

# Einfluss der Häufigkeit und Verteilung von Minutien auf die Qualität von Algorithmen zur Erkennung von Fingerabdrücken

Bachelor-Thesis von Sarah Ahmed

Darmstadt, Dezember 2011



TECHNISCHE  
UNIVERSITÄT  
DARMSTADT

Fachbereich Informatik  
Fachgebiet Sicherheit in der Informationstechnik





---

## Zusammenfassung

---

Diese Arbeit befasst sich mit dem Thema der Fingerabdruckerkennung im weitesten Sinne. Hierbei geht es insbesondere um eine mögliche Quelle für Falscherkennungen und somit erhöhte Falschakzeptanzrate bei der Fingerabdruckerkennung. Es soll ermittelt werden, ob besondere Häufungen von Minutien einen Einfluss auf Algorithmen zur Erkennung von Fingerabdrücken haben oder nicht.

Hierfür wird in der Arbeit eine spezielle Betrachtungsweise der Minutien gewählt, welche sich auf eine ringförmige Untersuchung aller Minutien um den Mittelpunkt herum stützt.

Für diesen Ansatz wird eine Hilfsanwendung implementiert, welche die Daten mehrerer Fingerabdrücke ermittelt und zum Zwecke der einheitlichen Betrachtungsmöglichkeit auf einen gemeinsamen Mittelpunkt normiert.

Die gewonnenen Daten werden dann unter der These von Minutienhäufungen in Form von Wolken beleuchtet und die daraus gewonnenen Erkenntnisse diskutiert.



---

## Inhaltsverzeichnis

---

1.....Einleitung	1
1.1. Motivation	1
1.2. Problemstellung	2
1.3. Ziel der Arbeit	2
2.....Allgemeine Grundlagen	3
2.1. Biometrische Merkmale und ihre Eigenschaften	3
2.2. Fingerabdruck-Erkennung	4
2.3. Sensor-Technologien zur Fingerabdruck-Erkennung	6
2.4. BioAPI Framework	8
2.5. Middleware – secunet biomiddle	9
2.6. Algorithmen	10
2.6.1. NIST Bozorth3 Funktionsweise	10
2.7. Qualitätsabschätzung und Sicherheit	12
3.....Lösungsansatz und Konzeption	15
3.1. Erste analytische Betrachtungen	15
3.2. Lösungsansatz aus theoretischer/mathematischer Sicht	16
3.3. Herangehensweise	17
3.4. Zwischenfazit	18
4.....Entwurf	19
5.....Realisierung und Auswertung	21
5.1. Projektumfeld	21
5.2. Software-Entwurf und –Realisierung	22
5.2.1. Benutzungsoberfläche	22
5.2.2. Prematching	23
5.2.3. Minutiendaten auslesen	26
5.2.4. Minutiendaten aufbereiten	28
5.3. Ergebnisse der Datenerhebung	29
5.3.1. Minutiendaten 10er-Pixel-Intervalle	29
5.3.2. Minutiendaten 20er-Pixel-Intervalle	31
5.3.3. Minutiendaten 40er-Pixel-Intervalle	32
6.....Fazit	33
6.1. Ausblick	35
Anhang	37
Abbildungsverzeichnis	46
Tabellenverzeichnis	47
Literaturverzeichnis	48

---

# Einfluss der Häufigkeit und Verteilung von Minutien auf die Qualität von Algorithmen zur Erkennung von Fingerabdrücken

Vorgelegte Bachelor-Thesis von Sarah Ahmed aus Darmstadt

1. Gutachten: Prof. Dr. rer. nat. Michael Waidner
2. Betreuer: Dr. Dirk Scheuermann, Thomas Kniess

Tag der Einreichung:

---

## Erklärung zur Bachelor-Thesis

Hiermit versichere ich die vorliegende Bachelor-Thesis ohne Hilfe Dritter nur mit den angegebenen Quellen und Hilfsmitteln angefertigt zu haben. Alle Stellen, die aus Quellen entnommen wurden, sind als solche kenntlich gemacht. Diese Arbeit hat in gleicher oder ähnlicher Form noch keiner Prüfungsbehörde vorgelegen.

Darmstadt, den 1. Dezember 2011

---

(Sarah Ahmed)



---

## 1. Einleitung

---

Biometrische Merkmale spielen in vielen Gebieten der Sicherheit und des Alltags mittlerweile eine große Rolle. Insbesondere Fingerabdrücke werden immer häufiger in unterschiedlichsten Bereichen zur Identifikation und Authentifikation eingesetzt.

Als Fingerabdruck wird ein Bild der Papillarleisten der Fingerbeere bezeichnet. Da bisher keine zwei Menschen mit dem gleichen Fingerabdruck bekannt sind, wird dieser als einzigartig für jeden Menschen angesehen.

Auf Fingerabdrücken sind zunächst Schleifen, Bögen und Wirbel erkennbar, welche eine grundlegende Klassifizierung der Abdrücke ermöglicht. Bei genauerer Betrachtung werden Minutien zur aussagekräftigen Unterscheidung genutzt und bei manchen Anwendungen werden zusätzliche Unterscheidungsmerkmale wie die Porenstruktur der Fingerbeere herangezogen (siehe Kapitel 2.2).

Ein Beispiel für den vermehrten Einsatz ist der seit dem 1. November 2007 erhältliche Elektronische Reisepass mit biometrischen Daten (ePass). Auf ihm befindet sich ein Chip, der persönliche Daten, darunter auch elektronisch auswertbare, biometrische Merkmale speichert. Neu am ePass ist, dass er neben den bisher üblichen biometrischen Daten, wie Lichtbild, Körpergröße, Augenfarbe und Unterschriftenprobe auch zusätzlich zwei Fingerabdrücke erfasst, die als Bilder auf dem Chip gespeichert werden. Dies soll das Ausweisdokument stärker an den Inhaber binden und so noch effektiver vor Missbrauch schützen.

Auch auf dem neuen Personalausweis, welcher seit 1. November 2010 den bisherigen ablöst, ist es optional möglich Bilder von zwei Fingerabdrücken zu speichern. Genau wie auch beim ePass dient dies zur Fälschungssicherung, bzw. um die Bindung zwischen Ausweis und Ausweisinhaber zu festigen.

Fingerabdruckleser sind mittlerweile in vielen elektronischen Geräten des Alltags wieder zu finden, z.B. an PCs, Notebooks, Zutrittskontrollsystemen und auch Mobiltelefonen. In Japan werden schon seit einigen Jahren in sehr vielen Mobiltelefonen Fingerabdruckleser zur Authentifizierung des Benutzers eingebaut. Dies soll den Benutzern das Merken und Eingeben von PINs ersparen und somit die Benutzung vereinfachen.

Ein weiteres interessantes Einsatzszenario ist das Bezahlen per Fingerabdruck. In einigen Filialen der Rewe-Gruppe ist die bereits seit Anfang 2010 möglich. Allerdings gibt es Bedenken seitens Datenschützern und Sicherheitsexperten, da der mögliche Schaden bei Missbrauch sehr hoch ist [HEI10].

### 1.1. Motivation

Man kann sagen, dass die Authentifizierung ausschließlich mit Fingerabdrücken zur Sicherheitswahrung in vielen Einsatzgebieten nicht ausreichend ist. Aus diversen Forschungsprojekten, wie beispielsweise dem „Life-Finger 2“ Projekt, das das Bundesamt für Sicherheit in der Informationstechnik (BSI) zusammen mit dem Fraunhofer Institut für Sichere Informationstechnologie SIT im Jahr 2008/09 durchgeführt hat, ergab sich, dass sich viele handelsübliche Sensoren mit Hilfe von Fingerabdruckattrappen überlisten lassen. Dies gelingt selbstverständlich nicht bei allen Sensoren und allen Abdrücken gleichermaßen. Aber bereits der Chaos Computer Club hat mit Hilfe einfachster Mittel gezeigt, wie man selbst den Abdruck eines Politikers mühelos nachahmen und für Identitätsklau missbrauchen kann.

Aus diesem Grunde setzt man die biometrische Erkennung häufig in Kombinationen mit anderen Authentifizierungsmethoden ein. Diese Mehrfaktor-Authentisierung mindert das Risiko einem Missbrauch zum Opfer zu fallen, da ein erfolgreicher Angriff auf einen Authentisierungsfaktor, nicht einen Einbruch im Gesamtsystem nach sich zieht.

---

Trotz der diversen Sicherheits- und Datenschutzbedenken gegenüber biometrischen Authentifizierungssystemen haben diese guten Anklang in der Bevölkerung gefunden. Ein Grund dafür ist die einfache Benutzbarkeit; biometrische Merkmale können nicht vergessen oder verloren werden. Durch die große Akzeptanz ist es unwahrscheinlich, dass die biometrische Authentifikation auf kurze Sicht verschwinden wird. Daher ist es wichtig die existierenden Verfahren und Methoden weiter zu verbessern.

## 1.2. Problemstellung

Fingerabdruck-Authentifizierungssysteme speichern die Fingerabdrücke von allen Nutzern des Systems. Oberflächlich betrachtet: Je größer die Zahl der Nutzer ist, desto höher ist auch die Wahrscheinlichkeit, dass Nutzer verwechselt werden. Doch wie stark wirkt sich die Anzahl der Nutzer tatsächlich auf die Erkennungsleistung eines Fingerabdruck-Authentifizierungssystems aus?

In dieser Arbeit soll ermittelt werden, wie sich die Wahrscheinlichkeit einer korrekten Identifizierung eines Unberechtigten verändert, wenn die dem System bekannte Teilnehmerzahl steigt. Um hierüber eine Aussage machen zu können, wird eine Statistik erstellt, mit der festgestellt werden soll, wie häufig Minutien in einem bestimmten Bereich vorkommen. Minutien können scheinbar beliebig über das Fingerabdruckbild verteilt sein. Legt man über dieses Bild ein Koordinatensystem, so können die Positionen und Bereiche in denen die Minutien erkannt wurden genau ermittelt und festgehalten werden. Doch sind diese Positionen wirklich so zufällig, oder gibt es Bereiche in denen sich Minutien in einem System mit hoher Teilnehmeranzahl häufen? Inwiefern hat das Auswirkungen auf die Erkennungsleistung?

## 1.3. Ziel der Arbeit

Wie in den meisten wissenschaftlichen Abschlussarbeiten, wird auch in dieser Arbeit zu Beginn eine These aufgestellt, die es zu beweisen oder zu widerlegen gilt. In diesem Fall soll dies mit der Aufstellung von Fingerabdruckdaten geschehen, welche wiederum mit Hilfe diverser Grundwerkzeuge und eine, im Zuge dieser Arbeit erzeugten, Hilfsanwendung geschehen.

Wie bereits in der Problemstellung erläutert, soll eine Aussage getroffen werden über das Verhältnis: Identifizierung Unberechtigter  $\leftrightarrow$  Steigende bekannte Teilnehmeranzahl. Über welchen Weg diese Aussage getroffen werden kann und in welchem Ausmaß hängt von den verwendeten Mitteln und den ermittelten Daten ab.

In den folgenden Kapiteln wird die These der Arbeit genau benannt und die verwendeten Werkzeuge erläutert, bzw. das dafür benötigte Grundwissen darüber vermittelt. Ebenso werden die Herangehensweise an die Problemstellung und der hier gewählte Lösungsweg definiert. Die Umsetzung des Lösungsweges mit Hilfe der entwickelten Hilfsanwendung und die Gewinnung der zu betrachtenden Daten ist der für die These der Arbeit entscheidendste Schritt. Jedoch kann eine Behauptung zum Beweis oder zur Widerlegung der These erst danach mit dem Auswerten der Daten getroffen werden, was den Abschluss der Arbeit einleitet, welcher dann einen Ausblick auf mögliche Auswirkungen und mögliche Modifizierungen der Herangehensweise erörtert.

---

## 2. Allgemeine Grundlagen

---

Dieses Kapitel soll vorbereitend auf die darauf folgenden Kapitel sowohl Begriffe als auch verwendete Hard- und Software erläutern und deren Zusammenwirken kurz darstellen. Es wird zu Beginn auf benötigte Definitionen im Bereich der Biometrie eingegangen, dann grundlegendes zu Fingerabdruck-authentisierung vermittelt, um dann auf die, für die Arbeit verwendeten Mittel einzugehen.

Biometrische Systeme zur Benutzererkennung können eine Person entweder identifizieren oder verifizieren, ob es sich bei der Person um eine berechnigte Person handelt.

Bei der Identifikation muss das System die Identität der Person feststellen, welche dieses verwendet. Das bedeutet, dass deren biometrisches Merkmal gegen alle in der vorliegenden Referenzdatenmenge vorhandenen Merkmale abgeglichen werden muss. Dies hat einen 1:n-Vergleich zur Folge.

Bei der Verifikation hingegen werden die Merkmalsdaten lediglich mit dem Datensatz einer einzelnen Person verglichen, da eine Bestätigung benötigt wird, ob es sich bei der Testperson tatsächlich um die Person handelt, die sie angibt zu sein. Dementsprechend findet lediglich ein 1:1-Vergleich statt. Hierfür muss die Person allerdings nicht nur die biometrischen Merkmale preisgeben, sondern zusätzliche Angaben, wie z.B. Name, Ausweis, User-ID oder Ähnliches.

Folglich ist eine Verifikation wünschenswert, aber nicht immer machbar. So kann man beispielsweise in der Kriminalistik beim Auffinden eines Abdrucks an einem Tatort meist diesen nicht direkt Verdächtigen zuordnen und muss diese mit einer Verbrecherdatenbank abgleichen, was einer Identifikation entspricht und aufgrund der Datenmenge sehr langwierig sein kann.

Ob es sich um eine Identifikation oder eine Verifikation handelt, hängt also vom jeweiligen Einsatzgebiet ab.

### 2.1. Biometrische Merkmale und ihre Eigenschaften

Ein biometrisches System authentifiziert den Benutzer, falls seine biometrischen Daten mit Referenzdaten in der vorliegenden Datenbank übereinstimmen.

Biometrische Systeme können sich auf unterschiedliche biometrische Merkmale stützen. Grundsätzlich kann man bei den zur Erkennung verwendeten Merkmalen zwischen physiologischen, auch passive oder statische Merkmale genannt, und verhaltensbasierten Merkmalen, auch aktiven oder dynamische Merkmale genannt, unterscheiden. Physiologische, biometrische Merkmale sind körperbezogene Merkmale und verändern sich nicht oder nur geringfügig über einen langen Zeitraum im Leben eines Menschen.

Beispiele für physiologische, biometrische Merkmale, welche auch bei Erkennungsverfahren eingesetzt werden sind:

- Gesicht
- Iris
- Retina
- Finger
- Handgeometrie
- Venenmuster
- Ohr

Hingegen sind verhaltensbasierte biometrische Merkmale angeeignet und können bei jeder Erkennung variieren, da sich angeeignete Dinge im Laufe des Lebens verändern können.

---

Beispiele hierfür sind:

- Unterschrift (dynamisch/statisch)
- Gestik / Mimik beim Sprechen
- Gang
- Stimme / Sprechverhalten
- Tippverhalten an der Tastatur

Häufig werden diese biometrischen Merkmale für eine aussagekräftigere Erkennung und dadurch sicherere Authentifizierung kombiniert. Beispielsweise die Kombination von Gesichtserkennung und Stimmerkennung.

Alle biometrischen Merkmale können entweder passiv erfasst werden, sozusagen im „Vorbeigehen“, wie z.B. Gesichtserkennung durch eine Kamera am Eingang eines Gebäudes oder aktiv durch Handeln der zu authentifizierenden Person, wie z.B. beim Fingerabdruck.

Biometrische Merkmale sollten sich durch folgende Eigenschaften auszeichnen:

- Eindeutigkeit: die Merkmalscharakteristik muss bei verschiedenen Personen signifikant unterschiedlich sein.
- Universalität: möglichst viele Personen besitzen dieses Merkmal.
- Einzigartigkeit: markant genug, dass man anhand des Merkmals mehrere Personen klar unterscheiden kann.
- Konstanz: möglichst keine oder geringe Veränderungen des Merkmals über einen längeren Zeitraum; Verlust des Merkmals muss miteinkalkuliert werden.
- Messbarkeit: definierbare, technisch messbare Größe (mit Hilfe von Sensoren)
- Anwenderfreundlichkeit: möglichst komfortable Messung des Merkmals, ohne dass die aufzunehmende Person dies als Belästigung empfindet. Des Weiteren sollte das Verfahren für den Nutzer so klar wie möglich nachvollziehbar sein um Vertrauen zu diesem aufzubauen, da private Daten aufgenommen und hinterlegt werden.
- Fälschungssicherheit: ein Diebstahl des ausgewählten Merkmals muss möglichst unterbunden werden, damit eine Nachahmung verhindert wird.

Eine Person kann sich nur dann bei einem biometrischen System authentisieren, falls sie zuvor Daten des benötigten biometrischen Merkmals im System hinterlegt hat. Eine spezielle Form der biometrischen Benutzer-Erkennung ist die Fingerabdruckerkennung.

