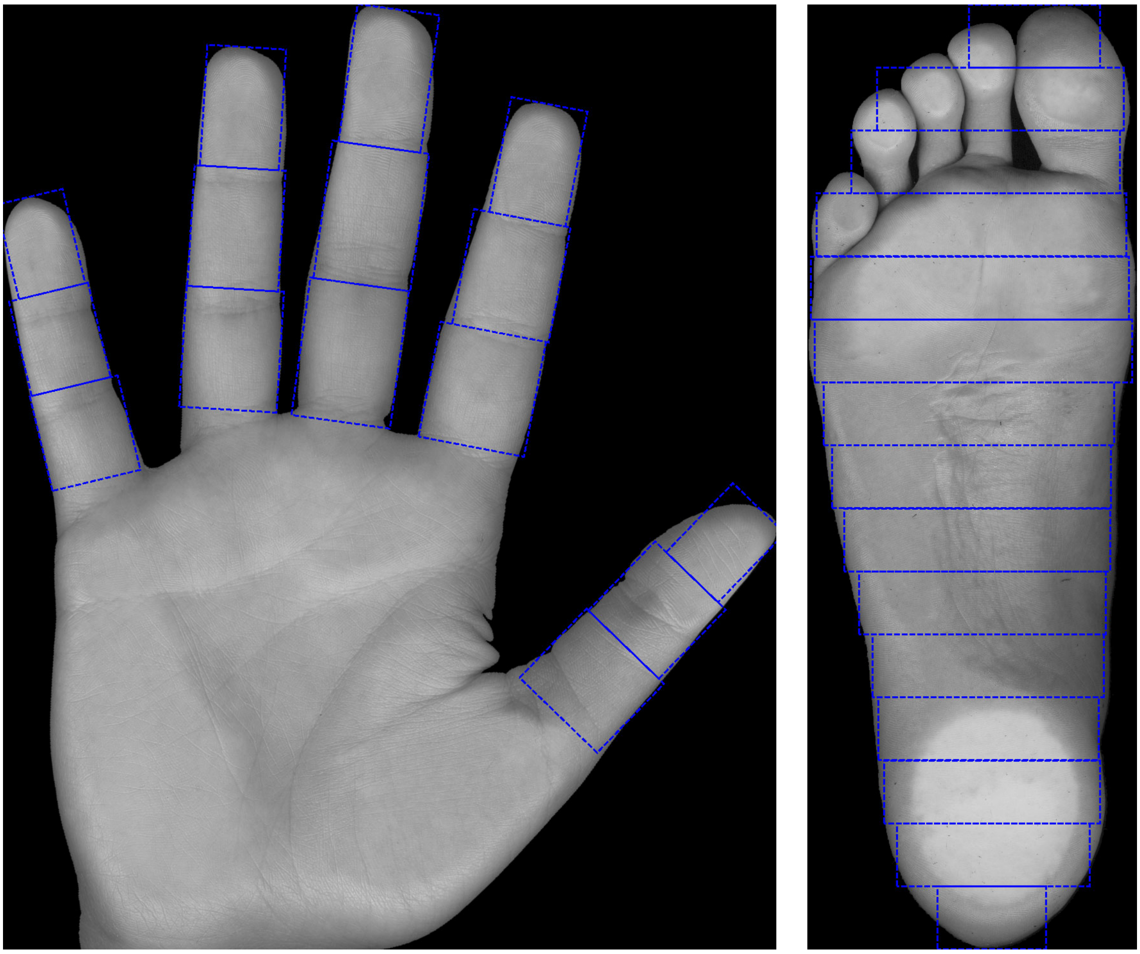
That is, the sequence of contour distances to the reference point is computed and area and length of the contour are inserted. In the case of hand-prints, feature vectors of single finger contours are *augmented*. Figure 5.1 illustrates the silhouette feature for both hands and feet.

5.3 Shape feature

In addition to the silhouette feature introduced before, another shape-based feature is examined by this algorithm. The definition of the following feature is motivated by the fact, that feet are generally characterised by their local widths and bending. Again, an application to both hands and feet is possible after translating feature extraction to single fingers in case of hand-prints. This is necessary in order to avoid high intra-class variability caused by different spreadings of fingers.

After aspect ratio preserving normalisation of the footprint or finger in order to achieve predefined rotation and size, the object is divided into a set of vertical slices $\mathcal{V}_0, \ldots \mathcal{V}_{a-1}$ with equal dimensions. In order to cope with different rotations of single fingers and resulting contraction or elongation of outer finger boundaries, each finger is cropped at the adjacent finger valley with closest distance to the top after normalisation. A left-right scan method estimates the y-monotone contour polygon $\langle \mathbf{S}'_0, \ldots, \mathbf{S}'_b \rangle$ and the set of enclosed pixels $\mathcal{S}$. For each slice, the object's average width w_i is calculated, which corresponds to the average width of the set $\mathcal{V}_i \cap \mathcal{S}$ for $i \in \{0, \ldots, a-1\}$ of in-object pixels. Using a binary image B of size $m \times n$ and let c_i denote the characteristic function of $V_i \cap \mathcal{S}$, then:

$$w_i = \frac{a}{n} \sum_{j=1}^{m} \sum_{k=1}^{n} c_i(j, k). \tag{5.2}$$



(a) Hand. (b) Foot.

Figure 5.2: Shape feature (average widths of vertical slices).

The final feature vector is now constructed as:

$$\mathfrak{x}_2 := \begin{pmatrix} w_0 \\ \dots \\ w_{a-1} \end{pmatrix}. \tag{5.3}$$

For hand-prints, $a = 3$ is selected and single feature vectors are augmented once again, yielding a total feature vector size of 15 components. Footprints are divided into a total of $a = 15$ slices, see Figure 5.2. But in contrast to fingers, feature vectors of footprints exhibit noise in the first two slices caused by toes. For this reason, the first two slices are ignored in the matching process. Another difference to fingers is the problem of hypostatic congestion mentioned in [26]: feet are generally about 5 millimeters larger in the evening than in the morning. In addition, a significant change in weight may cause high inter-personal variability. These problems are not yet covered in experiments and are subject to further investigations.

5.4 Fingerlength and Toelength feature

Hand extremities, i.e. finger tips and finger valleys, are exploited for various different measures using a binary representation of the input image. Jain et al. [13] extract features along 16 axes through fingers and palm. Sanchez-Reillo et. al. [35] use 25 features including $4-5$ finger widths per finger and deviation measurement using a side view of the hand. Kumar et. al. [18] use 12 hand extremity features including 4 finger lengths, 8 finger widths (2 widths per finger). Yoruk et al. [43] employ principal component analysis and/or independent component analysis on the hand shape and contour.

Mapping some of these features to toes and feet in foot biometrics is promising. But, due to close-fitting toes in unstrained pose, special preprocessing is necessary, see Chapter 4, and features are subject to errors. Having extracted all valleys $\mathbf{V}_1, \ldots, \mathbf{V}_4$ and peaks $\mathbf{T}_1, \ldots, \mathbf{T}_5$ for the input image, the outer finger valleys $\mathbf{V}_0$ and $\mathbf{V}_5$ are defined as the contour point approximating the intersection between contour $\langle \mathbf{S}_0, \ldots, \mathbf{S}_n \rangle$ and a circle around the corresponding finger tip with radius equal to the distance to the adjacent finger valley:

$$\mathbf{V}_0 = \mathbf{S}_i \; \textit{with} \; i = \min\{x \in \{0, \ldots, n\} : \| \mathbf{S}_x - \mathbf{T}_1 \| \leq \| \mathbf{V}_1 - \mathbf{T}_1 \| \}; \tag{5.4}$$

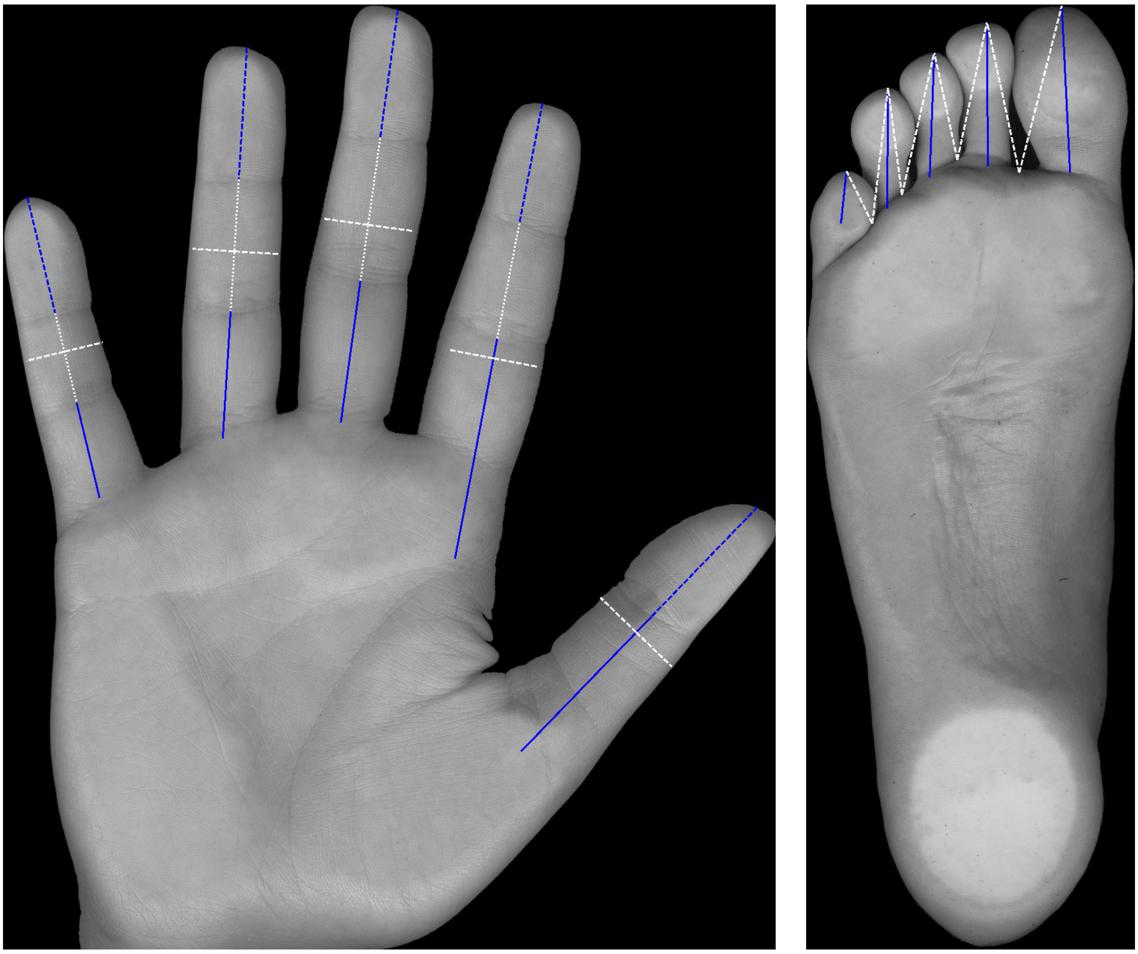
$$\mathbf{V}_5 = \mathbf{S}_i \; \textit{with} \; i = \max\{x \in \{0, \ldots, n\} : \| \mathbf{S}_x - \mathbf{T}_5 \| \leq \| \mathbf{V}_4 - \mathbf{T}_5 \| \}. \tag{5.5}$$

The feature vector for footprints is composed as follows:

$$\mathfrak{x}_3 := \begin{pmatrix} l_1 \\ \cdots \\ l_5 \\ \alpha_1 \\ \cdots \\ \alpha_4 \end{pmatrix} \textit{with} \; l_i = \| \frac{\mathbf{V}_{i-1} + \mathbf{V}_i}{2} - \mathbf{T}_i \|, \; \alpha_j = \angle(\mathbf{T}_j, \mathbf{V}_j, \mathbf{T}_{j+1}). \tag{5.6}$$

That is, the 9 toe extremity values comprising the 5 toe lengths and the 4 inter-toe angles as depicted in Figure 5.3 are extracted. Since inter-finger angles are not stable, a mapping of this feature to hand biometrics is not expected to perform well. Instead, additional finger parts, namely the length of *proximal, intermediate* and *distal phalanx* can be estimated by the contour or textural finger image. In fact, the lengths of bones are not measured, but rather the corresponding finger parts subdivided by creases in the finger texture are estimated. This is achieved by searching the normalised finger within certain windows for the vertical slice of height $h = 30$ with the lowest median of pixel intensities using the local image histogram. After having extracted the length of all three parts as p_j, i_j, d_j for each finger j and the average left and right width of the extracted finger w_j, w'_j, the feature vector for handprints is composed:

$$\mathfrak{x}_3 := (p_1, \ldots, p_5, i_1, \ldots, i_5, d_1, \ldots, d_5, w_1, \ldots, w_5, w'_1, \ldots, w'_5). \tag{5.7}$$



(a) Hand.

(b) Foot.

Figure 5.3: Fingerlength and Toelength feature (lengths and widths of local finger parts in case of hands, toe lengths and inter-toe angles in case of feet).

An interesting advantage having extracted the length of the big toe and its neighbouring one is a pre-classification of feet. Like fingerprints are separated into basic pattern-level classes known as *arch, left loop, right loop, scar, tented arch*, and *whorl* [51], it is also possible to classify feet. According to the differences in length of *hallux* and second toe one can identify the classes *Egyptian feet* (hallux longer than second toe), *Greek feet* (second toe longer than hallux) and *Square feet* (both toes have almost the same length). The relative frequency of each of these three classes is analysed in [8]: the analysed 498 left feet can be segmented into 19.1% Greek, 73.3% Egyptian, and 7.6% Square feet. Similar results are provided for right feet (497 right feet are divided into 18.7% Greek, 73.2% Egyptian, and 8.0% Square). Orthopaedic surgeon Morton [24] was the first to describe this phenomenon of the second toe (also called *Morton's toe*) being longer than the big toe as a part of Morton's syndrome.

5.5 Palmprint and Soleprint feature

While footprints may be classified with respect to the difference in lengths of first and second toe, Zhang [44] proposes a classification of hands into six classes according to the number of principal lines and their intersections. Classification results observed in [44] are: no more than 1 principal line 0.36%, 2 principal lines and 0 intersections 1.23%, 2 principal lines and 1 intersection 2.83%, 3 principal lines and 0 intersections 11.81%, 3 principal lines and 1 intersection 78.12% and 3 principal lines and more than 1 intersection 5.65% among 13800 palmprints. Since footprints do not exhibit typical principal lines, the implemented representative for line-based palmprint matching is a generic approach proposed by Kumar et al. [18], which has proven well for fusion purposes with hand geometry information. Generally, palmprint-based recognition tends to exhibit higher accuracy than geometrical features. In the literature [44], *EER*s of less than 1% are reported, but results largely depend on the employed algorithm and test circumstances. Fusion-based systems [34, 18, 28] operating on similar whole-hand input data like the proposed system in this work exhibit *EER*s in the order of $3 - 6\%$ for their palmprint features. Similar performance is expected for the employed feature in case of hand-prints. For footprints however, due to less distinctive line structures, textile defilement, dorsal injuries and skin creases caused by touching the scanning device, it is not clear if this general statement is true. In cooperative environments, such as access control in thermal baths, intra-class pressure distribution can be expected to exhibit low variance.

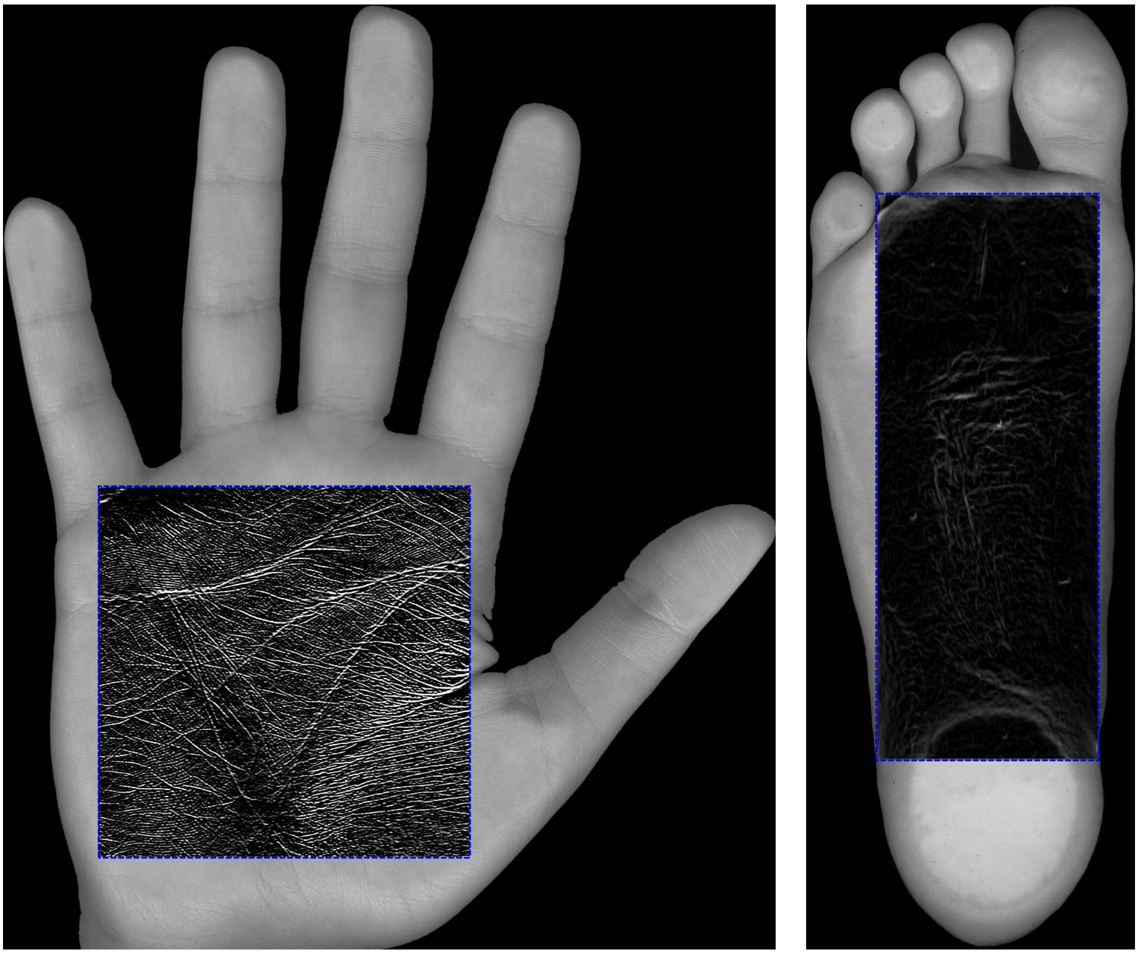
Feature extraction involves the following steps:

1. **Palmprint/Soleprint segmentation**: after rotational alignment, Kumar et al. [18] extract a square fixed-sized palmprint region centered at the center-of-mass $\mathbf{C}$ such that the square is completely inscribed the palm. Another method to determine a textural region used in [34, 44] is to introduce a palmprint coordinate system according to a line through *key points* referring to the inter-finger valleys $\mathbf{V}_2$ and $\mathbf{V}_4$ (X-axis) and a line normal to the segment $\overline{\mathbf{V}_2\mathbf{V}_4}$ through $\frac{\mathbf{V}_2+\mathbf{V}_4}{2}$ (Y-axis) as its origin. Using this coordinate system, a fixed region is selected for extraction. In the proposed approach, the *hand coordinate system* with $\mathbf{V}_3$ as its origin and the line parallel to the palm boundary line through $\mathbf{V}_3$ as its Y-axis is used. Then, a square region of size s equal to the average finger length (without thumb) centered in the Y-axis at offset $0.2 \cdot s$ is extracted, see Figure 5.4(a).

   While the palm can be approximated by a square, the part of the foot which constitutes the sole image used for feature extraction is yet to be determined. Let B denote the binary normalised $m \times n$ footprint and let n be the height of the foot, then the sole of the foot can be defined as the largest inscribed rectangle $\mathcal{R} \subset \{(x,y) : B(x,y) = 1\}$ with given height a such that (see Figure 5.4(b)):

   $$a = \frac{3n}{5} \text{ and } \mathcal{R} \cap \{(x,y) : y < \frac{n}{5} \vee y > \frac{4n}{5}\} = \emptyset. \quad (5.8)$$

2. **Region normalisation and edge detection**: this processing step involves scaling according to a predefined resolution yielding an image R, normalisation in mean and



(a) Hand. (b) Foot.

Figure 5.4: Palmprint and Soleprint feature (variance of overlapping blocks in edge-detected image).

variance and estimation of the first derivative of the image using edge detectors. In the case of hand-prints the extracted region is resized to 300×300 as proposed by [18], in the case of footprints to 300×600 pixels, being twice the size of hand-prints. A normalisation of region R to a predefined mean ϕ_d and variance ρ_d is achieved using the method in [18]. Each pixel intensity $R(x, y)$ is recalculated as follows:

$$R'(x,y) := \begin{cases} \phi_d + \lambda & \textit{if } R(x,y) > \phi, \\ \phi_d - \lambda & \textit{otherwise.} \end{cases} \tag{5.9}$$

where

$$\lambda = \sqrt{\frac{\rho_d(R(x,y) - \phi)^2}{\rho}}. \tag{5.10}$$

The implemented algorithm uses $\phi_d := 100$ and $\rho_d := 400$ for hand-prints, and $\phi_d := 100$ and $\rho_d := 200$ for footprints.

While for footprints line and crease detection is executed using 5×5 Prewitt kernels p_i in different directions ($0°, 45°, 90°$ and $135°$), hand-prints use a 7×7 Prewitt filter before downscaling to 300×300 pixel in order to get a better response for fine-grained principal lines. In case of merging information of different directions, the max-function is used, i.e. [18]:

$$K(x,y) = \max\{R_1(x,y), \dots, R_4(x,y)\}; \tag{5.11}$$

where $R_i(x,y) = p_i * R'(x,y)$, i.e. the normalised image is convoluted with each of the Prewitt kernels.

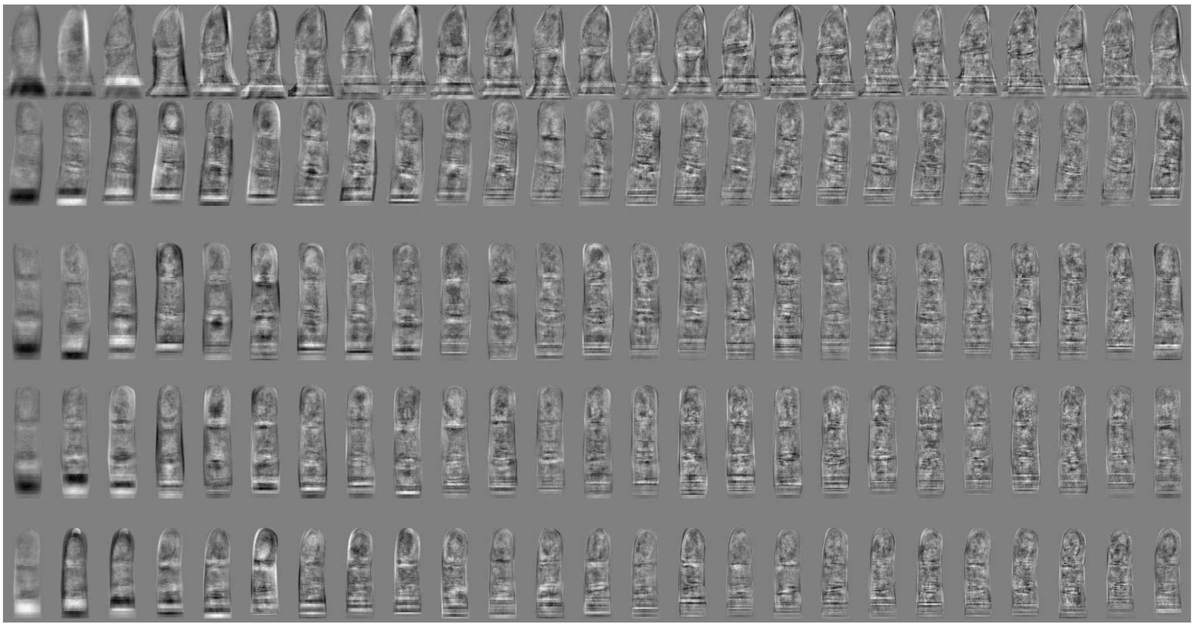
3. **Feature extraction**: the actual feature vector consists of an extraction of variances of b overlapping blocks each of size 24×24 pixel, i.e. $b = 144$ for hands and $b = 288$ for feet.

$$\mathfrak{x}_4 := \begin{pmatrix} \sigma_1^2 \\ \dots \\ \sigma_b^2 \end{pmatrix}. \tag{5.12}$$

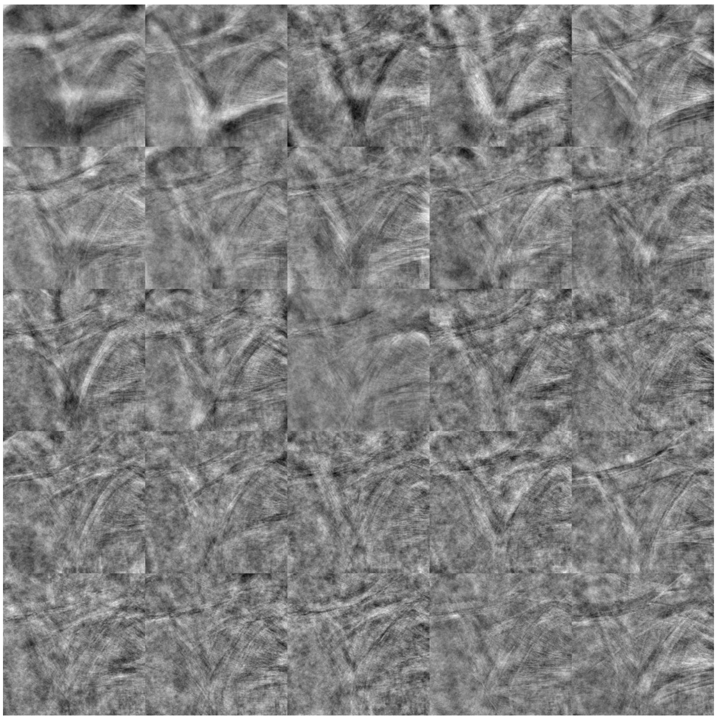
5.6 Eigenpalms, Eigenfingers and Eigenfeet feature

The motivation behind the *Eigenpalms + Eigenfingers* and *Eigenfeet* feature, which are all derived forms of *Eigenfaces* introduced by Turk and Pentland [37], is a method based upon the most relevant features for classification instead of an arbitrary selection of features. In a strict sense Eigenfingers and Eigenfeet features are a both texture-based and shape-based approach since foot and finger silhouette information is also encoded within eigenvectors. It is based on the K-L (Karhunen-Loeve) transform and converts original images of single fingers, palms or feet into a set of characteristic features [44] and thus exploits the fact that all images of a type have similar structures (e.g. all palms have principal lines, which are extracted by the first few principal components [4]). While Eigenpalms + Eigenfingers are reported to exhibit low error rates of 0.58% (see [7]), newly published results by Cheung et al. [4] indicate that some published recognition rates can not be obtained in real applications in the case of (a) larger time lapses between recordings (b) identical twins and (c) unseen objects. Also *virtual twins*, i.e. matching left and (mirrored) right hand of the same person, are reported to lead to a degradation in performance.

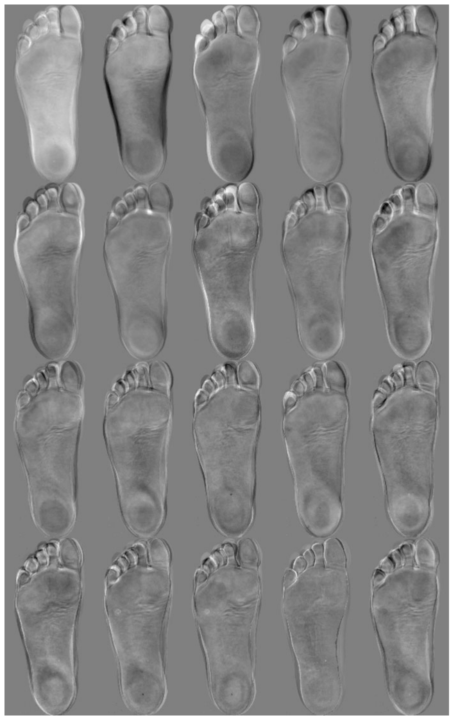
The main idea is to think of an image $\mathfrak{b}$ as a $m \cdot n$ dimensional vector which can be represented exactly in terms of a linear combination of principal components, i.e. eigenvectors (also called *Eigenfaces* for facial images, *Eigenpalms* for palmprints, *Eigenfingers* for finger images and *Eigenfeet* for footprints), computed on the covariance matrix of training images. Eigenvectors are ordered according to eigenvalues and only the ones with the l highest eigenvalues are kept, leaving the most important features that are critical for the recognition task. Since the calculation of eigenvectors is computationally expensive, and in order to reduce the number of input features, typically relatively small resolutions are selected for image input (e.g. Zhang [44] uses 128×128 palm images, but in some literature



(a) Eigenfingers.



(b) Eigenpalms.



(c) Eigenfeet.

Figure 5.5: Comparing eigenspaces calculated from a set of training images (20 hand images and 25 feet): (a) Eigenfingers, (b) Eigenpalms and (c) Eigenfeet.

approaches starting at 64×64 pixels have also been examined, see [4]). In this work, palm-print images are extracted using the hand-print coordinate system as introduced in the previous section and re-scaled to 256×256 pixels. Individual fingers are extracted using contour landmarks and rotationally normalised as introduced for the Shape feature. Then, depending on its type, each finger is aligned on top position, cropped at a specific part of its length and placed on a canvas of fixed size. All fingers except the index are cropped at $\frac{8}{9}$ of their length h, the latter is restricted to $\frac{5}{7}h$. Regarding the input resolution of 500 dpi, canvas size refer to effectively supported areas of $3.25cm \times 9.75cm$ (and $3.25cm \times 6.5cm$ respectively). After bilinear resizing, canvas size is 128×384 for the longer index, middle and ring fingers and 128×256 for the remaining thumb and little finger. In the case of footprints the entire normalised footprint image is proportionally re-scaled and padded to 128×256 pixels. Prior to eigenspace calculation and feature extraction, each image should be normalised in order to compensate sensor variations between different capture sessions [4]. While one alternative to cope with sensor variations is to normalise both mean and variance, the proposed algorithm applies *Contrast-Limited Adaptive Histogram Equalisation* [45] with a window size of 32 to each of the finger and palm images.

Feature extraction using the eigenspace-based algorithm is equal to projecting the input image onto the appropriate *feet, palm* or *finger space* spanned by the most significant eigenvectors. The number of eigenvectors used for this mapping process plays an important role since more eigenvectors are also, up to a certain point, capable of capturing more differences [44]. Typically, the number also depends on the size of the training database. A feature size of 100 eigenvectors is reported to form a good upper boundary [4] and outperforms feature lengths of 50, 150 and 200 in [44]. Investigations in [37] start with small feature sizes of 7 Eigenfaces. In this work, a feature size of 25 Eigenfingers per finger type and also 25 Eigenpalms have been selected from a training database of 25 hand-prints of 25 different users, see Figure 5.5(a) and 5.5(b). Footprints are projected into a space formed by 20 eigenvectors depicted in Figure 5.5(c), also obtained by a set of 20 training images.