## 2.2. Fingerabdruck-Erkennung

Die Authentisierung mittels Fingerabdruck funktioniert in der Praxis durch die vorherige Aufnahme (Enrolment) der Fingerabdrücke der später zu authentifizierenden Personen in die Datenbank des biometrischen Erkennungssystems. Hierzu werden immer mehrere Datenaufnahmen gemacht, welche auf verschiedene Art und Weise gesichert werden können. Darauf wird später in diesem Kapitel eingegangen. Mehrere Aufnahmen des gleichen Merkmals, hier des Fingerabdrucks, begünstigen eine bessere Wiedererkennung und vermindern gegebenenfalls sogar Falscherkennungen, abhängig vom Erkennungsalgorithmus.

Um mit Hilfe von Fingerabdrücken Sicherheit zu gewährleisten, werden diese durch Algorithmenberechnungen mit hinterlegten Daten verglichen. Dies geschieht entweder bei aktuell verwendeten Systemen durch Minutiendatenvergleich, da sich ein Bilddatenvergleich in der Vergangenheit nicht bewährt hat.

Beim Minutiendatenvergleich werden besondere Merkmale des Fingerabdrucks, sogenannte Minutien, verwendet um die Fingerabdrücke zu vergleichen. Minutien sind besondere anatomische Merkmale der Papillarlinien des Fingerabdrucks. Die häufigsten Minutien sind standardmäßig, wie in ISO/IEC 19794-2 [ISO19794-2] beschrieben folgendermaßen zu unterteilen:

- Das Papillarlinienende (Ridge Ending) und
- die Gabelung (Ridge Bifurcation)

Definition laut BSI [BSI04]: „Das Papillarlinienende definiert das Ende einer Linie. Die Gabelung ist als Punkt auf der Papillarlinie definiert, an dem sich die Linie in zwei Äste teilt. Die Minuzien sind allgemein stabil und robust hinsichtlich der Bedingungen während der Abtastung. Beispiele sind in der Abbildung 1 aufgezeigt. Minuzien können durch ihren Typ, die x- und y-Koordinate in einem Koordinatensystem und durch Richtung charakterisiert werden. Die Richtungsdefinition ist in der Abbildung 2 dargestellt.“

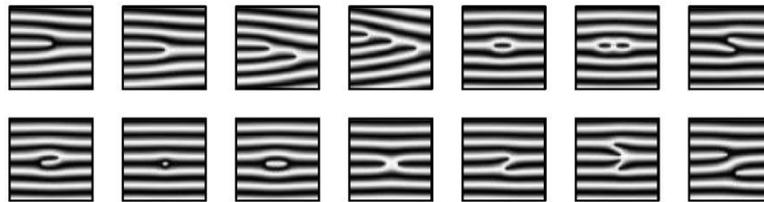


Abbildung 1: Zeilenweise von links nach rechts: Papillarlinienende, einfache Gabelung, zweifache Gabelung, dreifache Gabelung, einfacher Wirbel, zweifacher Wirbel, seitliche Berührung; Haken, Punkt, Intervall, X-Linie, einfache Brücke, zweifache Brücke, fortlaufende Linie

„Falls zwei Fingerabdrücke zur gleichen Kategorie gehören und eine bestimmte Anzahl übereinstimmender Minuzien enthalten, kann mit bestimmter Sicherheit gesagt werden, dass sie von dem gleichen Finger stammen.“

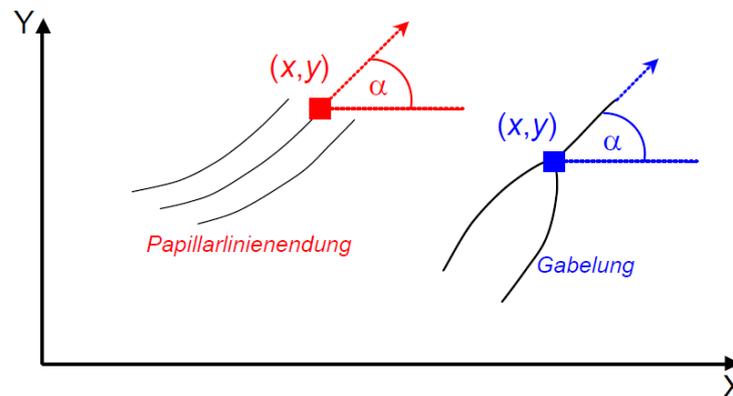


Abbildung 2: Papillarlinienendung und Gabelung

Hierfür werden verschiedene Algorithmen verwendet, einige werden in Abschnitt 2.6 näher betrachtet und erläutert. Es gibt verschiedene Algorithmen-Techniken für den Vergleich von Fingerabdrücken (sogenanntes Matching). Diese sind:

- Direkter, optischer Vergleich: Ein direkter Bildvergleich entweder „mit bloßem Auge“ oder mit einem Algorithmus. Dies ist jedoch eher ungeeignet für größere Datenmengen, daher bestehen weder praktischer Nutzen noch Einsatzgebiete.
- Henry-System: Um die Suche in großen Datenbanken gering zu halten wird die generelle Form eines Abdrucks mit den wichtigsten Eigenschaften gespeichert, hierbei kommt das Henry-System [HEN03] zum Einsatz: Wirbel, Schleifen und Bögen.
- Position und Richtung der Minutien: Die meisten Algorithmen nutzen Minutien für die Wiedererkennung von Fingerabdrücken. Lediglich Position und Richtung dieser werden für spätere Vergleiche gespeichert.
- Zählen der Papillarlinien: Einige Algorithmen zählen die Papillarlinien zwischen markanten Punkten (meist Minutien), anstatt der Entfernungen zwischen den Punkten.
- Struktur und Formen der Papillarlinien: Mustervergleichsalgorithmen verwenden die grundlegende Struktur und Formen der Papillarlinien. Dabei wird der Fingerabdruck in Segmente unterteilt und die Richtung der Papillarlinien nebst Phase und Winkel extrahiert und gespeichert. Als Beispiel hierfür gilt der Standard ISO19794-3, welcher sich mit dem Thema „Finger pattern spectral data“ befasst [ISO19794-3].

Sehr häufig werden die oben genannten Techniken kombiniert angewandt.

## 2.3. Sensor-Technologien zur Fingerabdruck-Erkennung

Zur Fingerabdruckerkennung stehen durch verschiedene Entwicklungsansätze inzwischen sehr viele Technologien zur Verfügung. Grundsätzlich lassen sich die Fingerabdrucksensoren, welche zum Einlesen der Daten verwendet werden, in drei Bauweisen unterscheiden: Flächensensoren, Zeilensensoren und berührungslose Sensoren. Einige, wenige Technologien lassen sich in mehr als einer Bauart realisieren und bieten somit flexible Einsatzmöglichkeiten. Folgende Technologien werden derzeit eingesetzt:

- a) Optische FTR (frustrated total internal reflection)
- Arbeiten mit dem Prinzip der gestörten Totalreflektion mit Hilfe einer CMOS-Kamera und einem Prisma oder einer Prismenfolie. Beim Einlesen des Fingerabdrucks muss der Finger auf einer Glasscheibe aufliegen. Diese Technologie gibt es nur als Flächensensor. Beispiele: Cross Match LSCAN 100, Dermalog ZF1  
Einsatzgebiete: Zugangskontrolle, Erfassung Fingerabdrücke für ePass

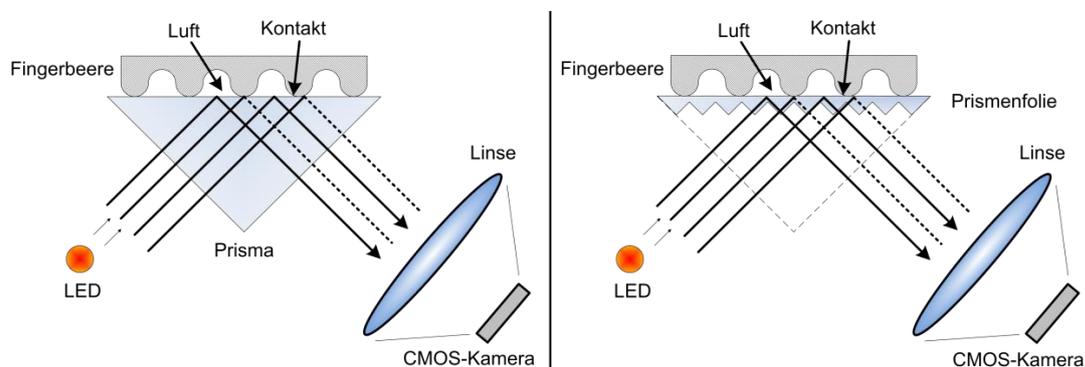


Abbildung 3: Grundaufbau eines FTR-Sensors mit Prisma; Grundaufbau eines FTR-Sensors mit Prismenfolie

- b) Optisch ohne FTR (kontaktlos)
- Hierbei wird der Finger beleuchtet und mit Hilfe einer CMOS-Kamera das Bild aufgenommen. Jedoch handelt es sich hierbei meist um berührungslose Sensoren, wobei der Finger mit Hilfe eines Abstandhalters gezwungenermaßen von der Kamera ferngehalten wird. Um diese nicht zu beschädigen ist sie jedoch zusätzlich mit einem Schutzglas abgedeckt. Beispiel: TST BIRD 3  
Einsatzgebiete: Zugangskontrolle, Gesundheitswesen, Finanzapplikationen, Office-Anwendungen

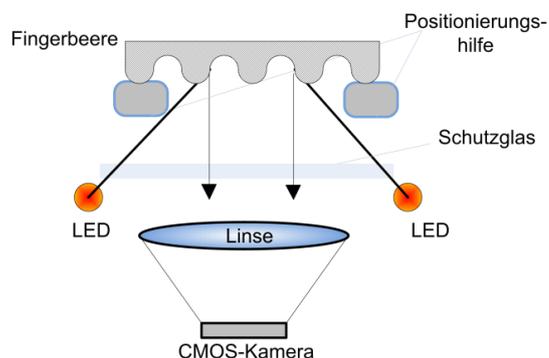


Abbildung 4: Grundaufbau eines optischen Sensors ohne FTR

- c) Kapazitiv
- Diese Technologie gibt es mittlerweile nicht nur als Flächensensor, wie zu Beginn, sondern auch als Zeilensensor. Grundsätzlich besteht die Oberfläche des Sensors aus einer Siliziumfläche, welche Kapazitäten misst und so ein Bild erzeugt. Genauer betrachtet besteht die Sensorfläche aus vielen einzelnen in einer Matrix angeordneten Kondensatorplatten (Elekt-

roden). Der Finger bildet dann beim Auflegen das Gegenstück, d.h. die gegenüberliegenden Elektroden. Je mehr solcher Platten die Fläche beinhaltet, desto höher ist die Auflösung und somit Genauigkeit des aufgenommenen Bildes.

Beispiele: Fujitsu MBF-200 (Flächensensor), Upek Eikon II (Zeilensensor)

Einsatzgebiete: PC-Zugang, Datenbank- und Netzwerkzugang, Webseitenzugang, Biometrische Ausweise

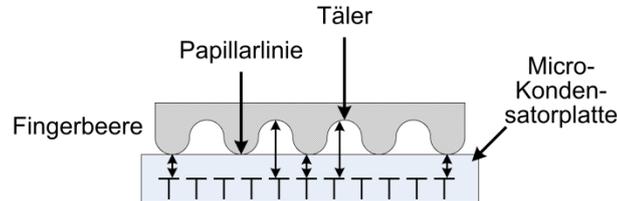


Abbildung 5: Grundaufbau eines kapazitiven Sensors

d) Thermisch

Sensoren dieses Typs erstellen ein Bild aufgrund von Temperaturdifferenzen zwischen den Papillarlinien und der Luft in den Tälern der Fingerbeere. Diese Technologie wird als Zeilensensor realisiert, da die Temperatur sich auf einer Fläche zu schnell angleichen würde.

Beispiel: Bergdata FCAT-100

Einsatzgebiete: PC-Zugang, Datenbank- und Netzwerkzugang, Webseitenzugang, Biometrische Ausweise

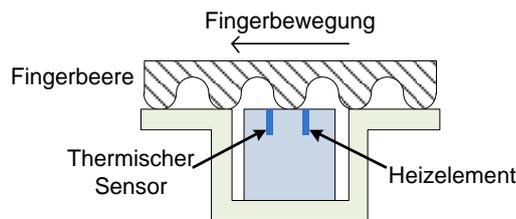


Abbildung 6: Grundaufbau eines thermischen Sensors

e) RF-Feld (radio frequency)

Diese Sensortypen werden des Öfteren mit kapazitiven verwechselt, wobei sie lediglich gemein haben, dass elektrische bzw. elektromagnetische Signale als physikalische Messgrößen zur Ermittlung des Fingerabdruckbildes herangezogen werden. Beim kapazitiven werden lediglich die Aufladungen zwischen Sensoroberfläche und Fingerbeere gemessen, hingegen misst der RF-Feld-Sensor in die Tiefe und wertet die von den inneren Hautschichten reflektierten RF-Signale aus.

Beispiel: Authentec AES2501 (Zeilensensor), Authentec AES3400 (Flächensensor)

Einsatzgebiete: PC-Zugang

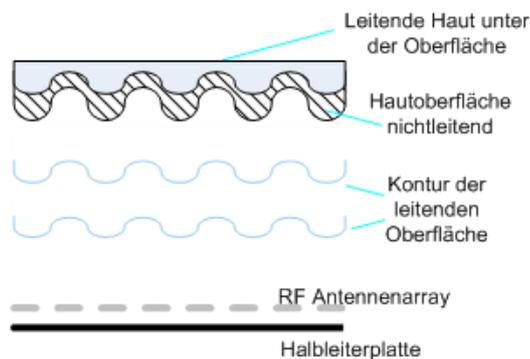


Abbildung 7: Grundaufbau eines RF-Sensors

f) Ultraschall

Diese Sensoren arbeiten mit Aussendung von Ultraschallwellen und Interpretierung der reflektierten Wellen, welche unterschiedliche Dichte aufweisen, je nachdem ob es sich um ein

Tal oder eine Papillarlinie auf der Fingerbeere handelt.  
Beispiel: Ultra Scan Ultra-Touch  
Einsatzgebiete: Gesundheitswesen, PC-Zugang, Zutrittskontrollsystem

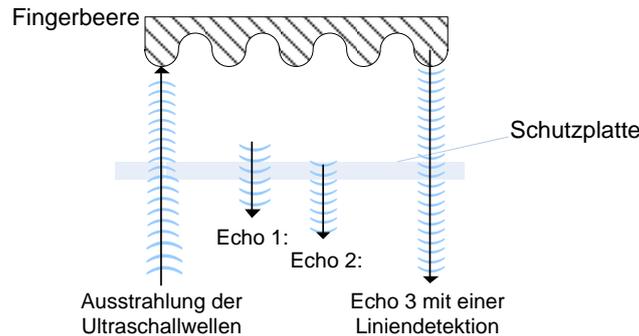


Abbildung 8: Grundaufbau eines Ultraschallsensors

Noch zu erwähnen ist, dass einige Hersteller bei ihren Sensoren zusätzliche Lebenderkennungsverfahren integrieren. Hierbei wird entweder mit zusätzlichen Sensoren gearbeitet, die für den Nutzer unmerklich weiterführende Daten aufnehmen (wie z.B. Puls) oder es wird auf andersartige Merkmale zurückgegriffen, unter Umständen auch auf systemexterne Lebenderkennung (z.B. durch überwachendes Personal). Es hat sich jedoch bisher keine integrierte Lebenderkennung etabliert, welche als hundertprozentig korrekt angenommen werden kann.

Im Folgenden wird auf die Sensoren an sich nicht mehr detailliert eingegangen, da dies nicht Bestandteil der Untersuchungen in dieser Bachelor-Arbeit ist. Für die Bearbeitung der Arbeit ist es irrelevant woher die Fingerabdrücke stammen, bzw. mit welchem Sensor sie aufgenommen wurden. Die zu untersuchenden Daten sollten allerdings von gleicher Qualität sein, um ein aussagekräftiges Ergebnis zu erzielen.

Die für diese Arbeit verwendeten Fingerabdrücke stammen vom „Biometric System Laboratory – University of Bologna“, welche Fingerabdruckdatenbanken in Form von Bildern für die immer wieder stattfindenden „International Fingerprint Verification Competition“ (FVC) [FVC] bereitstellt.

## 2.4. BioAPI Framework

Das Biometric Application Programming Interface (BioAPI) [ISO19784-1] ist ein wichtiger Bestandteil der internationalen Standards, welche biometrische Systeme mit Enrolment und Verifikation, bzw. Identifikation unterstützen.

Der Zweck der BioAPI Spezifikation besteht darin die nötigen Schnittstellen zur Verfügung zu stellen, damit biometrische Anwendungen in Module verschiedener Hersteller integriert werden können. Gerade in der heutigen Zeit ist die dadurch erzeugte Flexibilität von großer Wichtigkeit, da sich sowohl Hardware, als auch Software sehr schnell und getrennt voneinander entwickeln. So entstanden in den letzten 15 bis 20 Jahren sehr viele verschiedene Technologien für Fingerabdrucksensoren, aber auch die Algorithmen zur Erkennung von Personen anhand von Fingerabdrücken haben sich stark verändert. Wobei das BioAPI nicht auf Fingerabdrücke beschränkt ist, sondern zahlreiche Arten von biometrischen Verfahren unterstützt.