A computation of Eigenpalms, Eigenfingers and Eigenfeet which precedes enrolment and matching involves the following two tasks [37]:

1. **Acquisition** of an initial training set of normalised $m \times n$ palm, finger or foot images represented as vectors $\mathfrak{b}_i$ for $i \in \{1, \ldots, x\}$ from which the average image vector $\mathfrak{a}$ is subtracted:

$$\mathfrak{n}_i = \mathfrak{b}_i - \mathfrak{a}, \qquad \mathfrak{a} = \frac{1}{x} \sum_{i=1}^{x} \mathfrak{b}_i; \tag{5.13}$$

2. **Computation** of $mn \times mn$ covariance matrix:

$$\mathfrak{C} = \frac{1}{x} \sum_{i=1}^{x} \mathfrak{n}_i \mathfrak{n}_i^T = \mathfrak{A}\mathfrak{A}^T; \tag{5.14}$$

and eigenvectors $\mathfrak{u}_k$ with according eigenvalues λ_k. For computational efficiency often the $x \times x$ matrix $\mathfrak{A}^T\mathfrak{A}$ is used instead, since the x eigenvectors $\mathfrak{v}_k$ of $\mathfrak{A}^T\mathfrak{A}$

correspond to the x largest eigenvalues $\mathfrak{u}_k$ of $\mathfrak{A}\mathfrak{A}^T$ fulfilling the equation $\mathfrak{u}_k = \mathfrak{A}\mathfrak{v}_k$ and usually x is much smaller than mn.

3. **Ordering and selection** of l highest eigenvectors with corresponding eigenvalues.

For each image type (i.e. thumb, index finger, middle finger, ring finger, little finger, palm and foot), an independent set of eigenvectors $\mathfrak{u}_i$ with $i \in \{1, \ldots, l\}$ and average image $\mathfrak{a}$ is obtained.

Feature extraction refers to a specific image type and comprises the following steps:

1. **Normalisation** of the palm, finger or foot vector $\mathfrak{b}$ calculating $\mathfrak{n} = \mathfrak{b} - \mathfrak{a}$.
2. **Projection** onto the corresponding eigenspace to get the feature vector components $\omega_i = \mathfrak{u}_i^T \mathfrak{n}$. I.e. palms are projected into the *palm space*, single fingers are mapped according to their finger type onto the corresponding *finger space* and feet are projected using the *Eigenfeet* vectors. The feature vector consists of exactly l components:

$$\mathfrak{x}_5 := \begin{pmatrix} \omega_1 \\ \ldots \\ \omega_l \end{pmatrix}; \tag{5.15}$$

such that $\mathfrak{n}$ is approximated by:

$$\mathfrak{n} \sim \sum_{i=1}^{l} \omega_i \mathfrak{u}_i. \tag{5.16}$$

In the case of processed hands, feature vectors for each finger type and palm are augmented.

5.7 Minutiae feature

Typical ridge structure, i.e. the outermost structural part of the *epidermis*, is both present in hand-prints and footprints at high resolutions, even if no special ridge extraction device, such as a fingerprint scanner, is used. The *permanence* of ridge lines being $100 - 300$ micrometers in width in case of injuries or cuts is proven [21] and permits a variety of different features. The feature described in this section refers to local *Galton details*, also called *minutiae*, coarsely divided into *Termination* and *Bifurcation* of ridge lines and represented by coordinate, type and ridge direction at this point [21], see Figure 5.6, with ridges denoted as dark traces on light background. Note that in IAFIS/FBI systems, angles are defined by the rays pointing into positive direction of the x-axis and in direction of the ridge in case of Termination, and in direction through the middle of the intervening valleys in case of Bifurcation [51]. Several other classes (such as *Trifurcation, Lake, Island, ...*) exist, but are not subject to extraction of the employed NFIS2 [51] extraction software `mindtct`, which is designed to extract minutiae out of 8-bit grey-scale images at 500 dpi.

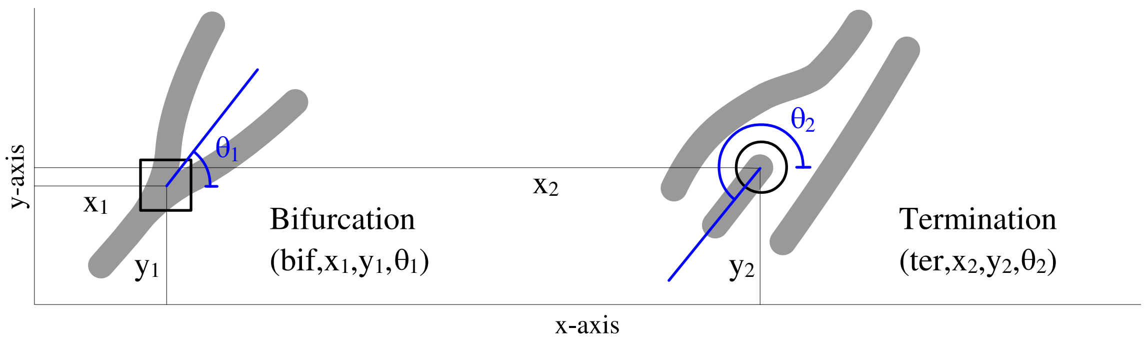
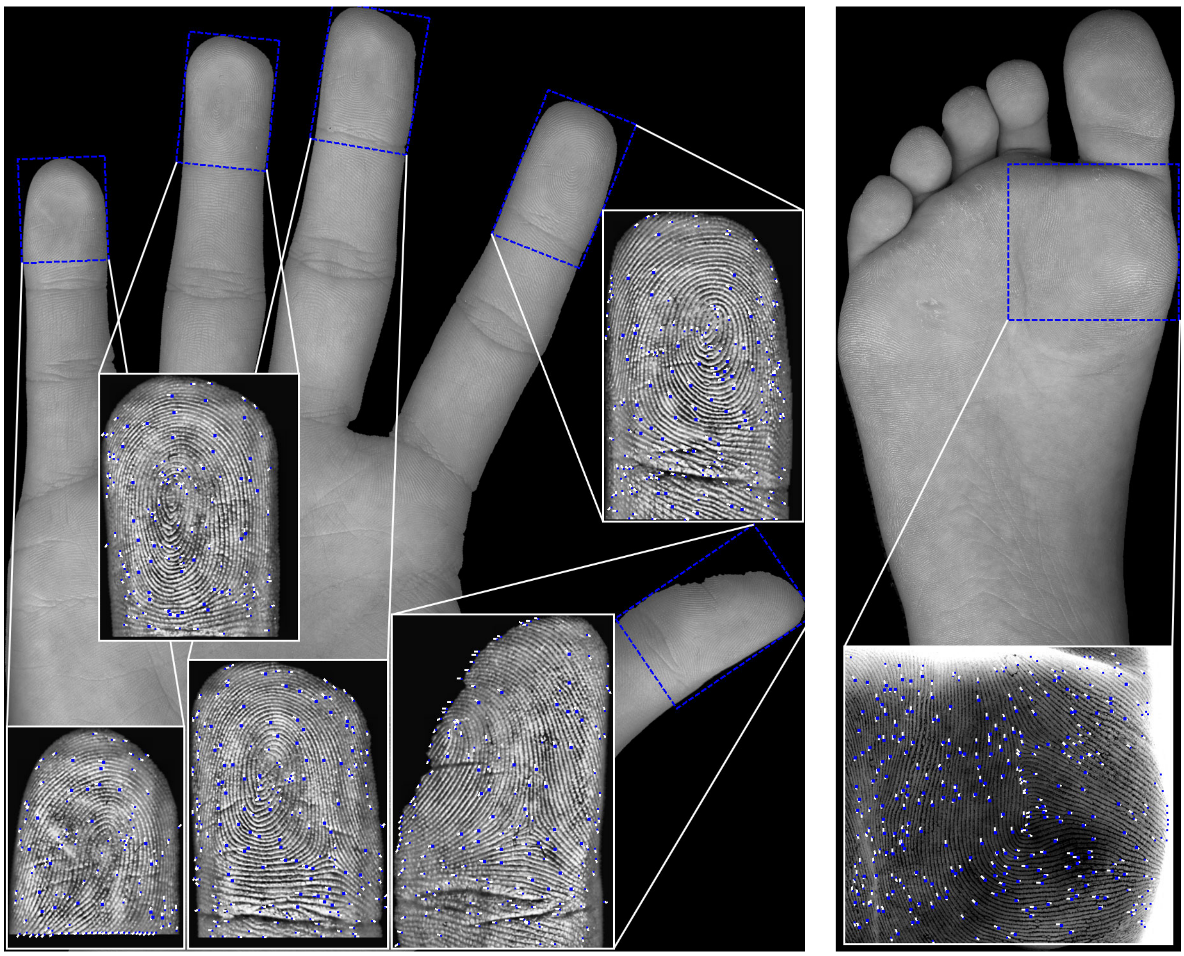


Figure 5.6: Examples of Bifurcation and Termination.



(a) Hand. (b) Foot.

Figure 5.7: Minutiae feature (position, orientation and quality of Bifurcation and Termination points).

Minutiae extraction is a sequential process, which involves binarisation, edge thinning and the generation of specific maps supporting reliable detection. Processing steps for binarisation-based methods are discussed in detail in [21], comparing different approaches and documented in [51] for the employed matcher:

1. **Enhancement**: the goal of image enhancement is to provide better input in case of the quality degradation types (a) ridge discontinuity, (b) not well separated ridges and (c) cuts, creases, and bruises [21]. Image quality can be measured in forms of contrast, orientation consistency of ridges, ridge frequency, etc. and a variety of global and local quality indices have been proposed [21]. In order to improve image quality, two main approaches exist, namely:

    - **General purpose contrast manipulation**: the application of standard image-processing techniques such as *Histogram stretching, Normalisation in mean and variance* or *Wiener filtering*. Especially when sophisticated algorithms for feature extraction are involved, this approach is promising;
    - **Contextual filters**: using a set of filters derived from a mother filter (e.g. Gabor filters with different discrete frequency and orientation), each local image region is convolved with a specific filter depending on the local context, such as ridge orientation or frequency. This method serves local averaging to suppress noise and fill gaps to enhance differentiation between valleys and ridges [21].

    The major problem for flatbed optical fingerprints is low contrast between ridges and valleys. In the case of difficulty in extracting orientation images reliably, a general purpose approach improving local image contrast is favoured. However, traditional histogram equalisation can not perform well in this case. Instead, this work follows the proposal in Wu et al. [41] to employ *Contrast-Limited Histogram Equalisation* described in [45]: local histograms are generated at a rectangular grid of points and the mappings for each pixel are generated by interpolating mappings of the four nearest grid points. In addition to this adaptive histogram equalisation approach, local histograms are clipped at a level, thus restricting the slope of the histogram.

2. **Generation of Image Maps**: among the most frequently used image maps for minutiae extraction is the *local ridge orientation map* or *direction map* computing block-wise the orientation $0 \leq \theta < 180$ of ridges, for example by averaging the gradient $\nabla(x_i, y_i)$ at each point (x_i, y_j) [21] or row-wise summation of pixels in oriented windows and selecting the orientation, which results in the highest response when convolved with a set of waveforms with increased frequency [51]. NFIS2 performs the second technique with 16 directions and 8×8 blocks. Blocks with low flow are recorded in *low flow maps* [51]. Another map for *local ridge frequency* may be acquired block-wise estimating the inverse of average distances between consecutive local maxima within an oriented window y-axis aligned to the ridge orientation [21], but is not used in [51]. NFIS2 also generates maps for *low contrast* for fingerprint segmentation (separation from background), *high curvature* and *quality*.

3. **Minutiae detection**: as the first step, the input image is binarised in order to differentiate between ridges and valleys. Typically, global binarisation is troublesome,

therefore several local methods have been proposed [21] using the generated maps. In [51] pixels are analysed block-wise and assigned binary values according to their intensity with respect to a grid of 9 horizontal rows of size 7, aligned to the ridge flow direction. The pixel is assigned white, if, and only if, the center row's sum of intensities multiplied by 9 exceeds the total grid's intensity [51]. This binarisation procedure is at the heart of minutiae extraction, since minutiae are identified directly in the resulting image (further thinning steps reducing the width of ridges to one pixel may be executed prior to detection). NFIS2 scans the resulting image both horizontally and vertically to identify local pixel patterns, which can be found in [51].

4. **Minutiae filtering**: typically, the number of detected minutiae exceeds the number of permanent minutiae in fingerprint images. Therefore, several structural approaches exist to eliminate or adjust minutiae in *lakes and islands*, *holes*, *blocks with neighbouring low-quality or invalid blocks*, when configured in *hooks* (spikes protruding off the side of a valley), *overlaps* (ridge discontinuities), or on malformed ridge and valley structures (too wide or too narrow) examining local configurations [51]. Additionally, minutiae quality may be assessed. An experienced weakness of the employed NFIS2 matcher is its frequent detection of false minutiae near the fingerprint's contour.

Prior to feature extraction, regions for fingerprint and footprint are defined at fingertips and under the big toe, see Figure 5.7. However, sensing regions may also be defined for other parts of the human foot and hand. For example, Rowe et al. [34] demonstrate the successful application of minutiae features on metacarpal skin texture.

- **Ballprint region**: after rotational alignment of the footprint, a rectangular region of fixed size $\frac{w}{2} \times \frac{h}{6}$ is extracted centered at $\mathbf{B} = (\frac{3w}{4}, \frac{3h}{12})$, where w, h are the width and height, respectively, of a bounding box circumscribing the input footprint.
- **Fingerprint regions**: each fingerprint is extracted as a rectangular area of fixed size $w \times \frac{h}{3}$ (and $w \times \frac{h}{2}$ for the thumb, respectively) aligned with respect to the major axis of the finger circumscribed by its $w \times h$ sized bounding box.

These regions are enhanced using the technique of *Contrast-Limited Histogram Equalisation* (in case of fingerprints) and simple global *Histogram stretching* (for ball-prints). Since a *duality* between the minutiae types *Termination* and *Bifurcation* exist, an inversion of the grey level in the input image only causes a switch in minutiae types [21], but does not further degrade matching, when performed consistently. The output of the `mindtct` extraction software is a minutiae descriptor file containing *position*, *angle* and *quality* information $(x_i, y_i, \theta_i, q_i)$ for each detected minutiae point $\mathfrak{m}_i$ for $i \in \{1, \ldots, c\}$ within the input image in *JPEG Lossless* format:

$$\mathfrak{x}_6 := \begin{pmatrix} \mathfrak{m}_1 \\ \ldots \\ \mathfrak{m}_c \end{pmatrix}. \tag{5.17}$$

In the case of fingerprints, feature vectors are augmented.

6 Matching and Decision

The purpose of the matching module is to compare two given feature vectors and return either a degree of similarity or dissimilarity represented by a *score*. Generally, little may be assumed concerning the scores retrieved from a matcher, except that they are monotonically increasing with higher probability of the corresponding null hypothesis [1], introduced in Chapter 2. Also, the output of matchers need not lie on the same numerical scale and may follow different probability distributions [33]. In this work, normalisation techniques are applied in order to achieve a discrete similarity score $s_i \in \mathbb{N} \cap [0, 100]$ for each matcher S_i with $1 \leq i \leq r$.

In some cases, matching may also incorporate score *fusion* of different independent matchers or *decision* estimating whether the input templates refer to the same person (in verification mode) or a determination of the identity, if it is known to the system (in identification mode), see [21, 33]. The latter two decision tasks are typically executed in a separate decision module. Since multiple biometric features are incorporated in this work, there are several possible *information fusion* mechanisms for matching, according to [32], namely (a) fusion at feature extraction level, (b) fusion at matching score level and (c) fusion at decision level. In this work, techniques from both categories (b) and (c) are applied. It is important to note that beside the fusion of all different matchers yielding a total matching score for the hand-print and footprint system, each individual matcher may itself combine the result of multiple invocations of single matchers. This is the case in fingerprint matching combining minutiae scores of single fingers.

The task of the decision module within a score-level-based multibiometric system in verification mode is to consolidate the vector of matching scores $\mathfrak{s} = (s_1, \ldots, s_r)$ of r different matchers (classifiers) S_i for $i \in \{1, \ldots, r\}$ obtained by matching the biometric sample X with a stored claimed identity I and returning one of the classes $\{genuine, imposter\}$ [33]. More precisely, each matcher uses a specific feature vector $\mathfrak{x}_i$ extracted from X and compares this evidence with a stored feature vector template $\mathfrak{i}_i$ of subject I. Without fusion, the class *genuine* is returned, if $S_i(\mathfrak{x}_\mathfrak{i}, \mathfrak{i}_\mathfrak{i}) \geq \eta$, i.e. the matching score exceeds a threshold, see Chapter 2. With score level fusion, the Bayesian minimum error-rate classification rule applies [33]:

$$assign\ X \ \rightarrow\ genuine\ if P(genuine|\mathfrak{s}_1, \ldots \mathfrak{s}_r) \geq P(imposter|\mathfrak{s}_1, \ldots \mathfrak{s}_r). \tag{6.1}$$

That is, the input pattern X is assigned to the class *genuine*, if, having the evidence of observed scores $\mathfrak{s}_1, \ldots \mathfrak{s}_r$, its posterior probability is larger than for the class *imposter*. Note, that there is already an approximation using scores instead of feature vectors, which is only reasonable in case of very small matching errors [33]. In identification mode, the decision module assigns the input sample to one of $m+1$ classes corresponding to the m enroled identities in the system database $\mathcal{M}$ and a class *reject* representing unseen objects [21]. Rule (6.1) is generalised as the selection of a class ω_j among classes $\{\omega_1, \ldots, \omega_{m+1}\}$ in [33] as follows:

$$assign\ X \ \rightarrow \ \omega_j \ if P(\omega_j|\mathfrak{s}_1, \ldots \mathfrak{s}_r) \geq P(\omega_k|\mathfrak{s}_1, \ldots \mathfrak{s}_r) \ \forall k \in \{1, \ldots, m+1\}. \tag{6.2}$$

Since score-based identification usually involves an invocation of the matching module for each of the enroled members, this work concentrates on multimodal score level fusion in verification mode (i.e. multiple scores are combined into a single consolidated matching score for an individual match). However, the application of hierarchial classifier combination schemes in identification architectures may further reduce computational requirements [21].

When applying fusion at the *decision level*, the fusion module is integrated into system design *after* the decision module operating on single-matcher data. That is, binary class-membership information $\mathfrak{d} \in \{genuine, imposter\}^r$ is consolidated (in case of verification) [32]. In identification mode, the binary decision vector is generalised to a member of the set $\{\omega_1, \ldots, \omega_{m+1}\}^r$. Fusion at this level is regarded to be most simple, and is thus frequently applied. Examples for multimodal systems can be found in [33].

6.1 Overview

Table 6.1 lists all employed matching classifiers (and intra-matcher fusion techniques). The employed matchers may be coarsely divided into:

- **General-purpose matchers**: employing metrics or distances directly on the feature vectors. Also dynamic time warping falls into this category, since it may be applied to arbitrary time-series. Algorithms using matchers of this type are Shape (foot), Fingerlength, Toelength, Palmprint, Soleprint or Eigenfeet. To mention just a few examples in the literature: Hausdorff distance on contours [43] or (weighted) Euclidian distance in eigenspace [44].

- **Context-sensitive matchers**: taking the composition of the feature vector into consideration and interpreting data. This allows pairwise alignment of extracted features prior to distance calculation, e.g. in fingerprint matchers [21]. The Minutiae algorithm applies a context-sensitive matcher.

- **Hybrid matchers**: here, feature vectors are first decomposed into context-dependent components and general-purpose matchers or further context-sensitive matchers are applied to each of the sub-components. Many of the fusion-based matchers fall into this category, such as Silhouette, Shape (Hand) and Eigenpalms + Eigenfingers.

Algorithm	Hand	Foot
Silhouette	dynamic time warp matching of silhouette, normalised distance for length and enclosed area for each finger, fusion of individual fingers using Sum Rule	dynamic time warp matching of silhouette, normalised distance for length and enclosed area
Shape	classifier based on Manhattan distance, fusion of individual fingers using Sum Rule	classifier based on Manhattan distance
Fingerlength/ Toelength	classifier based on weighted Euclidian distance	classifier based on weighted Euclidian distance
Palmprint/ Soleprint	classifier based on Euclidian distance	classifier based on Euclidian distance
Eigenpalms and Eigenfingers/ Eigenfeet	classifier based on Manhattan distance, fusion of individual normalised eigenspace-scores using Product Rule	classifier based on Manhattan distance
Minutiae	classifier based on NIST [51] `bozorth3` matcher, fusion of individual normalised eigenspace-scores using Max Rule	classifier based on NIST [51] `bozorth3` matcher

Table 6.1: Employed matching techniques for hands and footprints.

6.2 Minutiae matching

Typically, when different minutiae sets have to be compared, an alignment between both sets has to be achieved. This may be aggravated by (a) different rotation and translation (b) only partial overlap or (c) non-linear distortion [21]. While the first two restrictions are troublesome to fingerprint sensors, when applying the preprocessing presented in Chapter 4, only non-linear distortion remains a serious problem. Since finger alignment using the major axis of the finger already provides a good pre-alignment, the problem of different rotations within extracted fingerprints is not present when normalised hand images are used. Also the overlapping area almost corresponds to the entire fingerprint, except for the thumb image, which usually depicts a side view depending on the abduction of the *pollex*. This is expected to result in a performance degradation for this finger, which is reported in [34]. Also for footprint images, a pre-alignment can be achieved by using foot contour information. When the matching algorithm is aware of the fact that images are pre-aligned, matching accuracy may be increased. In this work, however, an existing generic minutiae matching algorithm is applied. Of course, small displacements are always present, and, according to Maltoni et al. [21], pressure, skin condition, noise, and feature extraction errors may lead to imperfect pre-alignments, which need to be compensated for by a proper matching software.

Alignment of two minutiae sets $\mathcal{M}_1, \mathcal{M}_2$ containing position and orientation information of minutiae (and possibly also type and quality, as provided by `mindtct` [51]) is executed by estimating a *pairing* of matching minutiae. Maltoni et al. give an overview of the problem in [21]: considering only the first two properties, two minutiae may be *matching* if their spacial and directional difference does not exceed a system-defined threshold t, α, i.e. for $\mathfrak{m}_i = (x_i, y_i, \theta_i) \in \mathcal{M}_1, \mathfrak{n}_j = (x'_j, y'_j, \theta'_j) \in \mathcal{M}_2$ a relation $\leftrightarrow$ can be defined as follows:

$$\mathfrak{m}_i \leftrightarrow \mathfrak{n}_j \Leftrightarrow \| \begin{pmatrix} x_i \\ y_i \end{pmatrix} - \begin{pmatrix} x'_j \\ y'_j \end{pmatrix} \| < t \wedge \min\{|\theta_i - \theta'_j|, 360 - |\theta_i - \theta'_j|\} < \alpha. \tag{6.3}$$

The minutiae-alignment problem corresponds to finding the transformation (with respect to a model, e.g. affine transformations), leading to the maximum number of matching minutiae, when applied to one of the minutiae sets. Note, that since the relation not necessarily reflects a $1:1$ mapping, the *pairing function* determining minutiae correspondence is also unknown and has to be determined (e.g. by exhaustive searching) [21].

Within the employed Minutiae algorithm, matching is executed using the provided matcher within the NFIS2 suite, `bozorth3`, see [51] for details. Score normalisation is executed after each invocation of the matcher and single scores are combined using the *Max Rule* (i.e. the maximum of all observed scores is returned). In contrast to [34], the score of a single finger is not combined with other algorithms, but the consolidated score is subject to fusion. The Max Rule method provides good results even if single fingers exhibit bad quality as long as one fingerprint can be matched well.

6.3 Dynamic time warp matching

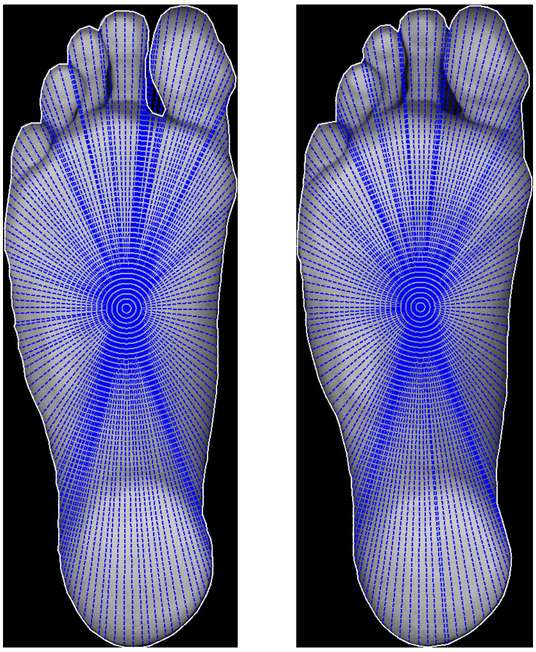
Euclidian or Manhattan metrics demand vectors with equal dimensions. In the case of the Silhouette algorithm, the feature vector of contour distances to the centroid represents a variable-length series needed to be aligned before matching. This sequence alignment in one dimension can be achieved in an *optimal* sense by matching the series non-linearly using the dynamic time warping technique:

Given are two (feature) vectors $\mathfrak{a} \in \mathbb{R}^n, \mathfrak{b} \in \mathbb{R}^m$ of possibly different dimensions n, m, which need to be aligned. Dynamic time warping [25] computes an optimal match between $\mathfrak{a}$ and $\mathfrak{b}$ using a cost function C for the comparison of two components $\mathfrak{a}_i$ and $\mathfrak{b}_j$ at the indices i and j respectively. This implementation uses:

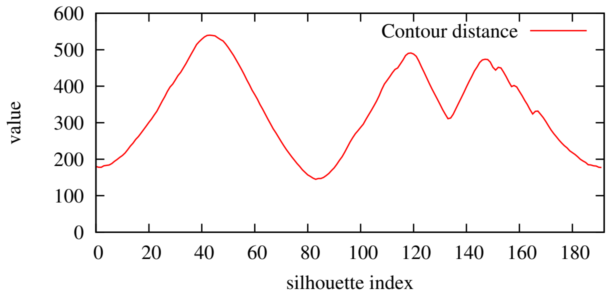
$$C(i,j) := (\mathfrak{a}_i - \mathfrak{b}_j)^2. \tag{6.4}$$

The optimal minimum distance with respect to the cost function is calculated iteratively by evaluating $D(n,m)$:

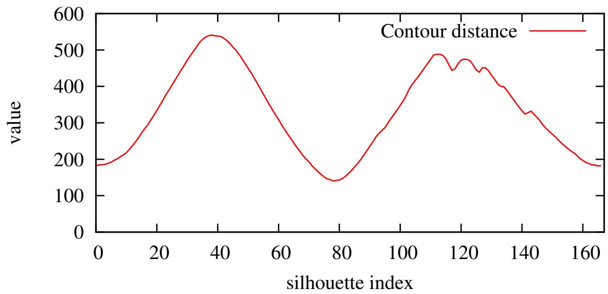
$$D(i,j) := \begin{cases} 0 & \text{if } i = 1, j = 1; \\ C(i,j) + D(i-1,1) & \text{if } i > 1, j = 1; \\ C(i,j) + D(1,j-1) & \text{if } i = 1, j > 1; \\ C(i,j) + \min(D(i-1,j), & \\ D(i,j-1), D(i-1,j-1)), & \text{otherwise.} \end{cases} \tag{6.5}$$



(a) Visual silhouette contours: reference (left) and sample (right).



(b) Reference contour.



(c) Sample contour.

Figure 6.1: Footprint-contours with a missing part in the sample of a genuine user.

The main motivation for using dynamic time warping is better tolerance of missing parts in the silhouette, see Figure 6.1. This is the most likely case in foot biometrics, when inter-toe valleys are clipped by touching toes. Since the Silhouette feature vector also contains components for length and area of the contour, only the first $n-2$ components of each feature vector with length n are matched using this technique. For the remaining components, a simple absolute distance is calculated and normalised. A common score can be achieved by the Product Rule or Sum Rule. Results for individual fingers are combined using the Sum Rule.

6.4 Score level fusion

Score level fusion differs from decision level fusion in the availability of matching scores of individual matchers. But unlike fusion at feature extraction level, no direct access to feature vectors is granted. Instead the posteriori probabilities $P(genuine|\mathfrak{s}_1, \ldots \mathfrak{s}_r)$ and $P(imposter|\mathfrak{s}_1, \ldots \mathfrak{s}_r)$ in Rule (6.1) have to be estimated. According to [33], there are three different techniques for this task:

- **Density-based fusion**: estimating the posteriori probabilities from genuine and imposter distribution densities $P(\mathfrak{s}_i|genuine)$ and $P(\mathfrak{s}_i|imposter)$ of single matchers using parametric and non-parametric methods with the Bayesian inference rule.

Given $k \in \{1, 2\}$ and $\omega_1 = genuine, \omega_2 = imposter$, and let $P(\omega_k)$ denote the prior probability of class ω_k, then [33]:

$$P(\omega_k|\mathfrak{s}_1, \ldots \mathfrak{s}_r) = \frac{P(\mathfrak{s}_1, \ldots \mathfrak{s}_r|\omega_k)P(\omega_k)}{\sum\limits_{l=1}^{2} P(\mathfrak{s}_1, \ldots \mathfrak{s}_r|\omega_l)P(\omega_l)}. \tag{6.6}$$

- **Transformation-based fusion**: Rule (6.1) is an approximation as no feature vectors, rather matching scores, are used for the inference. When no large amounts of training data are available, such as is the case in this work, combination rules are executed on normalised scores, see [32]: normalisation being a mapping of matching scores to a single common domain is necessary in order to be able to compare scores of different matchers. A common normalisation technique is *min-max normalisation*, which is a simple linear mapping of the score interval $[x, y]$ to $[0, 1]$, where x, y refer to the minimum and maximum scores obtained by matching a training set of data [33]. Unless otherwise stated, I have applied this linear mapping technique. Due to the small size of available training data for distance scores parameter x has been set to 0 (matching the template with itself) and parameter y has often been rounded up or manually adjusted (to better fit the score distribution). For more elaborate approaches on score normalisation I refer to [33].
- **Classifier based score fusion**: these refer to learning-based solutions estimating the posteriori probabilities from scores.

Assuming statistical independence of individual matchers (which is most likely the case when different modalities are combined) five prominent classifier combination rules have been identified, namely *Product Rule, Sum Rule, Max Rule, Min Rule* and *Median Rule* [33]. Note, that in case of the applied *transformation based fusion*, in a strict sense, classifier combination rules do *not* have a probabilistic interpretation any more [33]. Nevertheless, the first three methods used for fusion purposes in this work, will be introduced with their statistical background:

- **Product Rule**: being very sensitive to single classifiers returning small scores, and assuming equal prior probabilities $P(genuine), P(imposter)$, this rule can be formulated as follows [33]:

$$assign\ X \ \rightarrow \ genuine\ if \prod_{j=1}^{r} P(genuine|\mathfrak{s}_j) \geq \prod_{j=1}^{r} P(imposter|\mathfrak{s}_j). \tag{6.7}$$

  The Product Rule returned the best result of score-based fusion techniques in [34] for the fusion of directional palmprint bands. In this work, the Product Rule is applied for the fusion of Eigenfinger scores and Eigenpalm scores.