Die in Abbildung 9 dargestellte Grafik beschreibt vereinfacht die Architektur des BioAPI 2.0. Hierbei sind die biometrischen Anwendungen unabhängige Komponenten, welche mit dem BioAPI Framework interagieren. Dieses Framework wiederum sendet Meldungen an die einzelnen Biometric Service Provider (BSP), welche sowohl Aufnahmesensoren/-Geräte, als auch Merkmalsextraktionen, Merkmalsuche, Merkmalvergleiche oder andere Dienste zur Verfügung stellen können.

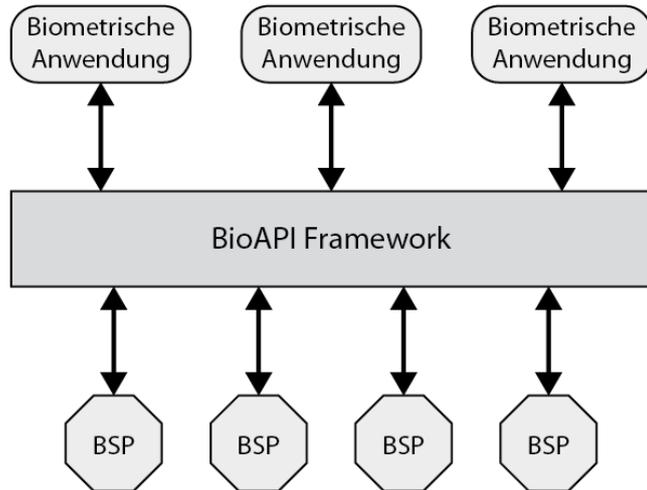


Abbildung 9: BioAPI Architektur

Bei dem BioAPI handelt es sich um ein Framework, also lediglich ein Rahmengerüst, aber kein fertiges Programm, welches direkt verwendet werden kann. Erst eine Middleware macht also die tatsächlichen biometrischen Berechnungen mit Hilfe von BSPs möglich. Die hier verwendete Middleware ist secunet biomiddle, die bereits in vorangegangenen Projekten mit dem BSI erfolgreich eingesetzt wurde[LF2].

## 2.5. Middleware – secunet biomiddle

„secunet biomiddle ermöglicht in den verschiedenen Biometrie- oder eID-Anwendungen den modularen Einsatz von biometrischen Systemkomponenten und Ausweislesegeräten. Beide Funktionalitäten werden über eine einheitliche Schnittstelle bereitgestellt. Durch die Verwendung international standardisierter Schnittstellen können Teilkomponenten jederzeit ausgetauscht werden. Secunet biomiddle kommuniziert über eine serviceorientierte Schnittstelle mit Client-Anwendungen und ist damit unabhängig von Systemplattformen und Programmiersprachen. Als gemeinschaftliche Entwicklung mit dem Bundesamt für Sicherheit in der Informationstechnik (BSI) ist secunet biomiddle international als Referenzimplementierung und bevorzugte Architektur für den Einsatz von Biometrie im Zusammenhang mit hoheitlichen Dokumenten anerkannt.“ [SECFS]

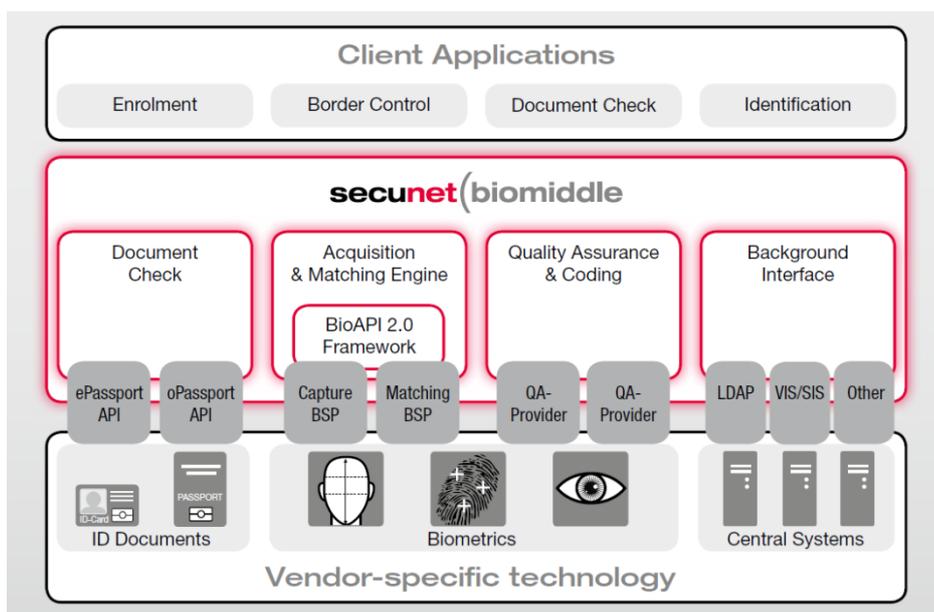


Abbildung 10: secunet biomiddle Architektur [SECFS]

---

Die Middleware von secunet findet im Zuge dieser Arbeit Anwendung, insbesondere bei den Matchingberechnungen und dem Extrahieren der Minutien als Koordinaten. Die tatsächlichen Berechnungen führt ggf. der ausgesuchte Algorithmus aus, speziell beim kompletten Matching ohne Zugriff auf die einzelnen Minutiendaten. Jedoch müssen die angebotenen Dienste des BioAPI mit dem Algorithmus-BSP verknüpft werden, was mit Hilfe der biomiddle geschieht.

## 2.6. Algorithmen

Da bei dieser Arbeit lediglich Fingerabdrücke als biometrische Merkmale beleuchtet werden, insbesondere Vergleiche dieser und die möglichen Falscherkennungsraten, sind Algorithmen in Bezug auf andere biometrische Merkmale hier irrelevant und werden auch nicht betrachtet. Die folgenden Algorithmen sind beispielhaft für minutienbasierte Matching-Algorithmen:

- Siemens SmartMatcher  
Algorithmus von Siemens, welcher so wenig Ressourcen benötigt, dass er „onCard“ verwendet werden kann, also als Matcher auf einer Smartcard. [SIEM]
- L-1 Identix  
Dieser Algorithmus ist Teil des von L-1 Identity Solutions entwickelten BioEngine<sup>®</sup> Software Development Kits. Dieser Algorithmus ist nicht frei verfügbar, sondern nur mit Lizenz verwendbar. [L1]
- NIST Bozorth3  
Vom amerikanischen National Institute for Standards and Technology – kurz NIST – spezifizierter und implementierter, freier Algorithmus. Der Name stammt von Allan S. Bozorth, welcher einen Fingerabdruck Matcher entwickelt hatte während er beim Federal Bureau of Investigation (FBI) war, auf diesem basiert der heutige Bozorth3 Algorithmus [NIST04-2].  
Ursprünglich war dieser Algorithmus Teil des NFIS 2 Software Pakets. Mittlerweile heißt das Paket NBIS (NIST Biometric Image Software). Dieses Software Paket wurde für das FBI und das Department of Homeland Security (DHS) entwickelt [NBIS].

Zwei der Algorithmen, L-1 Identix und Bozorth3, wurden bereits bei Studien des BSI verwendet, um die Robustheit von Fingerabdrucksensoren in Bezug auf Attrappen zu testen [LF2].

Da es sich beim NIST Bozorth3 um einen nicht-kommerziellen, also frei verwendbaren Algorithmus handelt, wurde dieser beispielhaft für die Berechnungen in dieser Arbeit verwendet. Des Weiteren ist das NIST eine anerkannte Institution, was die Qualität des Algorithmus bekräftigt.

### 2.6.1. NIST Bozorth3 Funktionsweise

Der Algorithmus kann zu den Minutien jedes Fingerabdruckpaares einen Matchscore errechnen und so einen aussagekräftigen Wert darüber liefern, ob diese beiden Abdrücke vom selben Finger stammen oder nicht.

Der Algorithmus ist invariant gegenüber Translation und Rotation. Es werden lediglich die Lage der Minutie mit den Koordinaten und die Orientierung verwendet, also  $\{x, y, t\}$  mit x- und y-Koordinate und theta. Die grundsätzliche Funktion des Algorithmus sieht folgendermaßen aus:

1. Erzeugung einer Intra-Fingerabdruck-Minutienvergleichstabelle:  
Eine Tabelle für den Prüffingerabdruck und eine Tabelle für jeden Galeriefingerabdruck.  
⇒ Es werden die relativen Maße von jeder Minutie zu allen anderen Minutien innerhalb eines Fingerabdrucks berechnet. Diese werden dann in einer Tabelle gespeichert und stellen die Invarianz des Algorithmus gegenüber Rotation und Translation sicher.  
In Abbildung 11 A sind beispielhaft zwei Minutien dargestellt. Die Minutie k in der linken unteren Ecke des Fingerabdrucks wird durch die Koordinaten  $(x_k, y_k)$  und die nach unten und rechts zeigende Pfeillinie, welche die Orientierung  $t_k$  repräsentiert, dargestellt. Eine zweite Minutie j oben rechts mit der nach oben rechts zeigenden Orientierung. Um die relative, verschobene Position zu berechnen, wird die Distanz  $d_{kj}$  zwischen den beiden Minutien ermittelt. Diese Entfernung bleibt relativ konstant zwischen zugehörigen Punkten auf zwei

verschiedenen Fingerabdruckbildern unabhängig der Translation und Rotation zwischen den Bildern.

Für die Abhängigkeit zweier Minutien reicht allerdings die Distanz alleine nicht aus, da ansonsten lediglich die Koordinaten und die Minutienorientierungen gespeichert werden. Es werden also noch Winkel benötigt zwischen der Minutienorientierung und der Distanzverbindung zur anderen Minutie (bei paarweiser Betrachtung). In Abbildung 11 A stellt  $\Theta_{kj}$  den Winkel dar, welcher aus den beiden Winkeln  $\beta_k$  und  $\beta_j$  folgendermaßen errechnet wird.

$\beta_1 = \min(\beta_k, \beta_j)$  und  $\beta_2 = \max(\beta_k, \beta_j)$ , so dass  $\beta_1 = \beta_k$  und  $\beta_2 = \beta_j$  in Abbildung 11 A gilt. Für jeden paarweisen Minutienvergleich gibt es in der Vergleichstabelle einen Eintrag  $\{d_{kj}, \beta_1, \beta_2, k, j, \Theta_{kj}\}$ . Die Einträge werden mit ansteigender Distanz in der Vergleichstabelle gespeichert. Die Tabelle wird in Abhängigkeit einer festen Distanz als Grenze beschnitten. Für jeden Fingerabdruck gegen den abgeglichen werden soll, muss eine solche Vergleichstabelle mit paarweisen Minutienbetrachtungen erstellt werden.

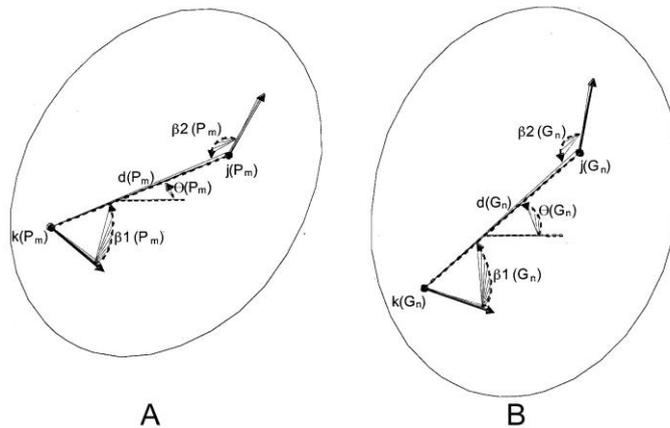


Abbildung 11: Minutienvergleich innerhalb eines Fingerabdrucks, bzw. Darstellung zweier Bilder des gleichen Fingerabdrucks (P = Prüffingerabdruck, G = Galeriefingerabdruck)

## 2. Konstruktion einer Inter-Fingerabdruck-Kompatibilitätstabelle:

Vergleich einer Prüffingerabdruck-Minutienvergleichstabelle mit einer Galeriefingerabdruck-Minutienvergleichstabelle, um eine neue Kompatibilitätstabelle zu erzeugen.

⇒ Es werden die Vergleichstabellen der zwei zu betrachtenden Fingerabdruckbilder betrachtet und nach „kompatiblen“ Einträgen gesucht. Abbildung 11 zeigt zwei Darstellungen des gleichen Fingers mit Veränderung durch Rotation, Translation und leichter Skalierung. Die beiden dargestellten Minutien k und j sind abhängig voneinander und in beiden Abdruckbildern „identisch“. Teil A auf der linken Seite ist der Prüffingerabdruck P, es handelt sich bei dem betrachteten Minutienpaar um das m-te Paar, also den m-ten Eintrag in der Vergleichstabelle des Fingers. Analog dazu ist rechts in Teil B der Galeriefingerabdruck G mit dem n-ten Eintrag wiedergegeben. Der NIST Bozorth3 führt drei Tests durch, welche eine Aussage darüber treffen, ob die Tabelleneinträge von  $P_m$  und  $G_n$  „kompatibel“ zueinander sind:

- i. Der erste Test prüft, ob die abhängigen Distanzen innerhalb einer spezifizierten Toleranzgrenze  $T_d$  liegen.
- ii. Die letzteren beiden Tests prüfen, ob die relativen Minutienwinkel innerhalb einer spezifizierten Toleranz  $T_\beta$  liegen.

Es ergeben sich durch diese Tests folgende Differenzfunktionen:

$$\Delta_d(d(P_m), d(G_n)) < T_d$$

$$\Delta_\beta(\beta_1(P_m), \beta_1(G_n)) < T_\beta$$

$$\Delta_\beta(\beta_2(P_m), \beta_2(G_n)) < T_\beta$$

Falls die relative Distanz und die Minutienwinkel zwischen zwei Einträgen der beiden Vergleichstabellen innerhalb einer akzeptablen Toleranz liegen, dann wird folgender Eintrag in die Kompatibilitätstabelle eingefügt:

$$\{\Delta_\beta(\Theta(P_m), \Theta(G_n)), k(P_m), j(P_m), k(G_n), j(G_n)\}$$

Es werden also zwei Paare von Minutien, also vom Prüffingerabdruck  $P$  und vom Galeriefingerabdruck  $G$ , einbezogen. Der Eintrag sagt weiterhin aus, dass die Minutie  $k(P_m)$  der Minutie  $k(G_n)$  entspricht, analog dazu entspricht  $j(P_m)$  der Minutie  $j(G_n)$ . Der erste Term des Eintrags,  $\Delta_\beta(\Theta(P_m), \Theta(G_n))$ , wird später im Algorithmus benötigt, um Gruppen mit ähnlichen globalen Rotationen zwischen „kompatiblen“ Prüf- und Galerie-Minutien zu bilden.

### 3. Traversierung der Inter-Fingerabdruck-Kompatibilitätstabelle

- a. Durchlaufen und Tabelleneinträge zu Gruppen zusammenfügen.
- b. Kombinieren von kompatiblen Gruppen und ansammeln eines Match Scores.

⇒ Zu Beginn dieses letzten, dritten Schrittes liegt dem Algorithmus eine Kompatibilitätstabelle vor, welche eine Liste von Kompatibilitätsverbindungen zwischen zwei Paaren potenziell übereinstimmender Minutien beinhaltet.

Diese Verbindungen verkörpern einzelne Verknüpfungen in einem Kompatibilitätsgraphen. Um festzustellen wie gut die zwei Fingerabdrücke übereinstimmen miteinander wäre ein guter Ansatz, den Kompatibilitätsgraphen zu traversieren, um den längsten Pfad von verknüpften Kompatibilitätsverbindungen zu finden. Um diesen Ansatz zu erreichen, sind einige erhebliche Herausforderungen zu bewältigen:

1. Die Kompatibilitätstabelle ist kein zusammenhängender Graph, sondern eher eine disjunkte Sammlung von einzelnen Verbindungen innerhalb eines Graphen.
2. Jeder Knoten des Graphen ist möglicherweise mit vielen anderen Knoten verbunden.
3. Dies kann zu Rundwegen im Graph führen.
4. Es gibt keinen eindeutigen Wurzelknoten im Graphen, welcher voraussichtlich zum maximalen Pfad führen kann.
5. Einschließungen und/oder Poren innerhalb einer der beiden zu vergleichenden Fingerabdrücke führen zu Unterbrechungen im Graphen.

Um diese Fragen zu klären, hat Allan Bozorth einen Algorithmus implementiert, welcher die Kompatibilitätstabelle so bearbeitet, dass die Betrachtungen von verschiedenen Startpunkten aus initiiert werden. Während diese Betrachtungen durchgeführt werden, bilden sich Teile bzw. Gruppen des Kompatibilitätsgraphen durch Verknüpfen der Einträge in der Tabelle. Wenn alle Betrachtungen beendet sind, werden „kompatible“ Gruppen (Cluster) kombiniert und die Anzahl der verknüpften Tabelleneinträge über diese kombinierten Cluster werden gesammelt, um den Match Score zu bilden. Je größer die Anzahl von verknüpften Kompatibilitätsverbindungen, desto größer der Match Score, und desto sicherer ist es, dass die beiden Fingerabdrücke von der gleichen Person, dem gleichen Finger sind.

Diese Arbeitsweise des NIST Bozorth3 stammt aus dem „User’s Guide to Export Controlled Distribution of NIST Biometric Image Software (NBIS-EC)“ [NIST04-1] und wurde hier lediglich zusammengefasst und übersetzt wiedergegeben.