- **Sum Rule**: the basic assumption for this rule is that posteriori probabilities do not differ much from the prior probabilities. Again, assuming equal prior probabilities the rule can be formulated as follows [33]:

$$assign\ X \ \rightarrow \ genuine\ if \sum_{j=1}^{r} P(genuine|\mathfrak{s}_j) \geq \sum_{j=1}^{r} P(imposter|\mathfrak{s}_j). \tag{6.8}$$

This rule is frequently applied and is claimed to outperform the other rules for many applications related with hand biometrics, see [21]. For example, in Kumar et al. [19], it is used for the fusion of fingerprint, palmprint and hand geometry. Sometimes, weights are introduced to emphasise scores provided by more accurate matchers, for example Rowe et al. [34] employ fingerprint and palmprint fusion using the *Weighted Sum Rule* with weights 0.25 for palmprint and 0.85 for fingerprint. It is used for the combination of Shape and Silhouette scores for each finger.

- **Max Rule**: this rule is derived from the Sum Rule by using the maximum summand, i.e. for equal prior probabilities the new rule is [33]:

$$assign\ X \ \rightarrow\ genuine\ if \max_{j=1}^{r} P(genuine|\mathfrak{s}_j) \geq \max_{j=1}^{r} P(imposter|\mathfrak{s}_j). \tag{6.9}$$

  Since this rule is designed to consider the highest posteriori probability of single matchers, it is (in contrast to the Product Rule) tolerant to single bad quality results, e.g. for single bad quality fingerprints. Therefore, scores of single fingerprint matches in this thesis are combined using the Max Rule.

6.5 Decision level fusion

Decision level fusion, also known as fusion at the *abstract level* [33], considers only classification information of single matchers. It is therefore easy to apply, as no assumptions about matchers or distributions are made. Earlier combination is often favoured, since it is assumed to contain more information, and thus be more effective [21]. However, this is not always true, as Kumar et al. [18] point out: their results show, that fusion at decision level outperforms fusion at representation level at least for their multimodal palmprint and hand geometry system.

In verification mode, decision level fusion uses the classes $\mathcal{O} := \{genuine, imposter\}$. But the following rule may also be applied to identification systems returning a matching identity. In this case, $\mathcal{O} = \mathcal{M} \cup \{reject\}$, where $\mathcal{M}$ is the system database of enroled identities.

While also *And rule* and *Or rule*, based on boolean operators on decision vectors, are possible fusion techniques [33], the most popular rule for fusion at decision level is the *Majority Vote Rule*. This rule assumes the equal performance of each of the employed matchers and selects the class with the majority of "votes". Let the number of matchers r be odd, let $\mathfrak{d} \in \mathcal{O}^r$ be the decision vector and let $C : \mathcal{O}^r \times \mathcal{O} \times \{1, \ldots, r\}$ be the supporting function (see [33]):

$$C(\mathfrak{d}, \omega, j) := \begin{cases} 1 & if\ \mathfrak{d}_j = \omega; \\ 0 & otherwise. \end{cases} \tag{6.10}$$

The Majority Vote Rule can be defined as follows (see [33]):

$$assign\ X \ \rightarrow\ \omega\ if \sum_{j=1}^{r} C(\mathfrak{d}, \omega, j) \geq \frac{r+1}{2}. \tag{6.11}$$

In case of an odd number of matchers in verification mode it is clear that such a class ω always exists. In identification mode, when no such class is found, *reject* is returned [33]. The Majority Vote Rule can easily be extended by introducing weights to the supporting function, thus yielding the *Weighted Majority Vote Rule*.

Note, that fusion does not always improve results, as is pointed out in [21]: in the case of combining strong and weak biometrics using the *And Rule* or the *Or Rule*, the overall performance will be reduced. Also fusion of positively correlated classifiers may lead to higher error rates.

7 Experiments

Each evaluation starts with a set of questions to be addressed. Having designed a single-sensor hand and footprint-based multimodal biometric system, a number of tests have been executed in order to investigate the following issues:

- **Question 1**: *"Which recognition accuracy can be achieved when combining multiple modalities on a single-sensor basis for hands?"* This question targets a quantification of accuracy for both systems in terms of the introduced performance measures in Chapter 2. Parts of this issue are already addressed by Rowe et al. [34] and Kumar et al. [18, 19], who have proposed and implemented single-sensor hand-based multibiometric systems. This work extends the considerations to a variety of additional features and assesses the ability to produce high-performance biometric protocols for commercially available flatbed scanners.
- **Question 2**: *"Can different techniques from hand geometry, palmprint and fingerprint biometrics be successfully applied to a foot biometric system?"* Whilst the first question concentrates on hand biometrics, this question aims to provide an overall performance assessment for footprints. Additionally, the performance degradation of individual features in case the same technique is applied to feet instead of hands (with all employed modifications) is addressed.
- **Question 3**: *"Do high-resolution features (e.g. Minutiae) show significantly better performance than low-resolution features (e.g. Eigenfeet, Eigenpalms + Eigenfingers)?"* This question focuses on a separately conducted relative comparison between employed algorithms for hand-based and footprint-based systems to identify the most suitable features of applications. These are characterised by both low error rates and high throughput. Since scanning time depends on resolution, as observed in Chapter 3, the trade-off between accuracy and throughput is the subject of interest for this question.
- **Question 4**: *"Which of these features allows reliable identification?"* Biometric system evaluation is focused on testing systems in verification mode estimating ROC curves and comparing error rates [1]. In applications however, identification mode has many advantages, since it is completely independent of physical possessions or knowledge [14]. Especially for the target domain of footprint-based biometric systems, i.e. wellness areas and spas, this advantage is worth taking into account when considering the overall system's performance in identification mode. Due to the fact that for some of the features (e.g. hand geometry-based features) scalability issues exist [14], only the three best performing algorithms were selected for (positive) identification mode assessment consisting of two experimental setups:

  - **Prepaid architectures**: Here, a database of all customers having paid for a service exists. This corresponds to simple classification into members and non-members according to Hypotheses (2.6) and (2.7). Results are depicted in form of an ROC curve.
  - **Pay-per-entry architectures**: This setup requires the determination of the identity of the person claiming access in order to reliably charge for the service. Since the implemented system may be classified as a ranking system based on scores, its ranking behaviour is analysed estimating *rank probability mass functions*, see [1].

- **Question 5**: *"Which of the fusion rules performs best when combining multiple fingerprints in a single-sensor environment?"* Fingerprints are reported to exhibit high levels of discriminating information [21] and numerous studies about individuality exist, such as [12]. Thus, it is also expected to contribute to a high extent to the overall matching result addressed by *Question 1*. When multiple fingerprints can be acquired simultaneously, it is interesting to see which of the standard fusion rules *Max Rule, Sum Rule* and *Product Rule* performs best. Also individual fingerprint performance is addressed, like in Rowe et al. [34]. In contrast to this approach, which uses just the ring finger's matching score for fusion with different modalities, this work employs the combined score of all fingerprints.

7.1 Test setup

For the evaluation of footprint-based biometric systems, employed test-databases in the literature are rather small. For example, Nakajima et al. [26] use 110 samples of 11 users over the period of 1 month, Jung et al. [15] use 300 samples of 5 users captured at the same time for testing. Thus, results are difficult to compare directly. For hand-based biometrics, the situation is different, since several open databases for single modalities exist (such as [54, 49]), but are not applicable to the employed single-sensor approach. Due to the absence of large-scale publicly available footprint databases and databases providing a whole-hand image at a reasonable resolution, all data used for experiments in this thesis has been collected from volunteers at the University of Salzburg, Austria.

For both footprints and hands, a high-resolution test data set (the test database *DB1* for hands and *DB2* for footprints) with multiple impressions per identity has been recorded. Separate training data sets with a smaller number of users and lower resolution were employed for computation of Eigenfeet, Eigenpalms and Eigenfingers matrices. Samples were captured using the HP Scanjet 3500c flatbed image sensor I have introduced in Chapter 3.

Recording conditions for test databases *DB1* and *DB2* are as follows: Each of the acquired footprint and hand samples per user is recorded with the user sitting in front of the scanning device, which was situated on the floor for feet and on a table for hands. Only the right foot or the right hand is acquired. Footprints are not heavily loaded with full weight and also hands are taken without pressing the object onto the surface of the flatbed scanner. For footprints, before image capturing, the scanning device and sole are cleaned. In case of hands, only the scanning surface is cleaned after each user. All samples were captured in a shaded room for footprints and inside a box for hands in order to minimise the influence of environmental light. The recording interval is a fixed time-span of fifteen minutes per user. Within this time, five scans are taken with respect to a certain acquisition protocol, see Section 7.1.2.

7.1.1 Test databases

The following two databases have been acquired to form the test basis for experimental evaluation.

1. **Hand test database** (*DB1*): This database consists of 443 right-hand samples of 86 people (~ 5 samples per person) captured at the Department of Computer Sciences, University of Salzburg, Austria. With a gender balance of 82.4% male versus 17.6% female samples, templates from both sexes are represented in this data set. Each template exhibits a dimension of 4250×5850 Pixels at 500 dpi and 8-bit grey-scale, which results in a storage requirement of 23.7 MB (uncompressed) per sample.

2. **Footprint test database** (*DB2*): With 160 right-foot samples of 32 people (5 samples per person) captured at the Department of Computer Sciences, University of Salzburg, Austria, this database is similarly gender balanced like *DB1* with 84.4% male and 15.6% female samples. In contrast to *DB1*, image size is regulated manually using the provided preview function. Thus, dimension is sample-dependent between 2566×5952 Pixels (14.5 MB uncompressed, smallest) and 3880×6972 Pixels (25.7 MB uncompressed, largest). All samples were captured at 600 dpi and 8-bit grey-scale.

7.1.2 Acquisition protocol

For reproducibility of recordings, the following acquisition protocol has been employed for *DB1*, *DB2*:

1. **Verification of recording conditions**: in the case of hand captures, users are free to take off any rings or watches, should they desire. For the acquisition of footprints, users are requested to take off shoes and socks and advised to preclean their sole. In either case, users are rejected, if they wear a band-aid, or equivalent.

2. **Instruction**: for the capture of hand images, users are instructed to put their right hand into a box containing a scanner and to touch the surface of the sensor. Users are advised to spread their fingers and choose an arbitrary position (slight rotations allowed) and try not to move during acquisition. In case of footprints, users are instructed to put their foot onto the scanner without pressing down and asked to try not to move during acquisition.

3. **Scanning process**: Footprint scans are manually previewed, hand scans are conducted automatically with default settings.

4. **Evaluation**: After image acquisition, each user is advised to remove his/her hand or foot from the sensor and is shown his scanned hand or foot on a monitor. For *DB1*, images are always stored with increased image counter, even if recording instructions are violated (in contrast to *DB2*, where images may be rejected by the instructor). If a user has violated recording guidelines, recording steps are repeated. The same procedure is executed until at least 5 samples have been acquired per user.

7.2 Hand matching

Within the following section, I conduct both verification and identification experiments using the implemented hand-based biometric system. First, *Question 1* is investigated regarding overall verification performance. Within this context, *Question 5* is also addressed by examining the ROC curve of single-finger minutiae matchers and the fusion strategies *Sum Rule, Max Rule, Product Rule*. Then, identification performance for hands, relating to *Question 4*, is examined.

When combining the performance of various algorithms, a common performance measure has to be selected. Typically, the application determines the appropriate choice of a threshold depending on desired security and convenience [1]. For an application-independent comparison of algorithms, competitions like the FVC2006 [50] have proposed a number of measures to compare different algorithms, like the *EER*, which is most frequently applied in the literature. However, in the case of the employed matchers an exact estimation of the *EER* is difficult, since the employed system thresholds are natural numbers. Thus, the *MinHTER* performance measure is employed for comparison. This indicator can be calculated easily and makes it possible to compare algorithms using a single value. Alternatively, matching performance could also be assessed selecting for each algorithm the operating point with closest distance to the first median, i.e. which is closest to a virtual operating point yielding *EER*.

7.2.1 Verification performance

Inter and intra-class variability with respect to the employed features is assessed with a cross-comparison of available templates, yielding 884 genuine attempts (each hand is matched against the remaining images of the same subject) and 92212 imposter attempts (each hand is also compared against images of all other subjects) on the test set of *DB1*. Due to enrolment errors caused by close fitting fingers and the inability of the algorithm to extract fingers reliably, for example, 11 of the 443 images were rejected, i.e. *FTA* is 2.48%. These samples are no longer considered in matches, thus I explicitly state the use of *FMR*, *FNMR*, which in contrast to *FAR*, *FRR* does not include normalisation errors [22]. Genuine and imposter score distributions (see Chapter 2) obtained for the employed

Algorithm	Hand			Foot		
	MinHTER	*ZeroFMR*	*ZeroFNMR*	*MinHTER*	*ZeroFMR*	*ZeroFNMR*
Silhouette	8.77%	80.88%	53.41%	25.94%	98.12%	100%
Shape	4.71%	70.02%	25.64%	5.72%	51.88%	43.82%
Fingerlength/ Toelength	8.12%	80.54%	100%	23.56%	82.19%	100%
Palmprint/ Soleprint	3.7%	34.73%	100%	19.21%	64.06%	100%
Eigenpalms + Eigenfingers/ Eigenfeet	1.18%	14.93%	10.81%	2.21%	59.38%	8.94%
Minutiae	0.12%	1.13%	16.5%	2.67%	12.81%	97.72%
Fusion of 3 best algorithms	$3 \cdot 10^{-3}$%	0.23%	$6 \cdot 10^{-3}$%	0.41%	7.81%	6.77%

Table 7.1: Verification results for the employed algorithms and Weighted Sum Rule fusion.

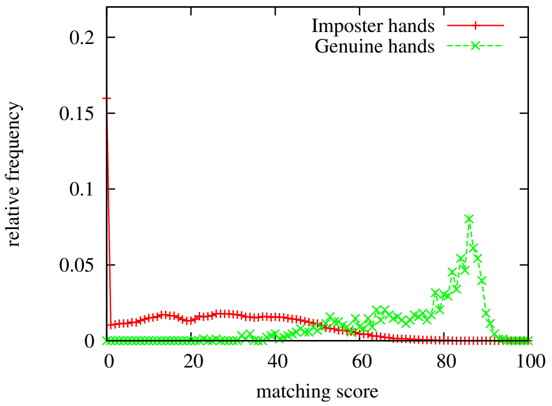
hand-based algorithms are depicted in Figure 7.1. Receiver Operating Characteristics are illustrated in Figure 7.2 for both hand and footprint-based systems in order to visualise performance differences. A collection of all *MinHTER*, *ZeroFMR* and *ZeroFNMR* values for the verification task is given in Table 7.1.

The first feature to be discussed is the Silhouette algorithm, which can be classified as a hand geometry-based feature. Since features derived from hand geometry are known to contain little discriminative information [14], its expected performance is very low. In the literature, *EER* values exceeding 4% [28] and *MinHTER* values exceeding 7% have been reported [18, 19] for Shape-based and geometrical features in multibiometric single-sensor systems. Also for systems exclusively designed for hand geometry, error rates are rather high, see Table 2.2. This expectation is confirmed by the verification experiment. The Silhouette feature exhibits the worst *MinHTER* of 8.77% at threshold $t = 52$.

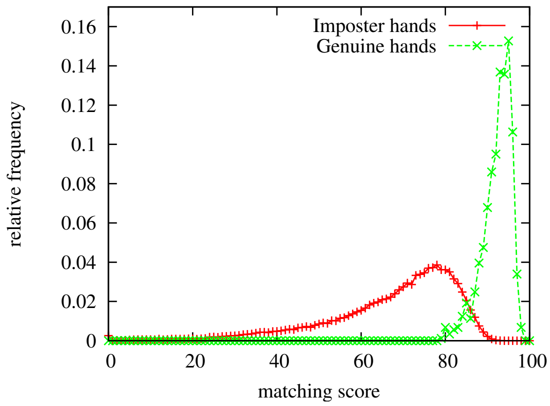
The Shape feature for single fingers performs best of all algorithms in the class of geometrical features, even though it has the smallest feature vector size of only 15 components. From Figure 7.1(b) one can see that the genuine and imposter score distributions for Shape are better separated than for Silhouette and Fingerlength yielding the lowest *ZeroFMR* and *ZeroFNMR* values of all three algorithms. Its *MinHTER* is equal to 4.71% at threshold $t = 87$.

The Fingerlength algorithm performs slightly better than Silhouette in terms of *MinHTER* performance, a rate of 8.12% can be reported for threshold $t = 71$. However, the intervals for genuine and imposter scores are overlapping to a large extent, such that *ZeroFNMR* is even higher than for the Silhouette algorithm. From the manual inspection of matching results, a problem causing low genuine scores is constituted by different abductions of the thumb. These cause differences in finger length of index and thumb. In addition, occasionally, finger parts can not be detected reliably.

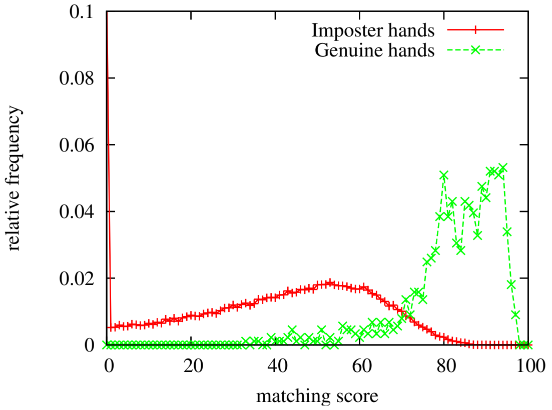
Texture-based features tend to exhibit high accuracy. As the Palmprint feature is extracted in an analogous manner to [18] (except for different preprocessing and normali-



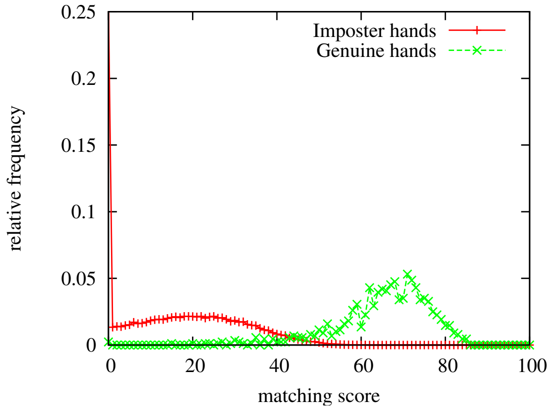
(a) Silhouette feature.



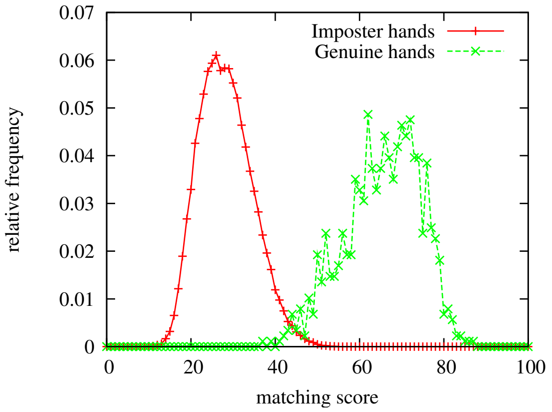
(b) Shape feature.



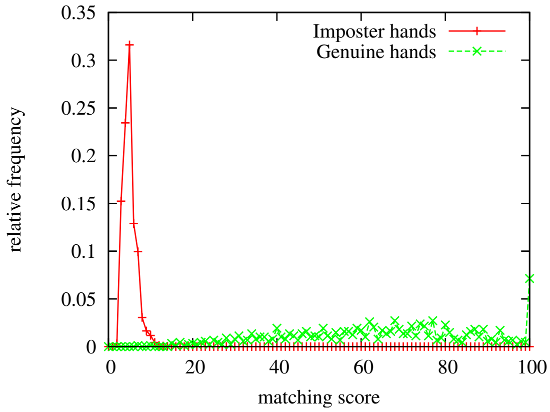
(c) Fingerlength feature.



(d) Palmprint feature.

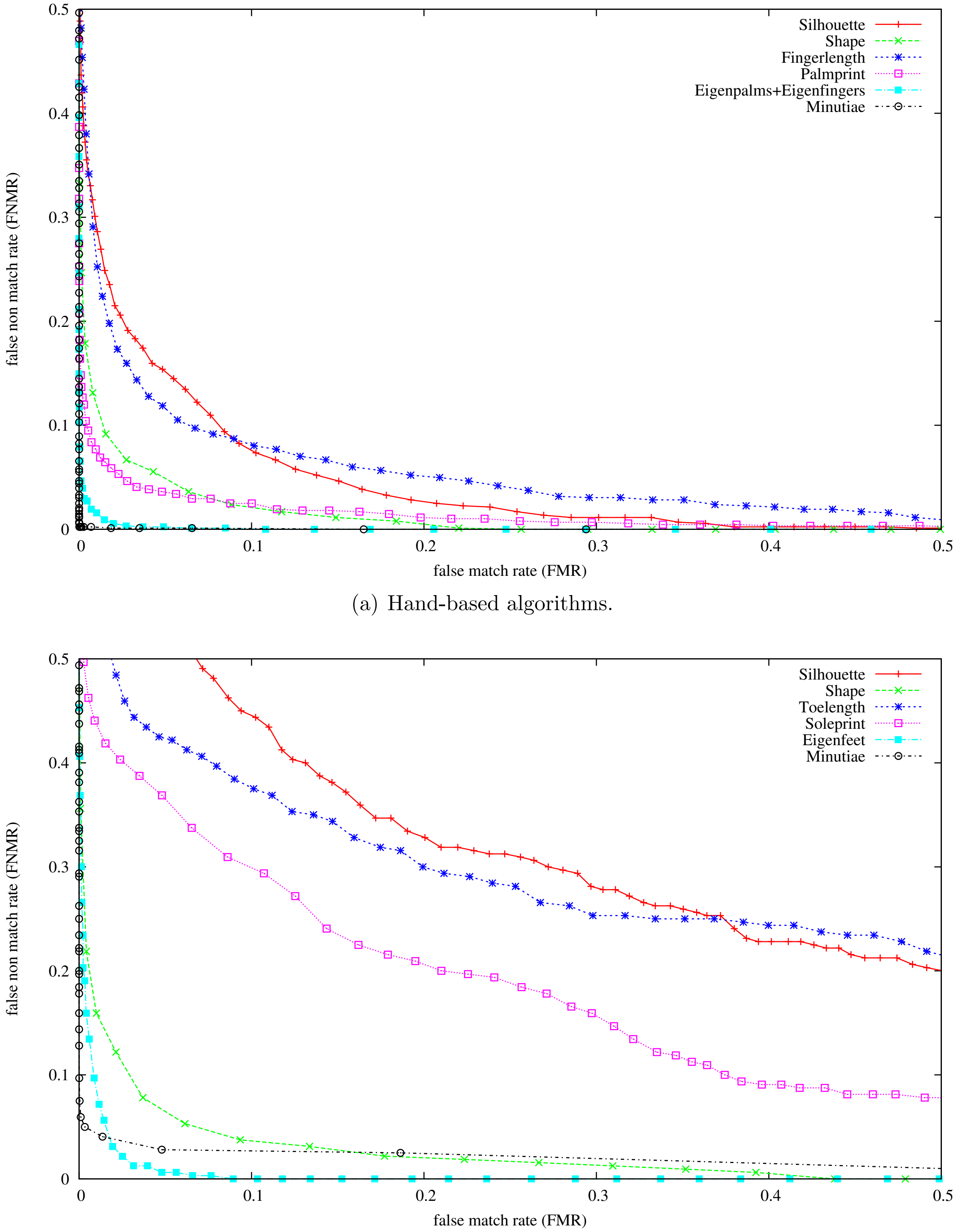


(e) Eigenpalms + Eigenfingers feature.



(f) Minutiae feature.

Figure 7.1: Genuine and imposter score distributions of hand-based features for the verification mode experiment.



(a) Hand-based algorithms.

(b) Footprint-based algorithms.

Figure 7.2: Comparison of Receiver Operating Characteristics in verification mode for (a) hands and (b) footprints.

sation parameters), I expected a performance similar to the reported 3.27% *MinHTER*. Observed results slightly deviate from this rate with a reported *MinHTER* of 3.7% at $t = 43$. This rate is the third best reported accuracy in terms of *MinHTER* for the employed algorithms. However, despite its good performance when inspecting the ROC curve in the area near *EER* of equal security and convenience requirements, it is less suited for applications demanding low *FNMR*.

Fusion-based Eigenpalms + Eigenfingers-based recognition delivers extremely accurate results with a *MinHTER* of 1.18% at $t = 44$. It exhibits the lowest *ZeroFNMR* of all of the employed algorithms and is thus the best choice, when user convenience is most important. Compared to the approach in [44], the observed error rate is higher (see Table 2.2). This might be caused by the relatively small number of features, namely 25 in contrast to 100.

Finally, the fusion-based Minutiae feature provides the best result. The *MinHTER* of 0.12% at threshold $t = 15$ is an order of magnitude smaller than the second best algorithm for the verification task.

If the three best performing single features Palmprint, Eigenpalms + Eigenfingers, and Minutiae are combined, overall system accuracy may be increased with the use of appropriate fusion rules presented in Chapter 6, namely *Weighted Sum Rule* at matching score and *Majority Vote Rule* at the decision level. I have selected these two fusion strategies because they are frequently applied in literature and have already been previously used for the successful fusion of hand-based modalities [18, 19, 34]. The best fusion results are provided by the Weighted Sum Rule using the weights 0.18 for Palmprint, 0.20 for Eigenpalms + Eigenfingers and 0.62 for Minutiae: a *MinHTER* of $3 \cdot 10^{-3}$% at $t = 24$, which represents the operating point of *ZeroFNMR* ($6 \cdot 10^{-3}$%) can be observed. Also the *ZeroFMR* of 0.23% at $t = 25$ is lower than the corresponding rate of each individual feature. The Majority Vote Rule performs worse with an operating point at $FMR = 7 \cdot 10^{-2}$% and $FNMR = 0.45$% corresponding to a half total error rate of 0.26% , which is even higher than for the single Minutiae feature.

Finally, *Question 5* is addressed by examining the ROC curve depicted in Figure 7.3. As can be seen, performance results returned by single fingers using the `bozorth3` matcher are quite different: the index finger exhibits the best *MinHTER* value of 0.98%, followed by middle (1.22%), ring (1.75%), thumb (2.63%) and little finger (7.64%). This is interesting, since Rowe et al. [34] rank the ring finger as the best and the thumb as the worst performing region. A reason for this unexpected behaviour may be caused by the definition of the fingerprint region used for minutiae extraction. The region's size depends on an estimate of the corresponding finger length in this work. In order to identify the best-performing standard fusion rule (*Max Rule, Product Rule, Sum Rule*), all scores of individual matching results have been combined accordingly and evaluated in verification mode. Since returned scores are discretised in our case and rather low for the Product Rule, important discriminating information would be lost during this discretisation process. Therefore, the Product Rule has been modified by the application of the fourth-root function before discretisation. This leads to a better separation of genuine and imposter results in the case of discrete scores. Results are as follows: the *Max Rule* performs best

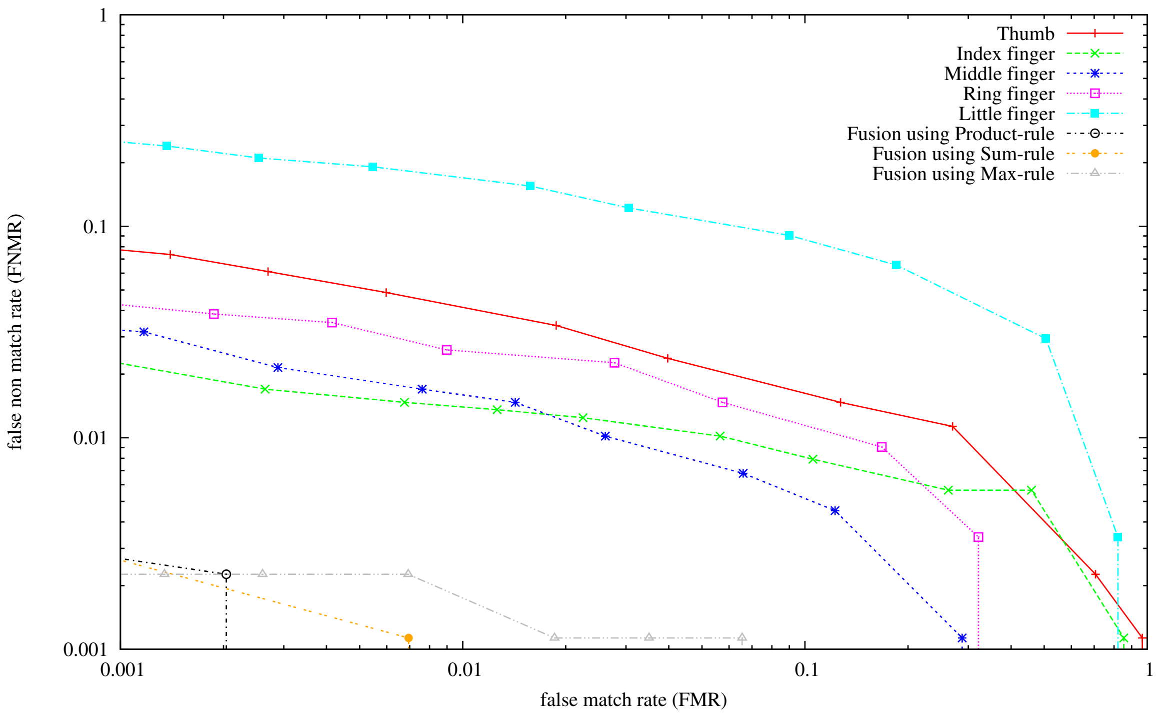


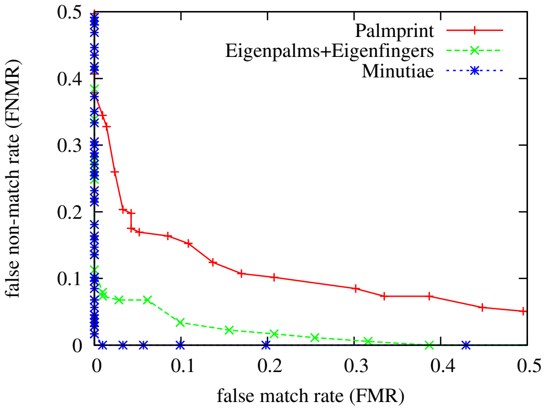
Figure 7.3: Receiver Operating Characteristics in verification mode for minutiae scores obtained when matching individual fingers and using different score level fusion rules.

with 0.12% *MinHTER*, followed by the *Sum Rule* with 0.17% *MinHTER* and the (modified) *Product Rule* with 0.22% *MinHTER*. Nevertheless, each of the employed rules is able to improve recognition accuracy.

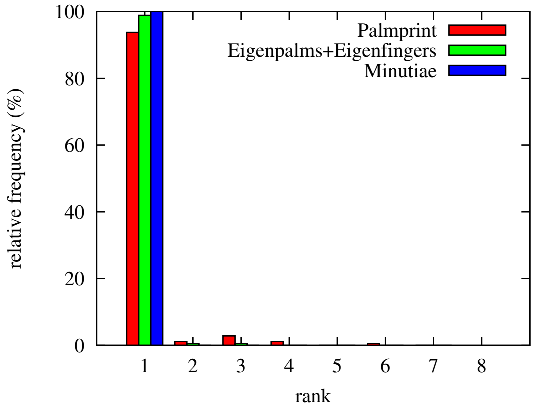
7.2.2 Identification performance

Since algorithms with large error rates in verification mode are hardly scalable to identification applications, only the algorithms Palmprint, Eigenpalms + Eigenfingers and Minutiae have been selected for identification performance evaluation. In order to operate with both seen and unseen objects, the initial test database *DB1* has been divided into a set of enrolment templates and test templates. The first successfully processed image of each of the first 36 male users and 7 female users, i.e. half of the available set of subjects, is selected as enrolment template. This leads to a total number of 177 genuine identification attempts (each attempt yields 43 comparisons with each of the enroled member templates) and 212 imposter identification attempts.