## 2.7. Qualitätsabschätzung und Sicherheit

In der Biometrie kann man nicht immer von optimalen, immer gleichen Testbedingungen ausgehen, was dazu führen kann, dass die Messergebnisse je nach Bedingungen schwanken und man dadurch nicht hundertprozentige Übereinstimmungen erzielen kann. Um biometrische Systeme bewerten zu können, werden verschiedene Fehlerraten herangezogen:

Wird ein bestimmter Schwellwert vom Administrator des biometrischen Systems festgelegt, so werden alle Personen, die einen Übereinstimmungsgrad unter diesem Schwellwert haben, vom System abgelehnt.

- False Acceptance Rate (FAR)

Es ist in dem oben genannten System möglich, eine Abschätzung darüber zu machen, wie groß in dem Fall die Wahrscheinlichkeit ist, dass eine nicht erfasste Person zugelassen wird. Der prozentuale Anteil fälschlich zugelassener Unberechtigter beschreibt die FAR. Definition:

$$FAR = \frac{NFA}{NIA} * 100\%$$

NFA: Anzahl fälschlicher Akzeptanzen; NIA: Gesamtanzahl unberechtigter Zutrittsversuche  
 Je größer der Schwellwert und damit der geforderte Übereinstimmungsgrad eines Datensatzes mit dem Referenzdatensatz gewählt wird, desto größer wird die Zahl der unberechtigten und damit falschen Zurückweisungen.

- False Rejection Rate (FRR)

In dem gleichen System lässt sich eine Abschätzung darüber machen, wie groß die Wahrscheinlichkeit ist, dass die zugelassene Testperson abgelehnt wird. Die FRR beschreibt den prozentualen Anteil fälschlich zurückgewiesener Berechtigter. Definition:

$$FRR = \frac{NFR}{NEA} * 100\%$$

NFR: Anzahl fälschlicher Rückweisungen; NEA: Gesamtanzahl berechtigter Zutrittsversuche  
 Je kleiner der Schwellwert und damit der geforderte Übereinstimmungsgrad eines Datensatzes mit dem Referenzdatensatz gewählt wird, desto größer wird die Zahl der Falsch-Akzeptanzen.

- Equal Error Rate (ERR)

Der Administrator eines solchen biometrischen Systems muss bei der Wahl des Schwellwertes zwischen Komfort und Sicherheit abwägen. Einen Maßstab gibt dafür die ERR welche sich genau am Schnittpunkt der beiden Fehlerraten FAR und FRR befindet, also durch die Bedingung  $FAR = FRR$  definiert ist.

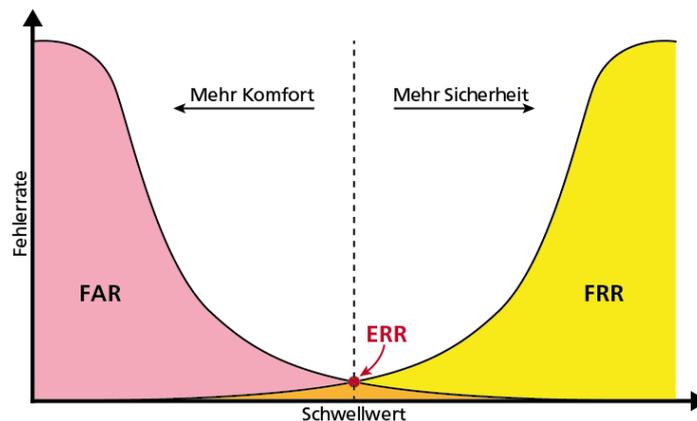


Abbildung 12: Fehlerkurven bei biometrischen Verfahren

In Abbildung 12 sind die Abhängigkeiten der Fehlerraten dargestellt. Wird der Schwellwert erhöht, die gestrichelte Linie also nach rechts verschoben, so steigt die FRR und die FAR fällt, demnach wird die Prüfung der Testperson strenger. Genau umgekehrt ist dies, wenn der Schwellwert gesenkt wird, was zur Folge hätte, dass mehr unberechtigte Personen vom System akzeptiert werden.

- Failure to Acquire Rate (FTA)

Bei den Fehlerraten FAR und FRR ist die FTA mit zu beachten, da es auch vorkommen kann, dass bei der Identifikation oder Verifikation die Aufnahme der Daten nicht korrekt verläuft und man fehlerhafte oder gar keine Daten zum Abgleich zur Verfügung hat.

$$FTA = \frac{NNA}{NPU} * 100\%$$

NNA: Anzahl der Personen, bei der die Aufnahme der aktuellen Daten misslang. NPU: Gesamtanzahl der potentiellen Nutzer innerhalb der Organisation des Betreibers (number of potential users).

Geht man nun davon aus, dass die Aufnahme der Daten erfolgreich war, also dem eher unrealistischen Idealfall, so spricht man von den folgenden beiden Raten als Ersatz von FAR und FRR:

- 
- False Match Rate (FMR)  
„Wahrscheinlichkeit, dass die von einem unberechtigten Benutzer aktuell aufgenommenen Daten fälschlicherweise eine korrekte Übereinstimmung mit den Referenzdaten eines berechtigten Benutzers ergeben.“ Zusammenhang zur praktischen Fehlerrate FAR:

$$FAR = (1 - FTA) * FMR$$

- False Non-Match Rate (FNMR)  
„Wahrscheinlichkeit, dass die von einem berechtigten Benutzer aktuell aufgenommenen Daten fälschlicherweise keine korrekte Übereinstimmung mit den Referenzdaten ergeben.“ Zusammenhang zur praktischen Fehlerrate FRR:

$$FRR = FTA + (1 - FTA) * FNMR$$

Es gibt noch weitere Raten, die hier aber nicht betrachtet werden, da sie keine Anwendung in dieser Arbeit finden. Die Beschreibung und Definitionen der hier aufgezählten Raten stammen aus dem TeleTrust Kriterienkatalog [TT06].

---

### 3. Lösungsansatz und Konzeption

---

Im vorangegangenen Kapitel wurde der Grundstein gelegt, um die in der Arbeit vorliegende Fragestellung, nach dem Zusammenhang zwischen den Akzeptanzraten eines biometrischen Systems und der Zahl der registrierten Nutzer wissenschaftlich und technisch betrachten zu können. Bereits in der Einleitung wurden diese Problemstellung und eine Reihe von Fragestellungen, die mit diesem Problem einhergehen kurz dargestellt. Dieses Kapitel konkretisiert nun diese und teilt das „Problem“ in kleinere Teilprobleme auf. Ziel ist die Spezifikation eines methodischen Ansatzes, der sich eignet, eine qualifizierte Aussage darüber zu treffen, ob eine steigende Teilnehmeranzahl einen signifikanten Einfluss auf die Falschakzeptanzrate hat und somit ein Urteil über den verwendeten Algorithmus getroffen werden kann.

Die These der Arbeit lässt sich demnach formulieren als: Steigende Fingerabdruckanzahl in vorliegender Datenbank hat einen signifikanten Einfluss auf die Falschakzeptanzrate (FAR), bzw. erhöht diese signifikant.

Welche Fragestellungen/Herausforderungen für die Arbeit resultieren aus dieser These? Dieses Kapitel widmet sich genau dieser Frage und erörtert erste Ansätze zu der These.

In diesem Kapitel werden lediglich Ansätze betrachtet, die zur letztendlichen möglichen Lösung führen sollen. Die Überlegungen können sich während der Betrachtung schnell als nicht umsetzbar oder nicht verwendbar herausstellen, was aber zum Prozess der Lösungsfindung dazu gehört.

Als Voraussetzung für diese Kapitel gilt die Verwendung der Fingerabdruckdatenbanken des „international Fingerprint Verification Competition“ (FVC) [FVC] mit jeweils 8 Bildern pro Finger. Bei den Datenbanken des FVC handelt es sich um jeweils ca. 100 Fingerabdrücke á 8 Bilder. Diese Datenbank wird verwendet, da es sich um die größte, frei verwendbare Datenbank von Fingerabdrücken handelt.

Es kann bei den Bildern von einer durchschnittlich guten Qualität der Fingerabdrücke ausgegangen werden. Für die analytische Betrachtung in dieser Arbeit wird aus den 8 Bildern ein Bild als Repräsentant ausgewählt. Dies wird durchgeführt, da die Berechnungen sonst den Rahmen der Arbeit sprengen würden und es in dieser Arbeit um die Auswirkungen einer ansteigenden Nutzeranzahl geht und diese durch einen Fingerabdruck guter Qualität repräsentativ dargestellt werden können.

Die Vorgehensweise bei der Auswahl und die genaue, technische Umsetzung der Auswahl wird in Kapitel 4 näher erläutert. Nach der Auswahl liegen für die Betrachtungen also genau so viele Fingerabdruckbilder wie Fingerabdrücke vor. Die Anzahl der Finger in den verschiedenen Datenbanken des FVC variiert, was aber nicht relevant ist für die hier durchgeführten Betrachtungen, da es lediglich darum geht, die Anzahl zu erhöhen und festzustellen, was dies für Auswirkungen auf die Gesamtauswertung hat.

#### 3.1. Erste analytische Betrachtungen

Fingerabdrücke werden in der Analyse dieser Arbeit als Fläche betrachtet. Die Betrachtung eines Fingerabdrucks als Fläche ermöglicht es das Fingerabdruckbild auf ein Koordinatensystem zu legen. Somit sind alle markanten Punkte (d.h. Minutien) dieser Fläche als Koordinatenpaare zu verstehen. Jedes Koordinatenpaar entspricht einem Pixel.

Sollen zwei Fingerabdrücke miteinander verglichen werden, so spielen einige Faktoren eine entscheidende Rolle. Meist sind Abdrücke nicht auf dem gleichen Mittelpunkt, d.h. sie sind verschoben oder auch verdreht zueinander. Der in dieser Arbeit verwendete Matching-Algorithmus (Bozorth3, siehe hierzu Kapitel 2.6.1) ist in der Lage dieses auszugleichen, indem er die Minutien für einen Vergleich in Abhängigkeit voneinander verwendet. Für direkte Vergleiche ist die Nutzung eines Matching-Algorithmus also unumgänglich. Es ist nicht Aufgabe dieser Arbeit einen neuen Algorithmus hierzu zu implementieren.

---

Jedoch ist ein direkter Vergleich von zwei Abdrücken nicht relevant für die Betrachtungen an dieser Stelle. Daher ist die Verwendung eines Matching-Algorithmus‘ hier nicht sinnvoll, weil die Daten bereits vor der Erkenntnis, ob die Finger matchen oder nicht, betrachtet werden sollen. Der Matching-Algorithmus findet lediglich im Vorfeld Verwendung beim Auswählen der Repräsentanten für die Finger aus den erwähnten 8 Bildern.

### 3.2. Lösungsansatz aus theoretischer/mathematischer Sicht

Es ist anzunehmen, dass es schwieriger wird Fingerabdrücke voneinander zu unterscheiden, je dichter ihre Minutien zusammen liegen. Im Extremfall können sich Minutien auch so stark häufen, dass regelrechte Minutienwolken entstehen. Es wird angenommen, dass sich diese Minutienwolken negativ auf die Akzeptanzraten in einem System auswirken. Aus diesem Grund wird untersucht, ob sich Minutien an bestimmten Stellen im Fingerabdruckbild häufen. Aus dieser Häufung kann man folgende Annahme formulieren:

Minutienwolken haben Einfluss auf die Wieder- oder auch Falscherkennung eines Fingerabdrucks, bzw. erhöhen die Falscherkennung signifikant.

Treffen also Minutienwolken beim Fingerabdruckbild und bei einem Referenzbild zusammen, so ist zu vermuten, dass die Wahrscheinlichkeit steigt, dass beide als derselbe Fingerabdruck angesehen werden, obwohl sich die Abdrücke unter Umständen signifikant unterscheiden.

Die hier eingeführten Minutienwolken sollen lediglich als Beispiel für markante Minutienansammlungen dienen. Dies ist das Kriterium für den hier realisierten Ansatz.

Die Minutiendaten (Koordinaten, Minutientyp usw.) eines Fingerabdrucks lassen sich mit Hilfe der Template-Erzeugung der biomiddle extrahieren. Dabei spielt der ausgewählte Matching-Algorithmus noch keine Rolle, da die Extraktion vom gewählten Ausgabe-Format abhängt, in diesem Fall kommt das „Minutia Record Format“ nach ISO Standard 19794-2 zum Einsatz [ISO19794-2]. Die Koordinaten der Minutien werden entsprechend der Bildauflösung im fixen Koordinatensystem pixelbezogen ausgegeben. Demnach sind die Daten leider nicht invariant gegenüber Translation und Rotation, wie die Daten, welche der Bozorth3 Algorithmus errechnet, bzw. verwendet (siehe hierzu 2.6.1). Aus diesem Grund sind die Minutiendaten im „Rohzustand“ nicht für direkte Vergleiche zwischen den Fingerabdrücken verwendbar. Diese müssen erst aufbereitet werden in Bezug auf Translation und Rotation.

Für die erste Betrachtung der Fingerabdruckbilder ist die Tatsache der Invarianz gegenüber Translation und Rotation irrelevant. Es geht in erster Linie nur darum festzustellen, ob Minutienwolken existieren. Hierfür kann gegebenenfalls die bildliche Darstellung herangezogen werden, welche durch die biomiddle erzeugt wird (Funktion „process“). Dies ist ein Bild welches den ursprünglichen Abdruck mit eingezeichneten Minutien darstellt.

Damit ließe sich vorerst optisch klären, ob sich sogenannte Minutienwolken in einem Fingerabdruckbild befinden oder nicht, ergo ob solche Wolken überhaupt existieren. Nun kann man aber auch diese Daten, also die Minutienwolken, nicht für einen aussagekräftigen Vergleich nutzen, weil auch hier gilt, dass sie nicht invariant gegenüber Translation und Rotation sind. Die Daten sind einfach gesagt, zu variabel interpretierbar für einen Vergleich. Des Weiteren ist diese Methode aufgrund der hier verwendeten, großen Datenmenge eher ungeeignet.

Die Vorgehensweise für die man sich entscheidet ist mitunter auch eine Frage des Aufwands bzw. des Rechenaufwands. Bei den Fingerabdruckvergleichen muss jeder Abdruck genau einmal mit jedem anderen verglichen werden. Der Gesamtaufwand ist aber logischerweise nicht nur von der Anzahl der Vergleiche abhängig, sondern auch von dem Aufwand des verwendeten Algorithmus. Aber selbst wenn dieser vorerst außen vor gelassen wird, ist der Aufwand der Vergleichsberechnungen doch sehr hoch, wenn man über eine aussagekräftige Menge an Fingerabdrücken Betrachtungen durchführen möchte.

Geht man davon aus, dass  $n$  die Anzahl der Fingerabdrücke definiert, so erhält man folgende Binomialkoeffizienten:

$$\sum_{k=1}^n k = \frac{(n-1) * n}{2} \leftrightarrow \binom{n}{2}$$

Um eine Aussage über die Häufung von Minutien in einem bestimmten Bereich eines Fingerabdrucks treffen zu können, könnte ein Histogramm verwendet werden, welche die Minutienverteilung grafisch darstellt und bei der Analyse unterstützt. Jedoch gilt auch hierfür, dass die Daten nicht genormt zu einem Mittelpunkt existieren und somit die Aussagekraft an Bedeutung verliert, bzw. diese verfälscht.

Es ist jedoch relativ simpel möglich, die Daten bezüglich der Translation in Hinsicht auf einen Schwerpunkt zu normieren. So könnte man wieder vergleichbare Werte schaffen und diese mittels Histogrammerzeugung auswerten.

Am Ende der Betrachtungen steht auch die Frage, was gewichtiger ist, Minutienwolken oder einzelne Minutien, welche Abseits der Wolken zu finden sind.

### 3.3. Herangehensweise

Eine optische Betrachtung von Fingerabdrücken ist sehr zeitaufwändig. Darüber hinaus ist diese Arbeitsweise nicht in einen vollautomatisierten Prozess umwandelbar. Aus diesem Grund wird diese mögliche Herangehensweise außen vor gelassen.

Demnach muss eine andere Herangehensweise gewählt werden. Ein aus den Minutiendaten herleitbarer Ansatz ist es, die Betrachtungsweise der Minutien zu verändern, aus der man schließlich Erkenntnisse erhalten möchte. So lassen sich die Minutiendaten, nach Beseitigung der Verschiebung/Translation, sozusagen aufeinandergelegt auch von einem Schwerpunkt aus betrachtet kreisförmig nach Außen beleuchten. Bei der Entfernung der Translation werden die Daten unterschiedlicher Finger lediglich neu betrachtet, nicht alle zusammen in einen Datensatz (einer Datentabelle) gespeichert, können also noch individuell betrachtet und verwendet werden. Daraus können Aussagen getroffen werden, ob sich Minutien überwiegend im Zentrum eines Fingerabdrucks, also um den Schwerpunkt herum ansammeln oder eher in äußeren Bereichen.

Diese Herangehensweise hätte unter anderem den Vorteil, dass die Rotation der Fingerabdrücke vernachlässigt werden kann. Allerdings sollte bei den Überlegungen bedacht werden, was mögliche Rotationen der Minutien für Folgen hätten. So kann man Rotationen des Fingerabdrucks auch als verschieben/bewegen der „Minutienringe“ betrachten. Dies würde bedeuten, dass aufgrund der unterschiedlichen Durchmesser bei einer Rotation um den gleichen Winkel die inneren Ringe nicht so stark verschoben werden können wie äußere Ringe, da diese einen kleineren Umfang haben.

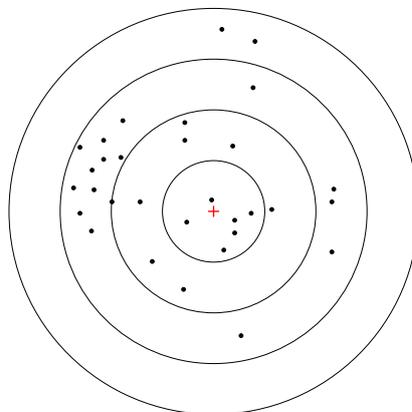


Abbildung 13: Veranschaulichung einer möglichen Verteilung von Minutien

Geht man davon aus, dass sich gerade im Zentrum besonders viele Minutiendaten häufen, also dort im Allgemeinen mehr Minutien zu finden sind als in den äußeren Bereichen, so ist dort unter Umständen auch ein erhöhtes Potential für Falscherkennung eines Abdrucks.

---

### 3.4. Zwischenfazit

Wie bereits festgehalten, ist eine Lösung mit optischer Betrachtung zwar theoretisch möglich, aber aufwändig und basierend auf den zuvor genannten Gründen, eher unrealistisch. Außerdem ist ein Ergebnis nur aussagekräftig, wenn man über entsprechende Datenmengen evaluiert, was diese Methode noch erschweren würde.

Die rein rechnerische Methode ist aus logischer Sicht auf jeden Fall präziser und auf nahezu jeder Datenmenge möglich.

---

## 4. Entwurf

---

In diesem Kapitel tragen die Überlegungen und Erkenntnisse aus dem vorangegangenen Kapitel zu einem ersten Entwurf bei. Dieser Entwurf beinhaltet erste umsetzbare Betrachtungen, welche dann in Kapitel 5 zur tatsächlichen Realisierung führen.

In der verwendeten Fingerabdruckdatenbank befinden sich zu jedem Fingerabdruck 8 Bilder, wobei jedes der Bilder unterschiedlichen Aufnahmen entspricht und die Qualität der Bilder sehr variieren kann. So liefert nicht jedes Bild gleich viele Minutien und stimmt dementsprechend auch nicht gleichermaßen gut mit andern Fingerabdrücken überein. Da aber für die Betrachtungen der Aufwand viel zu hoch wäre alle Bilddateien zu verwenden, soll ein sogenannter „Referenzabdruck“ gewählt werden, der als Repräsentant für den Finger stehen soll. Um diesen zu ermitteln, wird ein Verfahren mittels FMR-Berechnung in Verbindung mit Mittelwert- bzw. Medianwerterzeugung verwendet.

Nach Bestimmung eines Repräsentanten für jeden Finger werden mittels biomiddle, bzw. BioAPI die Minutiendaten (Minutienkoordinaten, Typ und Winkel) ermittelt. Wie in Kapitel 2 bereits beschrieben müssen Translation und Rotation beachtet werden, wobei die Rotation durch die Betrachtungsmethode mit Hilfe von „Minutienringen“ sozusagen vernachlässigt werden kann. Die Translation jedoch nicht, dies kann nahezu herausgerechnet werden indem man den Schwerpunkt der Minutiendaten jedes Fingers errechnet und dessen Differenz zum Mittelpunkt des Koordinatensystems ermittelt und diese dann von allen Minutienkoordinaten abzieht und somit eine Verschiebung des gesamten Abdrucks vollzieht. Die Schwerpunktermittlung geschieht mit Hilfe der Median-Berechnung aller Minutien eines Abdrucks. Hierfür werden die Median-Werte der x-Koordinaten und der y-Koordinaten gesondert errechnet und ein fiktiver Punkt errechnet, denn diese Minutie wird es in den meisten Fällen nicht geben. Dies ist aber irrelevant da es um den Schwerpunkt aller Minutien geht und nicht einen tatsächlichen markanten Punkt des Fingerabdrucks. Bei der Überlegung zur Berechnung des Schwerpunktes könnte man auch den arithmetischen Mittelwert verwenden, jedoch können dabei einzelne Minutien, die sehr weit außerhalb im Koordinatensystem liegen, also wie Ausreißer gewichtet sind den Schwerpunkt so stark verfälschen, dass dieser zwar dem Mittelpunkt aller Minutien entspricht aber nicht als Schwerpunkt im eigentlichen Sinne fungiert. Durch Verwendung des Median verlieren einzelne Ausreißer an Bedeutung, was insbesondere dann wichtig ist, wenn diese Ausreißer unter Umständen keine richtigen Minutien sind, sondern lediglich vom Erkennungsalgorithmus als solche erkannt wurden. Außerdem kann es auch vorkommen, dass auf einem Fingerabdruckbild nicht nur die eigentliche Fingerbeere zu sehen ist, sondern noch weitere Teile des Fingers, also z.B. das zweite Fingerglied. Diese Teile gehören nicht zum eigentlichen Fingerabdruck, können aber solche Minutien-Ausreißer erzeugen. Genau aus diesen Gründen wird hier der Median verwendet.

Den Medianwert einer Menge von Werten erhält man, indem alle Werte der Größe nach sortiert werden an mittlerer Position der Sortierung. Ist die Anzahl der Werte gerade, so werden die beiden Werte in der Mitte addiert und durch zwei dividiert. Zusammengefasst als folgende Formel mit  $\bar{x}$  als Median und  $n$  Werten  $(x_1, x_2, x_3, \dots, x_n)$ :

$$\bar{x}_{med} = \begin{cases} \frac{x_{\frac{n+1}{2}}}{2} & n \text{ ungerade} \\ \frac{1}{2} \left( x_{\frac{n}{2}} + x_{\frac{n}{2}+1} \right) & n \text{ gerade.} \end{cases}$$

Im Gegensatz dazu wird der Mittelwert berechnet, indem alle Werte addiert und durch die Anzahl der Werte dividiert werden. Hierbei ist wie gesagt jeder Wert gleichgewichtig.

$$\bar{x}_{arithm} = \frac{1}{n} \sum_{i=1}^n x_i$$

---

Nach Erhalt der Daten in einem zur Translation nahezu normierten Zustand, lassen sich die Minutien der verschiedenen Abdrücke miteinander vergleichen. Hierzu kann ein Histogramm erzeugt werden, welches die Abstände der Minutien zum Mittelpunkt grafisch darstellt und einen Vergleich anhand dessen einfacher macht.

Zusammengefasst kann man den oben genannten Prozess in folgende Arbeitsschritte unterteilen:

1. Matching über die 8 Bilder eines Fingerabdrucks um den Abdruck mit den besten Werten für weitere Vergleiche zu ermitteln (Median, Repräsentant)
2. Minutiendaten aus den erhaltenen „Referenzabdrücken“ extrahieren und die enthaltene Translation herausrechnen und „beseitigen“ bzw. die Abdrücke normieren mittels Schwerpunkt-berechnung (Median)
3. Daten in Ringen (ähnlich zu Radarkreisen) um den Mittelpunkt des Koordinatensystems betrachten – Histogramm erzeugen mit Abständen der Minutien zum Mittelpunkt

---

## 5. Realisierung und Auswertung

---

In Kapitel 2 wurden sämtliche verwendeten Mittel und dazu benötigte Grundkenntnisse erörtert. Wie dort bereits erwähnt wird mittels biomiddle von secunet auf die einzelnen Elemente des BioAPI zugegriffen, bzw. deren Funktionen bereitgestellt und in ausimplementierter Fassung verwendbar gemacht. Das BioAPI Framework ist sehr umfangreich und bietet eine gute Grundlage zur Erstellung biometrischer Software.

Diese gewonnen Grundkenntnisse und die in Kapitel 3 und 4 entwickelten Ansätze und Überlegungen bilden eine gute Basis für die Entwicklung der gewünschten Anwendung.

In diesem Kapitel werden die verwendeten Implementierungswerkzeuge für die Hilfsanwendung, sowie Form der zu untersuchenden Daten erläutert. Im Anschluss werden die Umsetzung der Anwendung und die Auswertung der erhaltenen Daten diskutiert.

Die hier entwickelte Anwendung dient lediglich als Hilfswerkzeug für die statistischen Beobachtungen. Sie liefert ausschließlich die Daten, aus denen letztendlich Erkenntnisse abgeleitet werden.

### 5.1. Projektumfeld

Die Programmiersprache C# bietet hervorragende Eigenschaften um eine Anwendung mit grafischer Oberfläche relativ einfach zu erstellen. Es ist möglich mit dieser Sprache sowohl .NET-Anwendungen zu entwickeln, als auch Anwendungen für Microsoft Windows. Für dieses Projekt fiel die Wahl auf eine Windows-Anwendung welche mit Hilfe von Microsoft Visual Studio 2010 in C# erstellt wurde. Die Implementierung des Hilfswerkzeugs wurde auf einem Computer mit Microsoft Windows 7 (64-Bit) durchgeführt. Für die Sicherung und Aufbereitung der Daten wurde mit SQLite Datenbanken gearbeitet.

Wie bereits in Kapitel 2 umfangreich beschrieben, benötigt die hier realisierte Implementierung das BioAPI Framework (Version 2.1.9.0) und die Middleware biomiddle (Version 2.1.14.0) von secunet. Der verwendete Bozorth3 Algorithmus hat die Version 2.1.2.0.

Die FVC [FVC] bietet die Fingerabdruckbilder im TIFF-Format an, welche dann für die Nutzung mit der biomiddle in das BMP-Format umgewandelt wurden. Die biomiddle, bzw. das BioAPI kann nur bestimmte Bildformate verarbeiten. Das TIFF-Format ist in den verwendbaren Datenformaten enthalten, jedoch ist das BMP-Format universeller einsetzbar. Um die Erweiterbarkeit der Anwendung zu gewährleisten, gerade in Bezug auf grafische Darstellung oder Verwendung von externen Grafikprogrammen, wurde das BMP-Format hier gewählt. Die Formatumwandlung ist dabei verlustfrei im Gegensatz zum beispielsweise verlustbehafteten JPEG-Format.

Der Ansatz mit grafischer Ausgabe wurde verworfen, da die Daten im Anfangsstadium der Berechnungen nicht zwingend ausgegeben werden müssen (als eingezeichnete Minutien in den Bildern in etwa). Aussagekräftig hingegen sind die Daten im Vergleich zu allen anderen der Betrachtung und im Zusammenhang bzgl. eines Koordinatensystems, wie zuvor im Entwurf beschrieben.

Zu den Bilddateien sei noch gesagt, dass sie immer eine spezielle Namenskodierung haben, welche hier teilweise direkt in den Code der Hilfsanwendung eingeflossen ist. Dies stellt zwar einen Verlust der Generalität des Hilfsprogramms dar, ist aber einfacher zu realisieren und wurde aus Gründen der Zeitersparnis so durchgeführt. Eine universellere Implementierung ist jedoch teilweise möglich (siehe dazu Kapitel 6). Die Namenskodierung beinhaltet Finger Nummer und Bildindizes mit Unterstrich getrennt, also z.B. 1\_1.bmp bis 1\_8.bmp sind alle acht Bilder zum Finger 1.

Für die Auswertung der Daten wird Microsoft Excel als externe Hilfsanwendung verwendet, da sich damit sehr schnell und einfach aus Datenmengen entsprechend aussagekräftige Diagramme erzeugen lassen.

## 5.2. Software-Entwurf und –Realisierung

Der Hilfsanwendung zum Erzeugen der auszuwertenden Daten genügt eine simpel gehaltene Benutzungsoberfläche, welche dem Benutzer mindestens folgende Funktionen erlaubt:

- Angabe des Pfades zu den Bilddateien. Um den Prozess einfach und schnell zu gestalten, wird hier angenommen, dass sich alle zu betrachtenden Bilddateien in einem Ordner befinden, also alle Bilder den gleichen Grundpfad haben.
- Auswahl des Matching-Algorithmus aus allen im biomiddle-Server geladenen BSPs anzeigen.
- Möglichkeit aus den vorhandenen Fingern (nicht Fingerabdruckbildern an sich) wählen zu können, welche Finger untersucht und deren Daten miteinander verglichen werden sollen.
- Steuerung der Funktionen, welche durch den Anwender (respektive Verfasser dieser Arbeit) mittels Aktivitäts-Schaltflächen ausgeführt werden sollen.

Das Design der Anwendung kann deswegen recht einfach gehalten werden, weil sie, wie bereits erwähnt, nicht als Produkt verwendet wird. In der Oberfläche sind einige Elemente nicht ausimplementiert. Diese wurden integriert, um unter Umständen eine Erweiterung der Anwendung zu erlauben. Jedoch wurde die Auswertung der Daten schlussendlich doch extern mit Microsoft Excel realisiert, anstatt wie ursprünglich vorgesehen mittels Integration einer Mathematik-Anwendung in die hier entwickelte Anwendung. Die Erweiterung um diese Funktionalität ist prinzipiell möglich aber nicht Gegenstand dieser Arbeit.

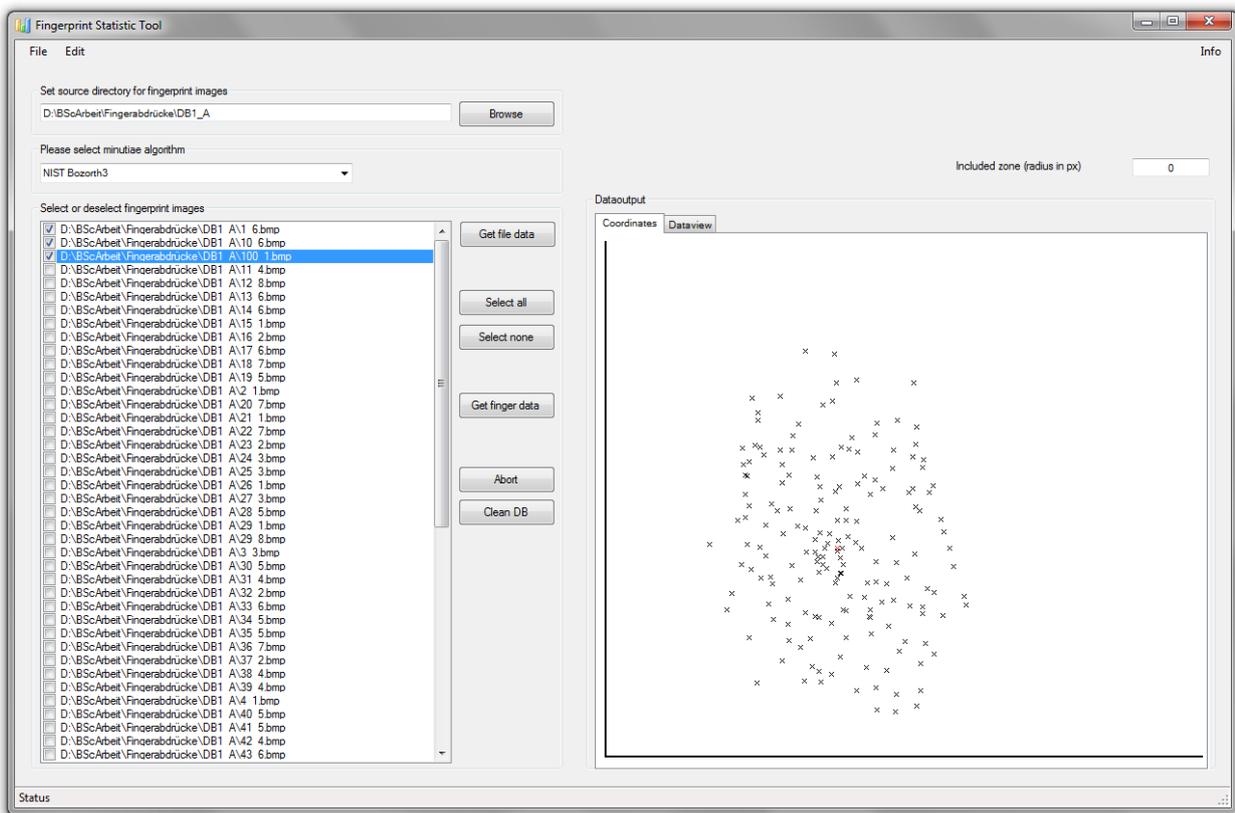


Abbildung 14: Anwendungsoberfläche des Fingerprint Statistic Tools

Im Folgenden werden die einzelnen Teile der Realisierung diskutiert und die dafür implementierten Klassen umschrieben. Aufgrund der Nutzung der implementierten Anwendung, welche sich als Tool für diese Arbeit beschränkt, werden hier lediglich die Hauptklassen kurz umschrieben. Die Implementierung könnte noch ausgebaut und verfeinert werden.

### 5.2.1. Benutzungsoberfläche

Die Klasse `FingerprintStatisticTool` (Abbildung 15) beinhaltet die Funktionen für die Benutzungsoberfläche. Die Benutzungsoberfläche selbst ist darin nicht direkt implementiert. Sie wird „automatisch“

mit Hilfe von Visual Studio erzeugt und kann über diese Klasse spezifiziert werden. Die Funktionen der Klasse beziehen sich im Großen und Ganzen auf die Funktionalitäten der Schaltflächen und Inhalte der Textboxen etc.