When assessing the prepaid-scenario, the performance ranking of the three employed algorithms does not change. The minutiae-based algorithm still performs best with a *MinHTER* of 0.47%, followed by Eigenpalms + Eigenfingers with 4.14% and Palmprint



(a) Receiver Operating Characteristics.



(b) Rank probability mass fuction.

Figure 7.4: Identification performance using hand-based features by means of (a) Receiver Operating Characteristics for the prepaid scenario (b) rank probability mass functions for the pay-per-entry scenario.

with 10.88%, see Figure 7.4(a) for details. Using a Weighted Sum Rule for matching score fusion with weights 0.01 (Palmprint), 0.07 (Eigenpalms + Eigenfingers) and 0.92 (Minutiae) an ideal classification with no errors can be achieved at threshold $t = 18$. Finally, while Minutiae and Eigenpalms + Eigenfingers are suited for identification, the Palmprint feature does not contribute much to fusion results and as a single feature it is not suited at all for the prepaid-scenario.

For the pay-per-entry scenario evaluation only genuine comparisons are accounted. The recognition accuracy of the full-automatic identification scenario considering only the first position of the ranking vector is observed to be 93.78% for Palmprint, 98.87% for Eigenpalms + Eigenfingers and 100% for the Minutiae feature. Again, minutiae-based biometrics proves to be the most accurate modality for identification. Recognition rates for further ranks, possibly considered for semi-automatic identification are depicted in form of a rank probability mass function in Figure 7.4(b).

7.3 Foot matching

Having executed a verification and identification experiment on hand images, the following section examines the performance of the same algorithms in case of footprints. The main task of this investigation is to find an answer to *Question 2* examining the performance differences between hands and feet as input images. In addition, *Question 5* is addressed measuring the performance in pay-per-entry and prepaid scenarios for access control in public baths using images of the human foot. As the performance measure of choice, the *MinHTER* is again employed for comparison.

7.3.1 Verification performance

The employed database for footprint verification experiments, *DB2* is much smaller than *DB1*, yielding totally 320 genuine and 12400 imposter comparisons. No enrolment errors were reported, thus $FTA = 0\%$. Genuine and imposter score distributions are diagrammed in Figure 7.5.

Silhouette shape is a volatile feature and shows an extremely high *MinHTER* of 25.94% at $t = 64$, which is approximately three times larger than the reported rate for hands. In contrast to the application on hand images, missing contour parts caused by close-fitting toes cause low genuine matching scores, even though dynamic time warping is applied. Together with the Toelength algorithm this geometric feature is not suitable for verification as a single feature.

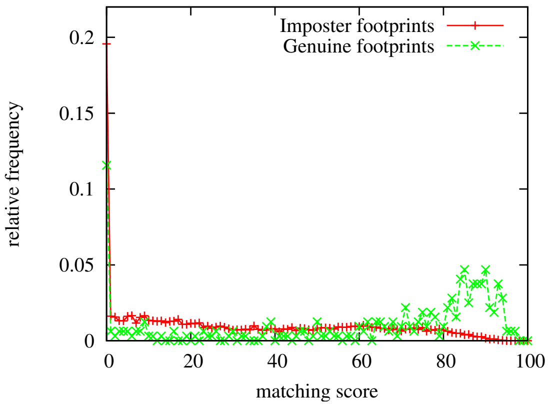
In contrast to all other geometrical algorithms, the Shape feature is least affected by performance degradation when compared to hand biometrics, with a reported *MinHTER* of 5.72% at $t = 90$. It is even outperforming the textural Soleprint approach in case of footprints and also close to the performance of textural features in case of hands.

Length of toes and inter-toe angles for footprint-based verification are less distinctive than length and width of fingers, since inter-toe valleys are more difficult to extract reliably (caused by close-fitting toes at the absence of pegs). The performance of the Fingerlength feature is degraded by a factor close to 3 compared to Toelength, yielding a total result of 23.56% *MinHTER* at $t = 71$ for Toelength. Thus, this feature is not satisfying for the biometric verification task.

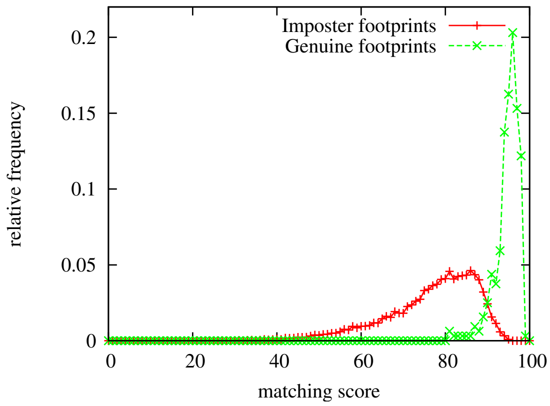
At first glance, for the Soleprint feature similar performance for both hand and footprint images can be expected. But indeed, rates are five times larger than in case of footprints, yielding a *MinHTER* performance of 19.21% at $t = 65$. When inspecting extracted soleprint images, a problem for the reliable extraction of permanent footprint lines can be identified: creases caused by slight rotations of the footprint cause a degradation in performance. Furthermore, the absence of typical expressive lines and textile defilement (due to participants wearing socks) are challenges for texture-based footprint matching.

The best feature for footprint-based personal recognition turned out to be the Eigenfeet feature. Without the need for highly resolved input images, it still produces high accuracy with a performance of 2.21% *MinHTER* at $t = 68$ and yet uses small-sized feature vectors of 160 bytes. It's accuracy is worse than the corresponding result for hand images, however taking into account that in the case of hands, a fusion-based feature of both Eigenpalms and Eigenfingers is used, the performance is quite satisfying. However, time lapses and different recording conditions, such as under-water capture deserve further attention.

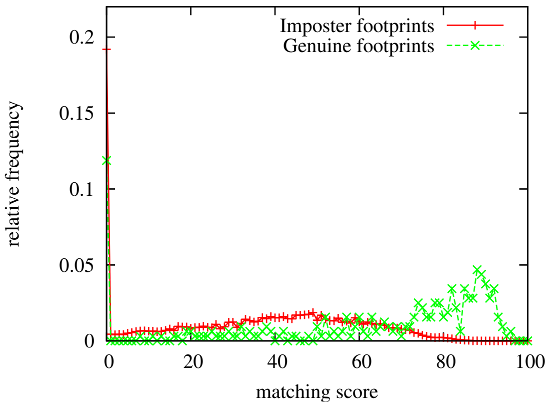
While the Minutiae feature does not outperform the eigenspace-based Eigenfeet feature for footprints in terms of *MinHTER* (its matching accuracy of 2.67% at $t = 7$ is slightly larger than for Eigenfeet), it continues to be the ideal candidate for high-security applications with a *ZeroFMR* of 12.81%. This rate is significantly lower than for all other footprint-based algorithms, as can be seen from Table 7.1. However, the performance degradation,



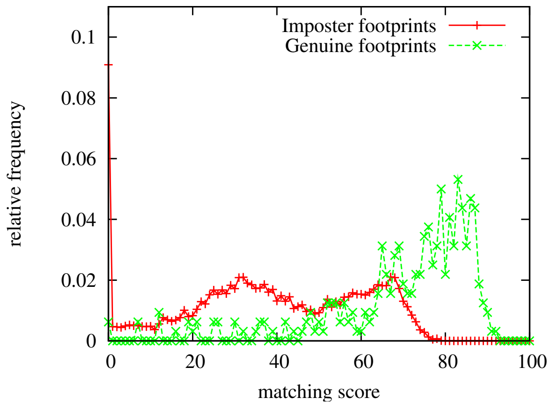
(a) Silhouette feature.



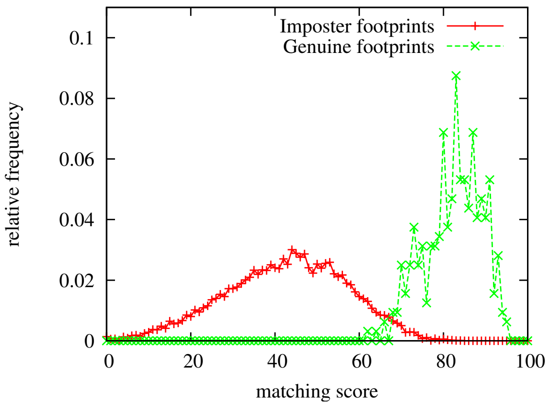
(b) Shape feature.



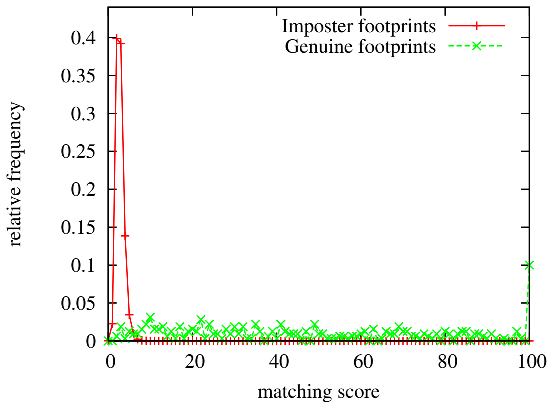
(c) Toelength feature.



(d) Soleprint feature.

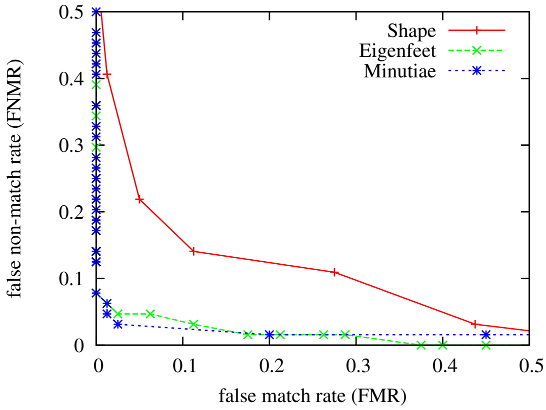


(e) Eigenfeet feature.

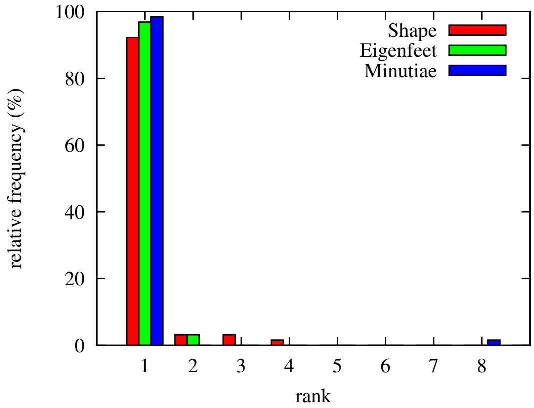


(f) Minutiae feature.

Figure 7.5: Genuine and imposter score distributions of footprint-based features for the verification mode experiment.



(a) Receiver Operating Characteristics.



(b) Rank probability mass function.

Figure 7.6: Identification performance using footprint-based features by means of (a) Receiver Operating Characteristics for the prepaid scenario (b) rank probability mass functions for the pay-per-entry scenario.

when compared to hand biometrics, is even worse than for other algorithms. This might be caused by (a) more accurate preprocessing by means of *Contrast-Limited Adaptive Histogram Equalisation* in case of fingerprints and (b) the missing fusion part, which further increases the accuracy of the minutiae feature. Genuine matching scores returned by `bozorth3` are lower than fingerprint acceptance thresholds recommended by NIST [51]. This might be caused by the higher number of minutiae ($300-400$ per ballprint in contrast to $40-100$ per fingerprint).

Again, I examine the fusion at matching score and at decision level using the three best algorithms Shape, Eigenfeet and Minutiae. In the first case using the Weighted Sum Rule, optimal weights have been identified as 0.05 for Shape, 0.35 for Eigenfeet and 0.60 for Minutiae. This yields a total performance of 0.41% *MinHTER* at $t = 32$, which significantly improves results of single features. In contrast to hand images, also fusion at decision level using the Majority Vote Rule performs better than each of the single algorithms, namely at a *HTER* of 1.32% at the operating point $FMR = 1.08\%$ and $FNMR = 1.56\%$.

7.3.2 Identification performance

Within this last experiment, footprints are matched against a member database obtained from *DB2* by enroling the first 16 users with their first acquired impression of the right foot.

When ROC curves are estimated for Hypotheses (2.6) and (2.7), it is easy to see that the general ranking of the employed features almost remains the same (see Figure 7.6(a)). Now, the Minutiae feature performs best, however with a *MinHTER* of 2.81% it is still an order of magnitude higher than fingerprint identification. Eigenfeet follows at close distance to Minutiae with $MinHTER = 3.59\%$, and finally the Shape feature exhibits the highest error rate of $MinHTER = 12.66\%$. Using the Weighted Sum Rule with weights 0.19 (Eigenfeet), 0.66 (Minutiae), and 0.15 (Shape), a final fusion result of $0.63\% MinHTER$ can be achieved.

Recognition rates for rank 1 in case of pay-per-entry scenario evaluation are 96.87% for Eigenfeet, 98.43% for the Minutiae algorithm and 92.19% for the Shape feature, as can be seen in Figure 7.6(b). Thus, using high-resolved plantar footprint images can further increase existing recognition rates presented in Chapter 2.

7.4 Discussion

The following results have been obtained with respect to the questions addressed by the experimental section:

1. **Result 1**: Concerning the comparison of hand-based features in a single-sensor environment, the best observed overall recognition accuracy is $3 \cdot 10^{-3}\%$ *MinHTER* for the Weighted Sum Rule fusion of Palmprint, Eigenpalms + Eigenfingers and Minutiae. Generally, textural features perform better than geometrical features. Three classes can be identified: a high-performance class (Minutiae, Eigenpalms + Eigenfingers) at approximately 1% *MinHTER*, a mid-range class (Palmprint, Shape) with approximately 5% *MinHTER* and a low-performance class (Geometry, Silhouette) with approximately 9% *MinHTER*.

2. **Result 2**: The best performance for footprints is provided by the Weighted Sum Rule of Shape, Eigenfeet and Minutiae with 0.41% *MinHTER*. When applying hand-based biometric features to footprints, several modifications are necessary, some of which reduce the feature space drastically. The employed experiments report matching performance degenerations by factors larger than 2 for almost every algorithm (except the Shape feature). Reasons for the degradation in performance include: (a) additional fusion of individual finger scores in hand images, e.g. for the Silhouette, Shape, Eigenpalms + Eigenfingers and Minutiae feature, (b) missing typical principal lines in feet for the Soleprint feature, (c) more difficult extraction of intra-toe valleys for the Toelength feature. Results show that matching performance is split into two classes: in the case of the better performing algorithms Shape, Eigenfeet and Minutiae *MinHTER*s of approximately $2 - 6\%$ are achieved, while Silhouette, Soleprint and Toelength show *MinHTER*s of $20 - 30\%$.

3. **Result 3**: Even though the Minutiae algorithm requires high resolution input and therefore causes long scanning durations, its performance justifies the incorporation of this feature into multibiometric hand-based systems. But, when throughput is the main design criterion for applications, Eigenpalms + Eigenfingers provide a reasonable alternative reducing the required resolution by a factor of 5 (input fingers are processed at 100 dpi, palm regions are processed at a resolution depending on the actual size of the palm, but usually less than 100 dpi). This also results in

a potentially 6 times faster acquisition speed (see Chapter 3) with respect to the employed flatbed scanner model HP Scanjet 3500c. While in case of hands, the high-resolution Minutiae feature exhibits error rates, which are an order of magnitude smaller (0.12% instead of 1.18% *MinHTER*) than for Eigenpalms + Eigenfingers, this situation is different for footprints. Here, Eigenfeet and Minutiae perform nearly equally well with $2-3\%$ *MinHTER*. Thus, the Eigenfeet algorithm is the best choice for low resolution input.