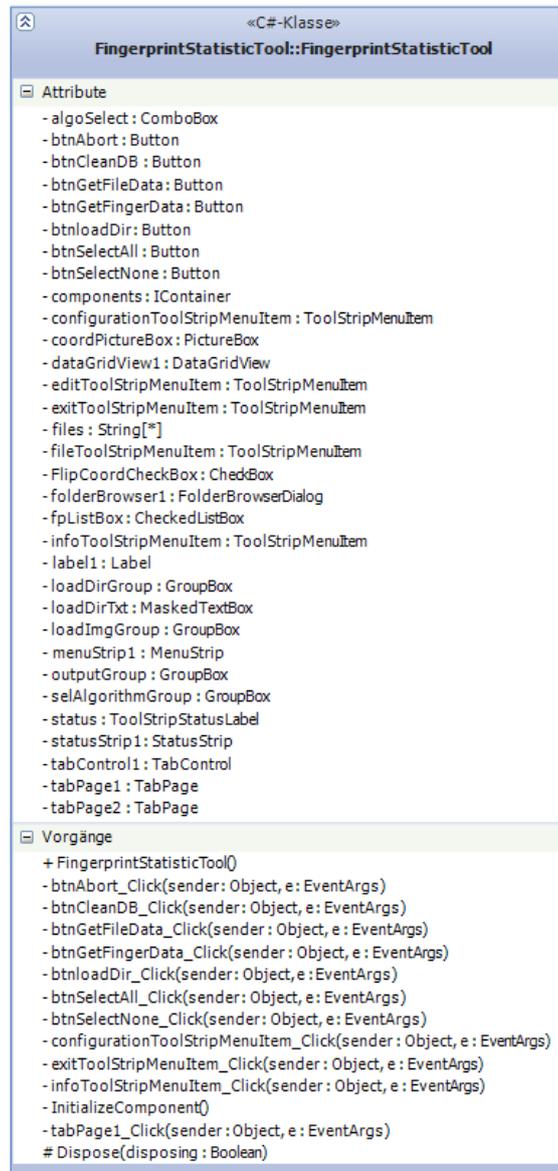


Abbildung 15: Klasse <FingerprintStatisticTool>

### 5.2.2. Prematching

Zu Beginn des sogenannten Prematching werden alle Fingerabdruckbilder im BMP-Format eines angegebenen Ordners eingelesen. Für das Prematching wird jedes Bild eines Fingers mit jedem anderen Bild des gleichen Fingers verglichen, aber auch in die umgekehrte Richtung, da dies nicht zwingend das gleiche Ergebnis liefert. Die Matchings werden dabei mit Hilfe des Bozorth3 Algorithmus durchgeführt (bedeutet Bild 1\_1 von Finger 1 wird mit 1\_2, 1\_3, 1\_4... 1\_8 verglichen, ebenso aber auch 1\_2 mit 1\_1, 1\_3, 1\_4... 1\_8 usw.) -> ergibt pro Finger  $8 \cdot 7 = 56$  Werte.

Der Begriff FMR-Wert wird in den folgenden Kapiteln für die Werte verwendet, welche hier errechnet werden. Dies geschieht mittels der Funktion „VerifyMatchFMR“ der biomiddle, wobei diese intern eine Matchscore für jeden Vergleich errechnet und anhand dessen den FMR-Wert für einen einzelnen Vergleich erhält. Nur so ist die Zuordnung eines FMR-Wertes zu einem Vergleich überhaupt möglich. Für die Funktion wird als Eingabewert zwar auch die Angabe eines Schwellwerts („MaxFMRRequested“) benötigt, jedoch dient dieser zur Einteilung in Matches oder Non-Matches der Vergleiche. Da dies aber hier unerheblich ist, ist auch dessen Wert irrelevant und kann auf irgendeinen Wert gesetzt werden. In

der Implementierung wurde er auf den Wert 0.001 , also 0.1% festgelegt. Alle Werte unterhalb dieses Schwellwerts ergeben einen Non-Match und alle oberhalb einen Match.

Um mit Hilfe der 56 FMR-Werte ein Bild auswählen zu können, wird zunächst der Medianwert der FMR-Werte innerhalb eines Fingerbildes errechnet, d.h. der Median aller Vergleichswerte von 1\_1 zu allen anderen Bildern des Fingers, analog mit den anderen Bildern. Medianwert wird dem Mittelwert vorgezogen, da ansonsten Non-Matchings und Ausreißer-Matching-Werte zu stark ins Gesamtgewicht der Werte fallen. Man erhält danach 8 Werte pro Finger statt den 56 vorigen Werten. Von diesen acht Werten wird das Minimum gewählt, da der FMR-Wert möglichst klein gehalten werden sollte, man möchte so wenig wie möglich Falsche Matchings. Somit wird ein Repräsentant gewählt pro Finger.

Das soeben umschriebene Verfahren fand auch in dem „Life-Finger 2“ Projekt in ähnlicher Weise Anwendung. Jedoch wurde damals die Mittelwertberechnung verwendet, was jedoch unter Umständen die erwähnten Ausreißer, bzw. Non-Matchings genauso wertet wie die Matchings. Aus diesem Grund, falls diese Non-Matchings überwiegen, ist dies eher ungünstig und daher wurde hier der Median verwendet.

Die Daten des Prematching werden folgendermaßen in SQL-Datenbanktabellen abgelegt: Für das Prematching werden zwei Tabellen erzeugt und mit Daten gefüllt, in der einen stehen sämtliche Vergleiche zwischen zwei Bildern, immer innerhalb eines Fingers, d.h. es gibt beispielsweise keinen Vergleich 1\_1 ↔ 2\_1. Aber wie zuvor beschrieben, gibt es Vergleiche in beide Richtungen innerhalb eines Fingers, also existiert sowohl ein Eintrag von 1\_1 → 1\_2, als auch umgekehrt.

id	finger1	finger2	match	fmr
1	1_1	1_2	match	4,869979E-05
2	1_1	1_3	match	1,327382E-06
3	1_1	1_4	match	0,000584098
4	1_1	1_5	no match	0,009264437
5	1_1	1_6	match	1,307634E-06
6	1_1	1_7	no match	0,00106273
7	1_1	1_8	no match	0,208065
8	1_2	1_1	match	4,869979E-05
9	1_2	1_3	match	1,34572E-06
		.	.	
		.	.	

Abbildung 16: Auszug der PreMatch Datenbanktabelle

In der zweiten Tabelle gibt es für jedes Fingerabdruckbild einen Eintrag mit dem errechneten Median. Des Weiteren gibt es eine Spalte mit einer Markierung des Minimums jedes Fingers innerhalb dieser acht Werte, womit der Repräsentant des Fingers nach Ermittlung gekennzeichnet wird. In Abbildung 17 sieht man einen Auszug aus der dazugehörigen Datenbanktabelle. Hier ist auch die Mittelwertberechnung integriert, um den Unterschied zu verdeutlichen. Repräsentanten sind aber nur die Fingerabdruckdateien, welche hier in der Spalte selectMED markiert wurden.

finger	fingerID	imageID	avqFMR	selectAVG	medFMR	selectMED
11_1	11	1	0,04470933961042857		1,347131E-06	
11_2	11	2	0,0093832090527142843		1,347131E-06	
11_3	11	3	0,058524362285714289		0,006466993	
11_4	11	4	0,013934753454		1,285064E-06	o
11_5	11	5	0,0016394018194285712	x	1,317508E-06	
11_6	11	6	0,011995374764571427		1,314687E-06	
11_7	11	7	0,081505445857142853		0,04198069	
11_8	11	8	0,048357211277571428		0,0001960508	

Abbildung 17: Beispiel für die Berechnungen des Medians und der Auswahl eines Repräsentanten

Die implementierten Klassen für den gesamten Prematching-Prozess sind die Klassen PreMatching und DatabaseManager:

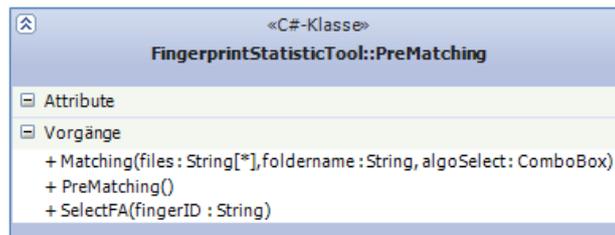


Abbildung 18: Klasse <PreMatching>

Die Klasse PreMatching beinhaltet den kompletten Prozess des Prematchings, wie zuvor beschrieben. In der Funktion „Matching“ werden immer zwei Fingerabdrücke miteinander verglichen. Dabei erhält die Klasse die komplette Liste aller Finger als Übergabeparameter inklusive dem Ordner in dem sie sich befinden. Das Prematching wird immer auf alle Bilder im Ordner angewandt, um das recht aufwändige Auslesen der Daten ohne weiteren Auswahlsschritt durch den Benutzer automatisieren zu können. Die Funktion „Matching“ enthält ebenso die Implementierung, die dafür sorgt bei dem Prematching nicht aus Versehen ein und dasselbe Bild mit sich selbst zu vergleichen. Die Vergleiche an sich werden mittels Aufrufe der biomiddle, bzw. im Detail der Funktionen „VerifyMatchFMR“, „CreateTemplate“ und „Process“ aus dieser realisiert. Wie diese Funktionen arbeiten, kann der biomiddle Dokumentation entnommen werden [SEC09]. Nach jedem Vergleich, wird dessen Ergebnis, wie in Abbildung 16 zu sehen per Aufruf der DatabaseManager-Klasse aus der Funktion „Matching“ heraus in die Datenbank geschrieben. Dies geschieht mit Hilfe der Funktion „PreMatching“, die gegebenenfalls die Tabelle erzeugt und dann die Werte hineinschreibt, sollten diese noch nicht existieren.

Die Finger-Repräsentanten werden mit Hilfe der Funktion „SelectFA“ aus der Klasse PreMatching, welche wiederum die Funktionen „AvgFA“ und „MarkSelectedFA“ aus dem DatabaseManager aufruft, ermittelt. Hierbei errechnet „AvgFA“ die Medianwerte aus allen Vergleichen eines Fingerabdruckbildes mit allen anderen des gleichen Fingers und speichert diese in der Datentabelle „PreMatchAverage“ (Abbildung 17) in die Spalte medFMR ab. Es wurden hierbei auch die Mittelwerte zum Vergleich berechnet und in die Spalte avgFMR abgelegt. Der Aufruf von „MarkSelectedFA“ bewirkt die Kalkulation des Minimums gerade aus diesen Medianwerten und markiert dann entsprechenden Repräsentanten mit einem o in der Spalte selectMED.

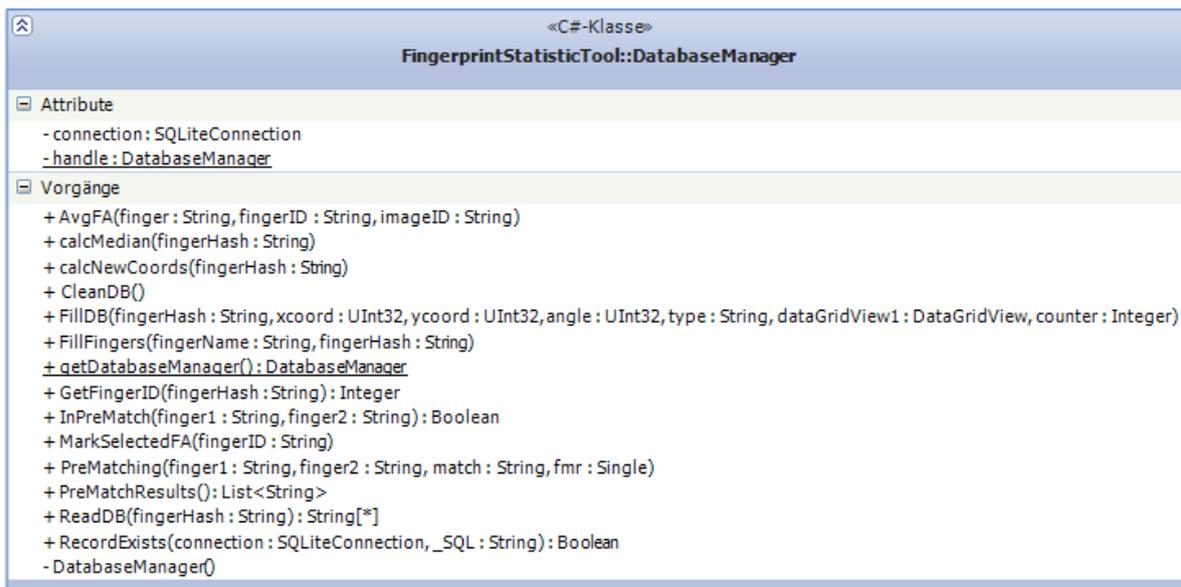


Abbildung 19: Klasse <DatabaseManager>

Der Prozess des Prematchings ist damit beendet und das Auslesen der benötigten Minutiendaten aus den Repräsentanten kann daran anknüpfen.

Die Funktion „InPreMatch“ der Klasse DatabaseManager dient lediglich dazu festzustellen, ob ein Eintrag bereits existiert, um diesen nicht nochmals zu berechnen. Dies bewirkt bessere Performanz.

### 5.2.3. Minutiendaten auslesen

Da nach dem Prematching jeder Finger durch genau ein Bild repräsentiert wird, können nun die Minutiendaten der Finger extrahiert werden. Ein direkter Vergleich via Bozorth3 ist hier nicht wünschenswert, weil die Minutien an sich betrachtet werden sollen und deren Häufung in Gebieten des Koordinatensystems. Beim Bozorth3 ist ein direkter Zugriff auf die Minutiendaten nicht möglich, da diese nicht als Koordinaten sondern als Punkte mit Abhängigkeit zu den anderen Punkten gespeichert werden, ein recht aufwändiges Verfahren, welches in Kapitel 2.6.1 genau beschrieben wurde.

Die Repräsentanten werden mittels Aufruf der Funktion „PreMatchResults“ des DatabaseManagers direkt an das Auswahlfeld in der Benutzungsoberfläche weitergegeben, wo dann die zu untersuchenden Finger ausgewählt werden können.

Die Extraktion der Minutiendaten, welche die Koordinaten, Typ, Winkel usw. je Minutie angeben, geschieht mittels der Template-Erzeugung des BioAPI (via biomiddle). Als Ausgabeformat wird das bereits genannte „Minutia Record Format“ des ISO Standard 19794-2 [ISO19794-2] verwendet. Die binären Daten des Formats werden dann für die benötigten Informationen bitweise ausgelesen. In Abbildung 20 ist ein Auszug aus dem Standard, der genau die Bytes umfasst, welche die Minutiendaten umfasst. Für die Verwendung in dieser Arbeit ist lediglich ein Teil davon von Bedeutung. Genauere Definitionen können der Dokumentation des ISO Standards entnommen 19794-2 [ISO19794-2] werden.

	Field	Size	Valid Values	Notes		
One per Record	Format Identifier	4 bytes	0x464D5200 (‘F’ ‘M’ ‘R’ 0x0)	‘FMR’ – finger minutiae record		
	Version of this standard	4 bytes	n n n 0x0	‘XX’, with XX = 20 or greater		
	Length of total record in bytes	4 bytes	24 - 4294967295	either 0x0018 to 0x0000FFFFFFFF		
	Capture Equipment Certification	4 bits		compliance with Annex B or future ISO standards		
	Capture Device Type ID	12 bits		Vendor specified		
	Image Size in X	2 bytes		in pixels		
	Image Size in Y	2 bytes		in pixels		
	X (horizontal) Resolution	2 bytes		in pixels per cm		
	Y (vertical) Resolution	2 bytes		in pixels per cm		
	Number of Finger Views	1 byte	0 to 255			
Reserved byte	1 byte	00	0 for this version of the standard (reserved for future use)			
One per View	Finger Position	1 byte	0 to 10	See Table 2		
	View Number	4 bits	0 to 15			
	Impression Type	4 bits	0 to 3 or 8	See Table 3		
	Finger Quality	1 byte	0 to 100	0 to 100		
	Number of Minutiae	1 byte				
One per minutiae	X (minutia type in upper 2 bits)	2 byte		Expressed in image pixels		
	Y (upper 2 bits reserved)	2 byte		Expressed in image pixels		
	$\theta$	1 byte	0 to 255	Resolution is 1.40625 degrees		
	Quality	1 byte	0 to 100	1 to 100 (0 indicates ‘quality not reported’)		
One per per view	Extended Data Block Length	2 bytes		0x0000 = no private area		
0+ per view	Extended Data Area Type Code	2 bytes		only present if Extended Data Block Length $\neq$ 0		
	Extended Data Area Length	2 bytes		only present if Extended Data Block Length $\neq$ 0		
	Extended Data	In prev. field		only present if Extended Data Block Length $\neq$ 0		
<b>Each extended data area may contain vendor-specific data, or one or more of the following (in any order):</b>						
0+ per view	Ridge count	Ridge count extraction method	1 byte	0 to 2		
		Ridge count data – idx #1	1 byte	1 to # of minutiae		
		Ridge count data – idx #2	1 byte	1 to # of minutiae		
		Ridge count data – count	1 byte			
		additional ridge counts...				
	Core and delta data	Number of cores	4 bits	0 to 15		
		Core information type	2 bits	0 to 1		
		X location	14 bits			
		Y location	2 bytes			
		Angle (if core info type $\neq$ 0)	1 byte	0 to 255		
		Number of deltas	4 bits	0 to 15		
		Delta information type	2 bits	0 to 1		
		X location	14 bits			
		Y location	2 bytes			
	Angles (if delta info type $\neq$ 0)	3 bytes	0 to 255			
	0+ per view	Zone qual	Cell Width	1 byte	1 to 255	
			Cell Height	1 byte	1 to 255	
			Cell Information Bit Depth	1 byte	1 to 255	
			Cell Quality Data	CellDataLen		

Abbildung 20: Minutia Record Format des ISO Standard 19794-2 [ISO19794-2, S. 21]

Es werden die Koordinaten, der Typ und der Winkel der Minutie in einer SQL-Datenbanktabelle gesichert. Dies geschieht für jeden gewählten Finger, welcher über die Benutzungsoberfläche der Anwendung ausgewählt werden kann.