4. **Result 4**: Assessing the behaviour of the introduced algorithms in a small-scale prepaid-scenario involving a database size of 16 footprints and 43 hands, the overall performance (using a Weighted Sum Rule) is 0% *MinHTER* for hands and 0.63% *MinHTER* for footprints. While each of the three best algorithms for hands (Palmprint, Eigenpalms + Eigenfingers, Minutiae) and feet (Shape, Eigenfeet, Minutiae) could contribute to the overall fusion result, considered as single features, only the eigenspace-based algorithms Eigenpalms + Eigenfingers, Eigenfeet and the minutiae-based matcher are suitable for identification, with *MinHTER* values less than 10%. The evaluation of pay-per-entry-scenarios estimating rank 1 of the rank probability mass function confirmed this result: results larger than 95% are only provided by the latter two algorithms.

5. **Result 5**: The *Max Rule* performs best for combining individual fingerprint scores in the implemented single-sensor multibiometric system using discrete scores for matching. Exhibiting a *MinHTER* of 0.12% in verification mode, it is closely followed by the Sum Rule (0.17%) and (modified) Product Rule (0.22%). Results for single fingers are in the range of approximately $1-8\%$ *MinHTER*. All employed fusion strategies could improve recognition accuracy.

7.4.1 A note on statistical significance

When error rates for biometric systems are estimated, their statistical significance is an important issue. This refers to the confidence that an examined error rate is not subject to change. Particularly when extremely low error rates are reported for small training sets, such as in the employed work, an analysis of statistical significance is important before assumptions about general behaviour are made. At this point, I want to note that the author of this thesis is aware of the fact that further analysis and more test data are necessary in order to obtain confidence that examined rates continue to remain stable for larger data sets. However, this work is a first attempt to evaluate high-resolution foot biometrics and single-sensor multimodal hand biometrics based on commercially available flatbed scanners. Thus, it examines general behaviour without claiming to provide a specific confidence interval for the observed error rates. Nevertheless, two rules are introduced which are employed in literature for a simple (optimistic) estimation of statistical significance introducing lower bounds.

The expressiveness with respect to statistical significance of the number n of (independent identically distributed) comparisons provided by test set can be estimated by the so-called

Rule of 3 [22]: according to this rule, if no errors within the tests set occur, an error rate of $p = \frac{3}{n}$ can be said with 95% confidence. Applied to the employed set *DB1* of hands in the verification experiment, if no errors occur, an *FNMR* of 0.3% and *FMR* of $3 \cdot 10^{-3}$% is in the 95% confidence interval. For the footprint test set *DB2*, only an *FNMR* of 0.94% and *FMR* of 0.02% is in the 95% confidence interval in case of no reported errors, since it exhibits fewer genuine and imposter matches.

For single reported error rates on the test set, Doddington's *Rule of 30* may be applied [22], demanding the occurrence of at least 30 errors within the test set to be 90% confident that reported error rates do not vary by ±30% of their value.

8 Summary

I examined a single-sensor approach for multimodal hand and footprint-based biometric recognition and evaluated implemented prototypes for both hand and footprint-based biometric systems operating in verification and identification mode. For each of the two systems, six geometric and texture-based algorithms, namely *Silhouette, Shape, Fingerlength/Toelength, Palmprint/Soleprint, Eigenpalms + Eigenfingers/Eigenfeet* and *Minutiae* were compared in terms of resource needs and accuracy. For experimental evaluation, a custom database of 443 right-hand images and 160 right-foot samples was collected. A simple HP Scanjet 3500c flatbed scanner was used to capture a high-resolution $500-600$ dpi 8-bit grey-scale palmar image of the body part.

Regarding image normalisation and alignment, due to the high resolution of input images, an extraction of finger peaks and valleys using state-of-the-art extraction with radial distance functions yielded unsatisfactory results. This problem was addressed by the introduction of salient point refinement using best-fitting-ellipse matching for individual fingers and intersection of their major axis with the contour for peaks and calculation of the intra-finger angle bisector approximating finger boundaries using least-squares for valleys. A novel refinement method to find intra-toe valleys was proposed as well.

Considering the overall performance in verification mode with respect to the employed data sets, footprint-based multimodal recognition exhibited a *MinHTER* of 0.41%, whereas the best fusion algorithm for hands yielded $3 \cdot 10^{-3}$%. Despite the higher error rates of footprints, in case of classifier combination, the performance of unimodal hand-based systems could still be reached. The employed Minutiae feature performed best for hands yielding 0.12% *MinHTER*, followed by Eigenpalms + Eigenfingers (1.18%), Palmprint (3.7%), Shape (4.71%), Fingerlength (8.12%), and Silhouette (8.77%). For footprints a ranking of individual features resulted in: Eigenfeet (2.21%), followed by Minutiae (2.67%), Shape (5.72%), Soleprint (19.21%), Toelength (23.56%), and Silhouette (25.94%).

For the identification experiment, the three best algorithms were further investigated. In the case of a pay-per-entry scenario, only Minutiae-based and Eigenpalms + Eigenfingers/ Eigenfeet-based recognition proved to be suitable for this task, with recognition rates of $97-98$% for footprints (with respect to a database of 16 enroled members) and $99-100$% for hands (with 43 enroled members). Regarding the less restrictive prepaid scenario using the same database, fusion of all three algorithms at matching score using the Weighted Sum Rule yielded no errors in case of hands and 0.63% *MinHTER* in case of feet.

Finally, when compared to unimodal systems, multimodal fusion of hand-based modalities could increase recognition accuracy significantly. Also an image-based approach applied to footprints outperformed existing approaches for comparable database size. Commercially available flatbed scanners were shown to provide good results when used as biometric input sensors. Whilst for traditional access control, hand-based solutions provided more accurate results, the use of foot biometrics in spas or public baths, i.e. areas where footprints may be captured without the need to take off shoes or socks, could be encouraged. For concrete applications targeting this new modality, further topics of interest include a performance evaluation under more realistic conditions, such as (a) underwater capture, (b) larger time lapses between recordings and (c) camera-based instant acquisition to increase transaction times. Regarding the investigations of multimodal fusion of hand-based modalities, where competitive error rates could be observed, future work should especially concentrate on statistical significance analysis using a larger training set size in order to justify low error rates.

Bibliography

[1] R. M. Bolle, J. H. Connell, S. Pankanti, N. K. Ratha, and A. W. Senior. *Guide to Biometrics*. Springer, New York, NY, USA, 2004.

[2] J. F. Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(6):679 – 698, 1986.

[3] R. Cappelli, M. Ferrara, A. Franco, and D. Maltoni. Fingerprint verification competition 2006. *Biometric Technology Today*, 15:7–9, 2007.

[4] K.-H. Cheung, A. Kong, D. Zhang, M. Kamel, and J. You. Does EigenPalm work? A System and Evaluation Perspective. In *Proceedings of the 18th International Conference on Pattern Recognition (ICPR)*, pages 445–448, 2006.

[5] B. Curry. Document Scanner Market Roundup. *Digital Imaging Review*, 28(10):1–9, 2006.

[6] P. Gejgus and M. Sperka. Face tracking in color video sequences. In *Proceedings of the 19th spring conference on Computer graphics*, pages 245–249, 2003.

[7] P. H. Hennings-Yeomans, B. V. K. V. Kumar, and M. Savvides. Palmprint Classification Using Multiple Advanced Correlation Filters and Palm-Specific Segmentation. *IEEE Transactions on Information Forensics and Security*, 2(3):613–622, 2007.

[8] C. Hofgärtner. *Evaluierung der Fußmaße von Industriearbeitern mittels 3D-Scan unter besonderer Berücksichtigung von Fußfehlstellungen*. PhD thesis, University of Tübingen, Tübingen, Germany, 2007.

[9] A. K. Jain and N. Duta. Deformable matching of hand shapes for verification. In *Proceedings of the International Conference on Image Processing (ICIP)*, pages 857–861, 1999.

[10] A. K. Jain, L. Hong, and S. Pankanti. Biometric identification. *Commun. ACM*, 43(2):90–98, 2000.

[11] A. K. Jain, S. Pankanti, S. Prabhakar, L. Hong, and A. Ross. Biometrics: A Grand Challenge. In *Proceedings of the 17th International Conference on Pattern Recognition (ICPR)*, pages 935–942, 2004.

[12] A. K. Jain, S. Prabhakar, and S. Pankanti. Twin Test: On Discriminability of Fingerprints. In *Proceedings of the 3rd International Conference on Audio and Video-Based Biometric Person Authentication (AVBPA)*, volume 2091 of *Lecture Notes in Computer Science*, pages 211–216. Springer, 2001.

[13] A. K. Jain, A. Ross, and S. Pankanti. A prototype hand geometry-based verification system. In *Proceedings of the 2nd International Conference on Audio- and Video-based Biometric Person Authentication (AVBPA)*, pages 166–171, 1999.

[14] A. K. Jain, A. Ross, and S. Prabhakar. An Introduction to Biometric Recognition. *IEEE Transactions on Circuits and Systems for Video Technology*, 14(1):4–20, 2004.

[15] J.-W. Jung, K.-H. Park, and Z. Bien. Unconstrained Person Recognition Method using Static and Dynamic Footprint. In *Proceedings of the 18th Hungarian-Korean Seminar, Budapest, Hungary*, pages 129–137, 2002.

[16] J.-W. Jung, T. Sato, and Z. Bien. Dynamic Footprint-based Person Recognition Method using Hidden Markov Model and Neural Network. *International Journal of Intelligent Systems*, 19(11):1127–1141, 2004.

[17] R. Kennedy. Uniqueness of bare feet and its use as a possible means of identification. *Forensic Science International*, 82(1):81–87, 1996.

[18] A. Kumar, D. C. M. Wong, H. C. Shen, and A. K. Jain. Personal Verification using Palmprint and Hand Geometry Biometric. In *Proceedings of the 4th International Conference on Audio-and Video-Based Biometric Person Authentication (AVBPA)*, pages 668–678, 2003.

[19] A. Kumar and D. Zhang. Combining Fingerprint, Palmprint and Hand-Shape for User Authentication. In *Proceedings of the 18th International Conference on Pattern Recognition (ICPR)*, pages 549–552, 2006.

[20] P.-S. Liao, T.-S. Chen, and P.-C. Chung. A Fast Algorithm for Multilevel Thresholding. *Journal of Information Science and Engineering*, 17(5):713–727, 2001.

[21] D. Maltoni, D. Maio, A. K. Jain, and S. Prabhakar. *Handbook of Fingerprint Recognition*. Springer, New York, NY, USA, 2003.

[22] A. Mansfield and J. Wayman. Best practices in testing and reporting performance of biometric devices. version 2.01. NPL Report CMSC 14/02, 2002.

[23] T. Mansfield. Biometric Product Testing. CESG/BWG Biometric Test Programme, 2001.

[24] D. Morton. Metatarsus atavicus: the identification of a distinct type of foot disorder. *The Journal of Bone and Joint Surgery*, 9:531–544, 1927.

[25] C. S. Myers and L. R. Rabiner. A comparative study of several dynamic time-warping algorithms for connected word recognition. *The Bell System Technical Journal*, 60:1389–1409, 1981.

[26] K. Nakajima, Y. Mizukami, K. Tanaka, and T. Tamura. Footprint-based personal recognition. *IEEE Transactions on Biomedical Engineering*, 47(11):1534–1537, 2000.

[27] K. Nandakumar, Y. Chen, A. K. Jain, and S. C. Dass. Quality-based Score Level Fusion in Multibiometric Systems. In *Proceedings of 18th International Conference on Pattern Recognition (ICPR)*, pages 473–476, 2006.

[28] M. G. K. Ong, T. Connie, A. T. B. Jin, and D. N. C. Ling. A single-sensor hand geometry and palmprint verification system. In *Proceedings of the 2003 ACM SIGMM workshop on Biometrics methods and applications*, pages 100–106, 2003.

[29] N. Otsu. A threshold selection method from gray-level histograms. *IEEE Transactions Systems, Man, and Cybernetics*, 9(1):62–66, 1979.

[30] J. Pierrot, J. Lindberg, J. Koolwaaij, H.-P. Hutter, D. Genoud, M. Blomberg, and F. Bimbot. A comparison of a priori threshold setting procedures for speaker verification in the CAVE project. In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 125–128, 1998.

[31] N. K. Ratha, J. H. Connell, and R. M. Bolle. Enhancing security and privacy in biometrics-based authentication systems. *IBM Systems Journal*, 40(3):614–634, 2001.

[32] A. A. Ross and A. K. Jain. Information fusion in biometrics. *Pattern Recognition Letters*, 24:2115–2125, 2003.

[33] A. A. Ross, K. Nandakumar, and A. K. Jain. *Handbook of Multibiometrics*. Springer, Secaucus, NJ, USA, 2006.

[34] R. K. Rowe, U. Uludag, M. Demirkus, S. Parthasaradhi, and A. K. Jain. A Multispectral Whole-hand Biometric Authentication System. In *Proceedings of Biometric Symposium*, 2007.

[35] R. Sanchez-Reillo, C. Sanchez-Avila, and A. Gonzalez-Marcos. Biometric Identification through Hand Geometry Measurements. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(10):1168–1171, 2000.

[36] K. Sobottka and I. Pitas. Extraction of facial regions and features using color and shape information. In *Proceedings of the 13th International Conference on Pattern Recognition (ICPR)*, pages 421–425, 1996.

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

[38] A. Uhl and P. Wild. Personal identification using Eigenfeet, Ballprint and Foot geometry biometrics. In *Proceedings of the IEEE First International Conference on Biometrics: Theory, Applications, and Systems (BTAS)*, pages 1–6, 2007.

[39] A. Uhl and P. Wild. Footprint-based biometric verification. *Journal of Electronic Imaging*, 2008. to appear.

[40] J. L. Wayman. Technical testing and evaluation of biometric identification devices. In *Biometrics: Personal Identification in a Networked Society*, pages 345–368, Dordrecht, The Netherlands, 1999. Kluwer Academic Publishers.

[41] C. Wu, S. Tulyakov, and V. Govindaraju. Image Quality Measures for Fingerprint Image Enhancement. In *Multimedia Content Representation, Classification and Security*, volume 4105 of *Lecture Notes in Computer Science*, pages 215–222. Springer, 2006.

[42] E. Yörük, H. Dutagaci, and B. Sankur. Hand biometrics. *Image Vision Comput.*, 24(5):483–497, 2006.

[43] E. Yörük, E. Konukoglu, B. Sankur, and J. Darbon. Shape-based hand recognition. *IEEE Transactions on Image Processing*, 15:1803–1815, 2006.

[44] D. Zhang. *Palmprint authentication.* Kluwer Academic Publishers, Dordrecht, The Netherlands, 2004.

[45] K. Zuiderveld. Contrast limited adaptive histogram equalization. In *Graphics gems IV*, pages 474–485, San Diego, CA, USA, 1994. Academic Press Professional, Inc.

[46] AskNumbers Men's Shoe Size conversion. `http://www.asknumbers.com/ShoeSizeMensConversion.aspx`, 2007.

[47] Biometrics Market and Industry Report 2007-2012. International Biometric Group Independent Expertise, 2007.

[48] FBI Integrated Automated Fingerprint Identification System. `http://www.fbi.gov/hq/cjisd/iafis.htm`.

[49] Fingerprint Verification Competition 2004. `http://bias.csr.unibo.it/fvc2004`, 2006.

[50] Fingerprint Verification Competition 2006. `http://bias.csr.unibo.it/fvc2006`, 2006.

[51] NIST Fingerprint Image Software 2. `http://fingerprint.nist.gov/NFIS`, 2004.

[52] NSTC Biometrics Glossary. `http://www.biometrics.gov/docs/glossary.pdf`, 2006.

[53] NSTC Palm Print Recognition. `http://www.biometrics.gov/docs/palmprintrec.pdf`, 2006.

[54] PolyU Palmprint Database. `http://www.comp.polyu.edu.hk/~biometrics/`, 2003.

[55] Scanner Market Reaches Maturity - Penetration Nearing One Third of U.S. PC Households. `http://www.infotrends.com/public/Content/Press/2001/06.19.2001.html`, 2001.