Für das Auslesen der Minutiendaten aus den Fingerabdruckbildern ist die Klasse GetFPData (Abbildung 21) in der Implementierung zuständig. Oder genauer, deren Funktion „GetData“, welche als Eingabewert unter anderem die Liste mit in der Benutzungsoberfläche ausgewählten Fingerabdrücken erhält. Diese Funktion verwendet ebenfalls „CreateTemplate“ aus der biomiddle. Aus dem Template, welches im Minutia Record Format erzeugt wird, kann dann bitweise entsprechende Information entnommen werden (siehe hierzu obige Abbildung 20). Die erhaltenen Daten werden wieder mit Hilfe des DatabaseManagers in eine erzeugte Datenbanktabelle gespeichert.



Abbildung 21: Klasse <GetFPData>

Es werden für das Auslesen der Minutiendaten mehrere Datenbank-Tabellen erzeugt. Eine Tabelle (\_Fingers, Abbildung 22), welche alle gewählten Fingerabdruckdateien mit Pfad enthält und ihnen IDs (Spalte fingerID) zuordnet. Dort werden auch der Hashwert der Datei (Spalte filehash) und später die Median-Werte der Minutien-Koordinaten (Spalten xMedian und yMedian) abgelegt.

fingerID	filename	filehash	xMedian	yMedian
1	D:\BScArbeit\Fingerabdrücke\DB1_A\1_6.bmp	E219C6701A4BE122C0300EEFC9BA05	319	230
2	D:\BScArbeit\Fingerabdrücke\DB1_A\10_6.bmp	6E1941A0FAC77D18F1FABAF6D02A15	310	225
3	D:\BScArbeit\Fingerabdrücke\DB1_A\100_1.bmp	B0D85F5A30FDDC89F048F6B8EACCCE	265	290

Abbildung 22: Datenbanktabelle mit allen gewählten Repräsentanten

Es sei noch zu erwähnen, dass der hier abgespeicherte Hashwert zur Identifizierung der Dateien dient. Die Identifizierung sollte nicht nur allein auf der ID ruhen, da eine Umbenennung der Dateien jederzeit möglich ist, und zu fehlerhaften Ergebnissen führen kann. Die Hashwerte werden während der Betrachtungen immer wieder mit abgefragt um korrekte Berechnungen sicher zu stellen.

Die komplette Tabelle der in Abbildung 22 angedeuteten Daten kann der Tabelle 1 aus Anhang A entnommen werden.

Es wird im Ausleseprozess für jeden Finger eine Tabelle erzeugt (z.B. finger\_1, Abbildung 23). Die Nummerierung im Namen wird dabei anhand der fingerID aus der obigen Tabelle gewählt. Die ID-Vergabe funktioniert automatisch und in Einlesereihenfolge der Dateien. So gibt es für die Datei 1\_6.bmp, welche nun in diesem Fall auch den Finger mit der fingerID 1 darstellt, eine Tabelle namens finger\_1. Dort werden dann alle zuvor erwähnten Minutiendaten abgelegt. Wobei für spätere Berechnungen noch Spalten für verschobene Koordinaten und die Distanz zum Mittelpunkt des Koordinatensystems angelegt werden. Hierzu mehr in den folgenden Teilkapiteln.

Für das eintragen der Daten in die Datenbank sind die Funktionen „FillDB“ (einzelne Fingertabellen für jeden Finger) und „FillFingers“ (Gesamttabelle wie in Abbildung 23) des DatabaseManagers zuständig.

minutiaeID	xcoord	ycoord	angle	type	newXcoord	newYcoord	distCenter
1	240	261	199	ridge bifurcation	161	351	84,864598037108507
2	251	184	223	ridge bifurcation	172	274	82,09750300709517
3	230	252	199	ridge bifurcation	151	342	91,678787077491378
4	224	198	223	ridge bifurcation	145	288	100,24470060806208
5	263	215	215	ridge bifurcation	184	305	57,974132162542979
6	347	155	151	ridge bifurcation	268	245	80,05623023850174
7	319	201	15	ridge bifurcation	240	291	29
8	339	202	159	ridge bifurcation	260	292	34,409301068170507
9	366	168	23	ridge bifurcation	287	258	77,801028270839709
10	325	294	7	ridge bifurcation	246	384	64,280634719952786

Abbildung 23: Beispiel für Finger-Datenbanktabelle

### 5.2.4. Minutiendaten aufbereiten

Nun befinden sich in der Datenbank sozusagen alle „Grundzutaten“ für die eigentlichen Betrachtungen. Allerdings sind die Minutiendaten, insbesondere die Koordinaten, in diesem Rohzustand bzgl. Translation nicht invariant. Daher muss sozusagen die Translation bzw. die Verschiebung der Fingerabdrücke zueinander herausgerechnet und eine Normierung durchgeführt werden.

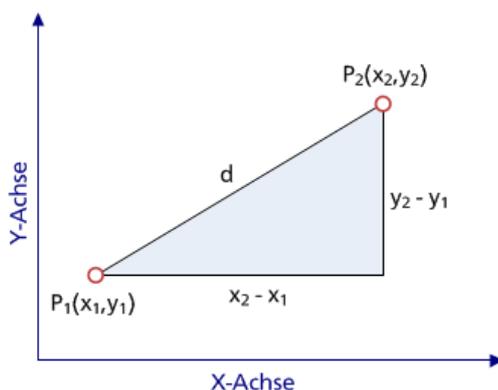
Als neuer Bezugspunkt für alle Abdrücke wird der Mittelpunkt des Koordinatensystems gewählt. Da das Koordinatensystem gerade so groß ist wie die Bildauflösung der Fingerabdruckbilder, welche 480x640 (BxH) entspricht, befindet sich der Mittelpunkt beim Punkt (240, 320).

Von jedem Finger wird nun der Medianwert der Koordinaten aller Minutien bestimmt, jeweils für die x-Achse und die y-Achse. Diese Medianwerte werden in die Datenbanktabelle mit allen Fingern zum jeweiligen Fingerbild gesichert (siehe Abbildung 22). Dies geschieht mittels der Funktion „calcMedian“ innerhalb des DatabaseManagers, weil dort der Eintrag in die Datenbank direkt nach Berechnung vorgenommen werden kann.

Ausgehend von diesen Medianwerten, werden dann die aktualisierten Minutien-Koordinaten errechnet und in die einzelnen Datenbanktabellen der Finger geschrieben (siehe hierzu newXcoord und newYcoord in Abbildung 23). Dies geschieht sozusagen mittels Parallelverschiebung der Daten. Dabei werden die Koordinaten folgendermaßen verschoben:

$$\begin{aligned} \text{newXcoord} &= \text{xcoord} + 240 - \text{xMedian} \\ \text{newYcoord} &= \text{ycoord} + 320 - \text{yMedian} \end{aligned}$$

Die Berechnungen hierfür finden ebenfalls direkt im DatabaseManager statt, da auch hier die Daten erst einmal für weitere Auswertungen gesichert werden sollen. Hierfür ist die Funktion „calcNewCoords“ zuständig, die genau obige Formeln umsetzt. Auf Grundlage dieser neuen Koordinaten wird der Abstand jeder Koordinate zum Mittelpunkte berechnet. Dies geschieht mit Hilfe des Satzes von Pythagoras:



Distanz zwischen  $P_1$  und  $P_2$ :

$$d^2 = (x_2 - x_1)^2 + (y_2 - y_1)^2$$

⇓

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \Leftrightarrow d = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$

---

Da nun die Abstände zwischen dem Mittelpunkt und jeder Minutie bekannt sind, können anhand dessen Abstands-Intervalle betrachtet werden. So entstehen die Ringe, welche in Kapitel 3.3 erwähnt wurden. Betrachtet man also alle Minutien mehrere Fingerabdrücke, welche einen Abstand von  $\leq 10$  Pixeln zum Mittelpunkt haben, so ist dies ringförmig um den Mittelpunkt herum zu verstehen.

Da dies für die Realisierung zwecks des Datenbankabrufs auf die gleiche Tabelle sinniger war, wurden diese Berechnungen ebenfalls in der Funktion „calcNewCoords“ implementiert und Tabellen speziell für die Sicherung der Anzahl von Minutien in bestimmten Bereichen erzeugt. Diese Datenbanktabellen wurden dann direkt in eine Microsoft Excel Tabelle exportiert, was bei geschickter Formatierung im Vorfeld, also in der Erzeugung der Tabelle, kaum Nacharbeit in Excel für die Erzeugung von Histogrammen verlangt.

### **5.3. Ergebnisse der Datenerhebung**

Die Berechnungen wurden über 70 Finger, also auch 70 Abdruckbilder nach dem Prematching-Prozess, durchgeführt. Im Folgenden werden die Werte dieser Abdrücke, insbesondere auf die Verteilung der Minutien bezüglich des Mittelpunktes, aufgelistet und in Histogrammen dargestellt.

Die Betrachtungen der Minutiendaten wurden in drei verschiedenen genauen Varianten durchgeführt. Hier werden lediglich die Diagramme und Daten zu den Untersuchungen aufgelistet. Die daraus gewonnen Erkenntnisse und dessen Folgen für die Thematik der Arbeit werden ausführlich in Kapitel 6 erörtert.

#### **5.3.1. Minutiendaten 10er-Pixel-Intervalle**

In der ersten Variante wurde die Verteilung der Minutien in 10-Pixel-Ringen betrachtet. Dies bedeutet, dass jeder Ring zehn Pixeln entspricht. Die Werte können Tabelle 2 bis Tabelle 6 aus dem Anhang B entnommen werden.

Wie dem folgenden Diagramm (Abbildung 24) zu entnehmen ist diese Betrachtungsweise sehr fein und unter Umständen zu fein für eine sinnvolle Betrachtung.

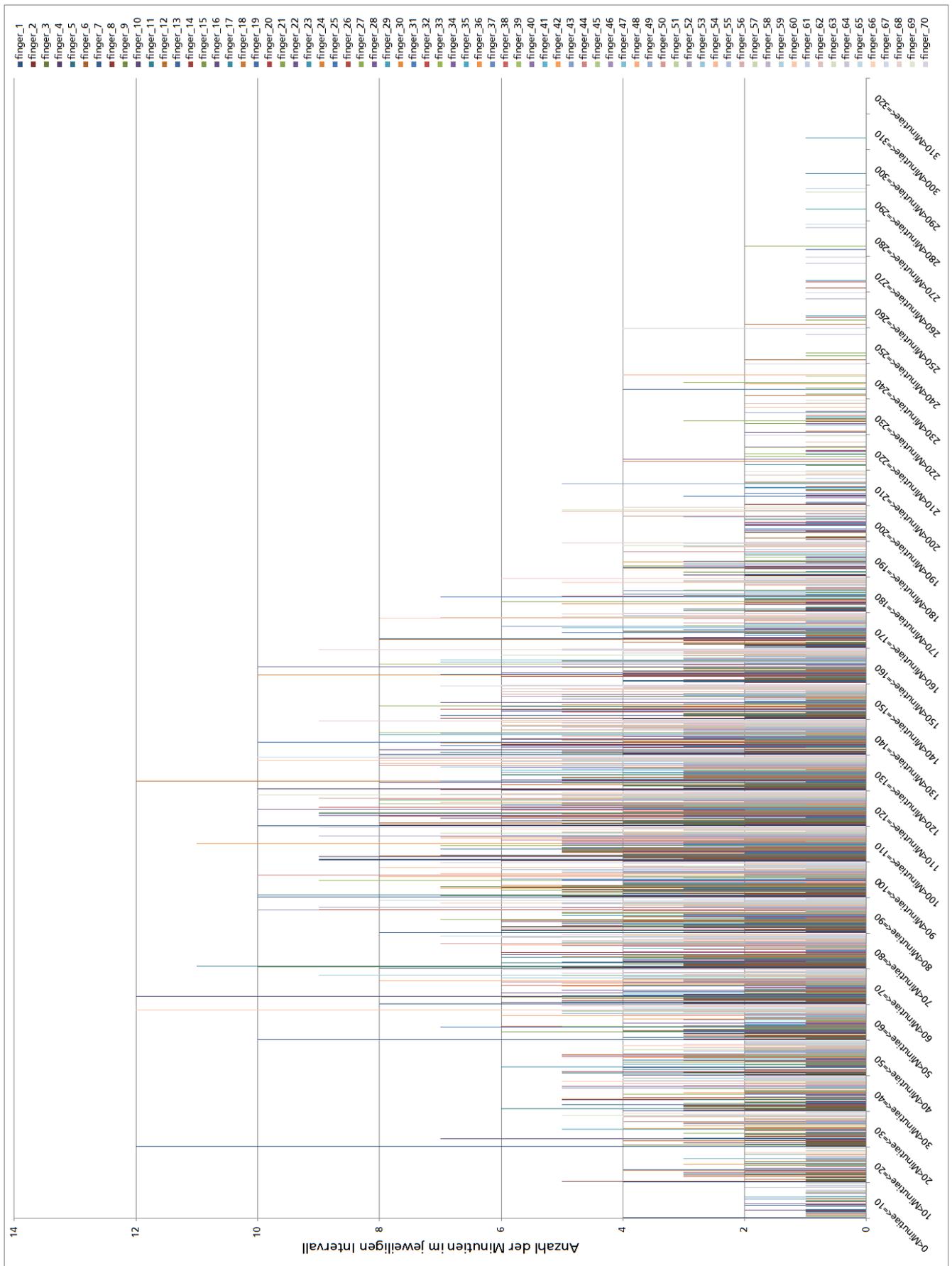


Abbildung 24: Verteilung der Minuten in 10-Pixel-Ringen um den Mittelpunkt

### 5.3.2. Minutiendaten 20er-Pixel-Intervalle

Nun ist deutlich erkennbar, dass die einzelnen Finger kaum differenziert angeschaut werden können, aufgrund der Datenmengen und der feinen Intervalle. Die Intervalle in dem Diagramm der sind nun auf 20 Pixel eingestellt, um einen weiteren Blickwinkel zu erhalten. Die betrachteten Daten sind immer noch die gleichen. Die erhobenen Daten befinden sich im Anhang C in Tabelle 7 bis Tabelle 8.

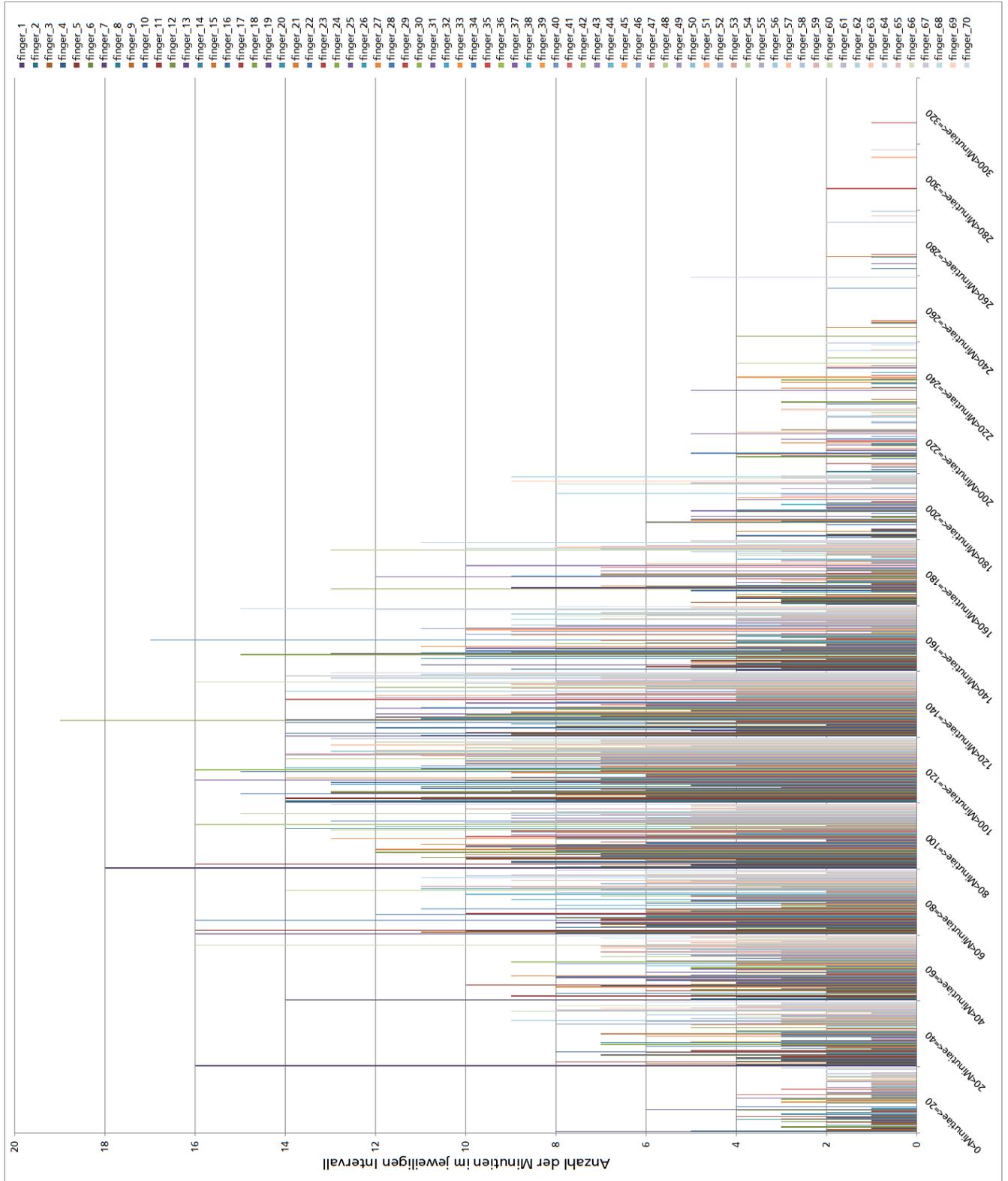


Abbildung 25: Verteilung der Minuten in 20-Pixel-Ringen um den Mittelpunkt

### 5.3.3. Minutiendaten 40er-Pixel-Intervalle

Die dritte Variante ist lediglich für den Vergleich gedacht, da diese unter Umständen etwas zu grob gewählt ist für aussagekräftige Betrachtungen. Tabelle 9 in Anhang D enthält hierzu die Daten.

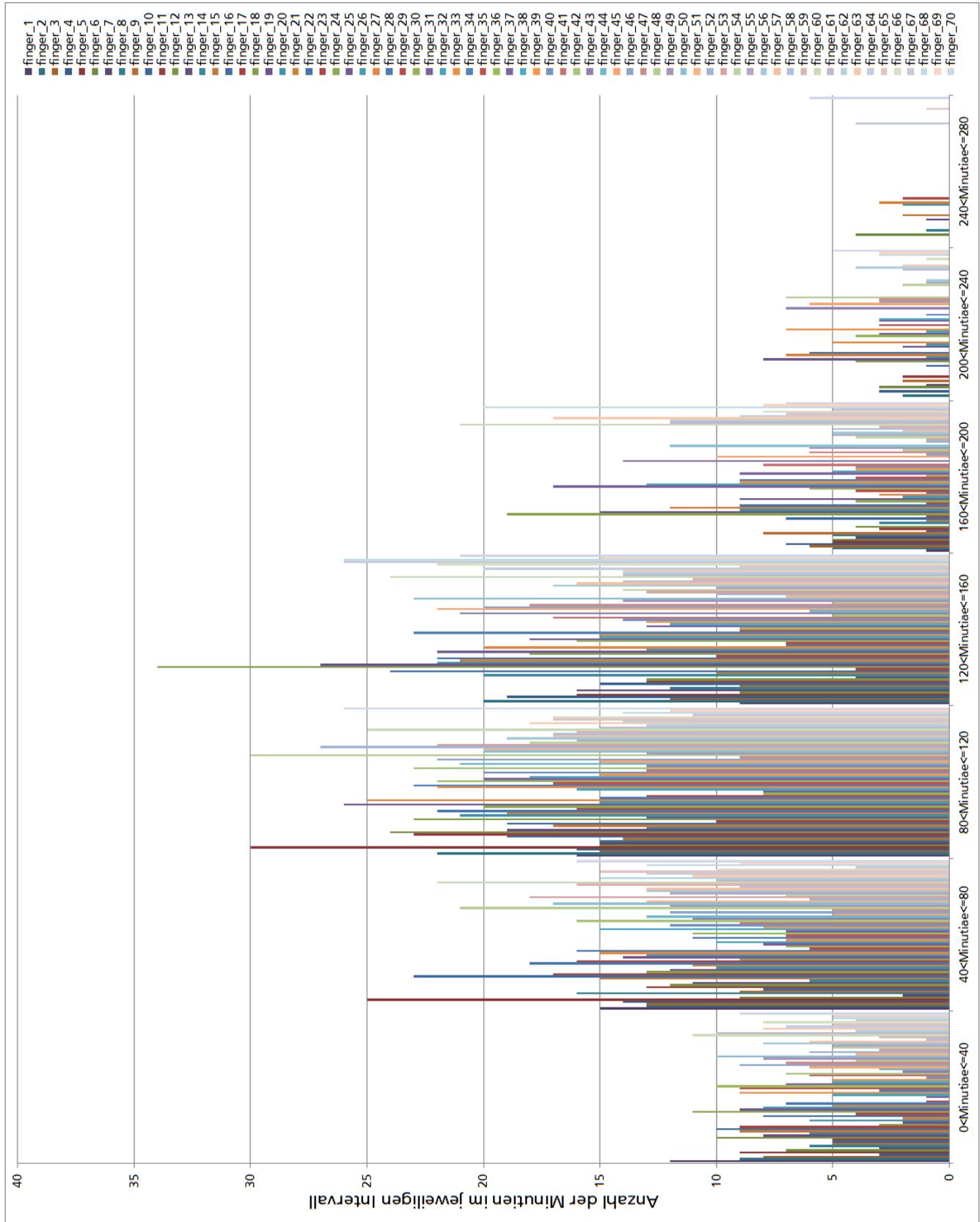


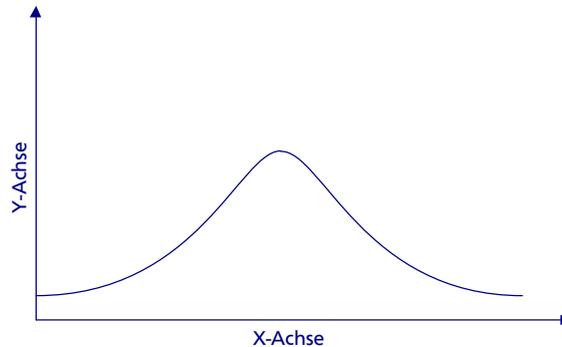
Abbildung 26: Verteilung der Minuten in 40-Pixel-Ringen um den Mittelpunkt

---

## 6. Fazit

---

Aus den Diagrammen ist klar erkennbar, dass die gewonnenen Daten, der Kurve einer Normalverteilung ähneln.



Dies könnte bedeuten, dass sich direkt um den Mittelpunkt herum und im äußeren Bereich der Abdrücke eher weniger Minutien befinden und es dazwischen eine „stärker besiedelte“ Region gibt. Jedoch muss die Tatsache mit beachtet werden, dass diese Ringbetrachtung die Flächengrößen der Ringe missachten würde. Durch die immer gleichen Ringbreiten steigt die Fläche der Ringe linear an, dies muss in die Betrachtung der Minutienverteilung mit einbezogen werden, da die steigende Anzahl von Minutien in einem Ring auch der steigenden Flächengröße zuzuordnen sein könnte. Entscheidend ist also die Minutiendichte auf jedem Ring.

Ein Ring ist hier nichts anderes als der Kreis mit dem maximalen Abstandswert als Radius abzüglich des inneren Kreises also mit dem minimalen Abstandswert als Radius. Lediglich der innerste Ring entspricht der einfachen Kreisfläche. So gilt für alle anderen Ringe mit äußerem Radius  $R$  und innerem Radius  $r$ :

$$A = (\pi * R^2) - (\pi * r^2) \quad \Leftrightarrow \quad A = \pi(R^2 - r^2)$$

Die linear ansteigende Flächengröße muss also mit in die Datenbetrachtung einbezogen werden.

Prinzipiell kann man auf die bereits berechneten Werte, um die Minutiendichte betrachten zu können die Minutienanzahlen in den jeweiligen Ringen auf den Ring- bzw. Kreisradius bezogen analysieren. Nehme man nun die Zahlen für die 20-Pixel-Ringe, da diese ein gutes Mittelmaß erzeugt haben zwischen Lesbarkeit und Genauigkeit, und dividiere jede Minutienanzahl durch den äußeren Radius. wobei noch erwähnt sei, dass es unerheblich ist, ob der innere oder der äußere Radius gewählt wird, Hauptsache die Berechnungen bleiben konstant bei einem Radius, allerdings würde dies für die erste Spalte bedeuten, dass der Nenner 0 wäre und eine Teilung durch 0 nicht möglich ist.

Wendet man also dieses Verfahren auf die 20er Datenreihe an, so erhält man ein Diagramm (Abbildung 27), welches die Minutiendichte in den einzelnen Ringen darstellt.

Aus diesem Diagramm erkennt man deutlich, dass im Allgemeinen die meisten Fingerabdrücke im Zentrum des Fingerabdrucks eine höhere Minutiendichte aufweisen und nach Außen diese Dichte immer mehr abnimmt. Spezielle Wolken gibt es jedoch nicht direkt zu erkennen, dies kann man insbesondere auch der Abbildung 28 entnehmen, welche alle Minutien der 70 Finger enthält.

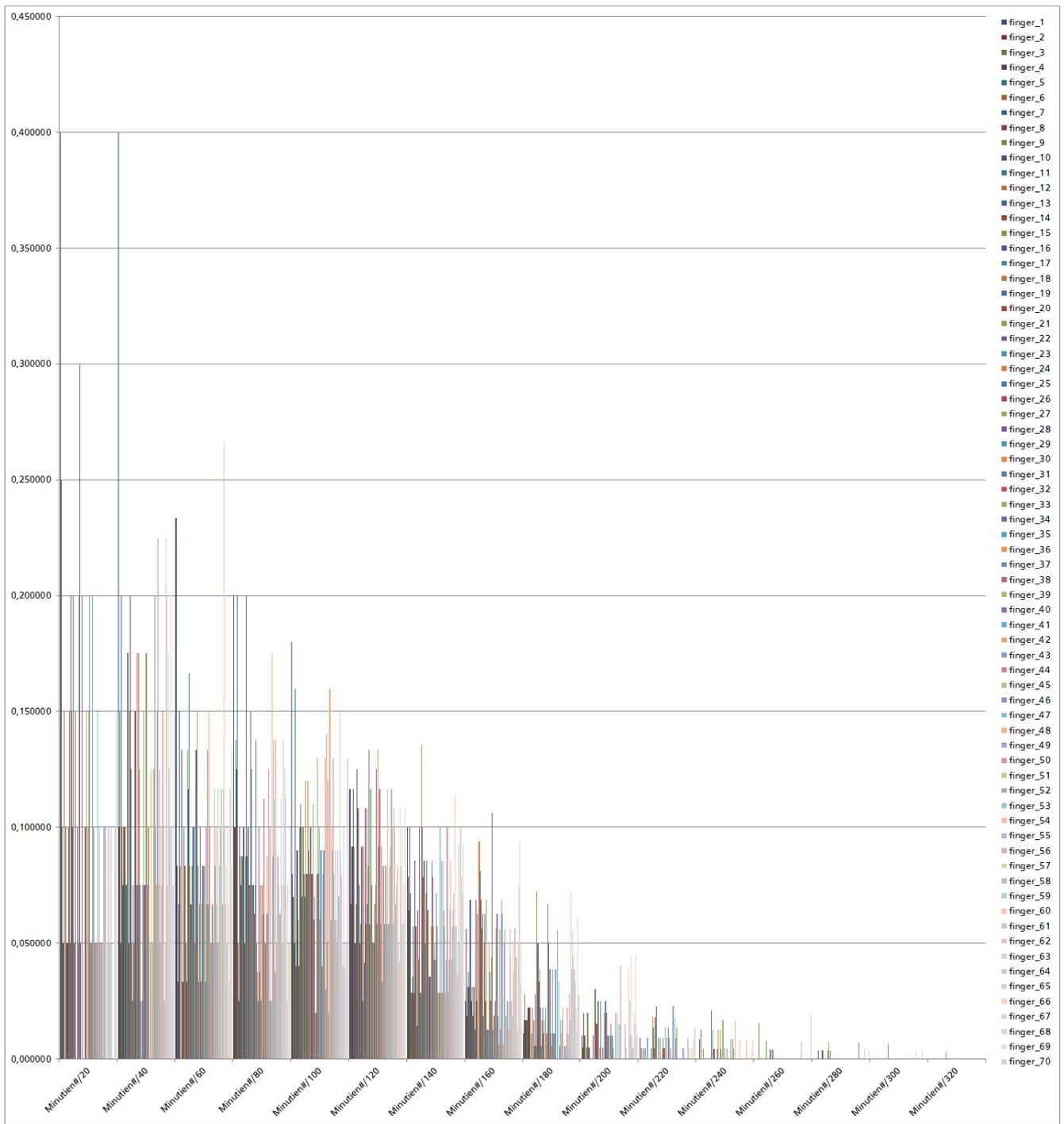


Abbildung 27: Verteilung der Minuten pro Radius

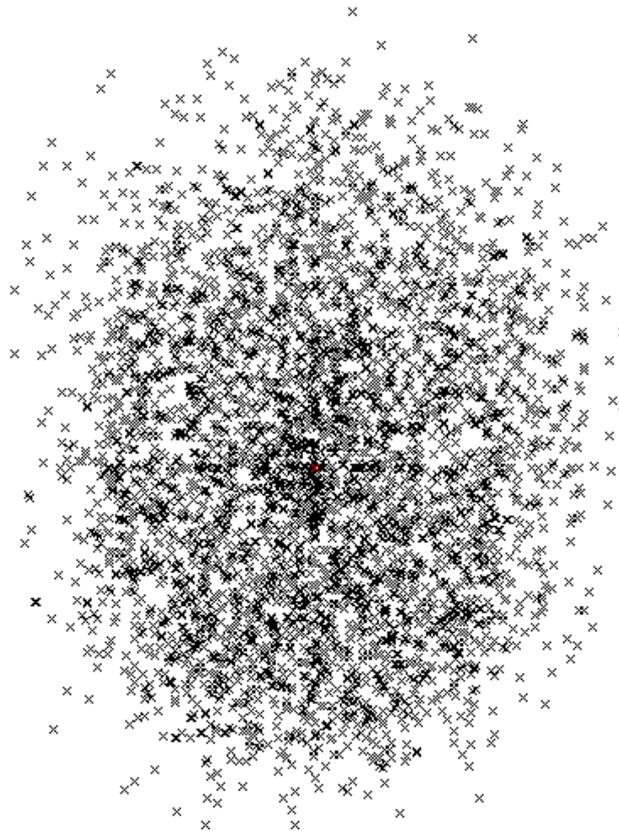


Abbildung 28: Verteilung der Minutien aus allen 70 Fingern

Aus diesen Daten lässt sich schlussfolgern, dass die ursprüngliche These der Arbeit, welche von Minutienwolken ausging in dem Sinne widerlegt ist, da es zwar eine Häufung im Zentrum von Fingerabdrücken zu geben scheint, aber dies sich lediglich auf die Existenz von verdichteten Minutienaufkommen beschränkt. Dieses dichtere Auftreten kann durch die doch sehr starke Verteilung der Minutien in dem Bereich nicht ausschlaggebend für Falscherkennung sorgen, als der Rest der Minutien, welche eher verstreut verteilt sind. Somit steigt die Falscherkennungsrate durch die hier geführten Beobachtung, bzw. die angenommenen Minutienwolken, welche offensichtlich so nicht existieren, wie angenommen, nicht explizit an.

Hinzu kommt auch, dass die Matching-Algorithmen nicht alleine auf die Position und Verteilung von Minutien ausgelegt sind, sondern einige andere Werte, wie Winkel, Art der Minutie etc. zusätzlich betrachten, was die Falschakzeptanzrate stark reduzieren kann. So kann man weiterhin schlussfolgern, dass eine explizite Häufung von Minutien, wie sie hier angenommen wurde keinen signifikanten Einfluss auf die Qualität von Erkennungsalgorithmen hat.

## 6.1. Ausblick

Bezüglich der implementierten Anwendung, so gibt es eine Vielzahl an Erweiterungsmöglichkeiten, die den Auswertungsprozess flexibler und umfangreicher gestalten könnten. So sind einige Aspekte fest in die Anwendung integriert worden, welche teilweise recht einfach dynamisch integriert werden können. Ein Beispiel sind die Fingerbilder-Dateien, diese sollten zwar vom Grundmuster gleich sein, jedoch wurde die Anzahl der Bilder pro Finger fix mit 8 in den Code integriert, dies ließe sich auch per Eingabefeld realisieren, indem der Anwender dort die Anzahl einträgt und dies dann als Variable im Programmcode implementiert ist.

Es gibt viele weitere Beispiele hierfür, doch da die Vorgaben für diese Arbeit von Beginn an feststanden, wurde hier auf Kosten der Nutzbarkeit, jedoch in Hinsicht auf Zeitersparnis eine Anwendung realisiert. Dies wurde jedoch mit dem Hintergrund so gewählt, dass diese Hilfsanwendung lediglich für diese Arbeit und lediglich von dem Verfasser der Arbeit genutzt würde.

---

In Hinsicht der Aussagekraft, der Ergebnisse, so gäbe es die Möglichkeit weitere Attribute der Minutien mit in die Betrachtung einfließen zu lassen und so mehr Informationsgehalt pro Minutie zu bekommen.

Jedoch wurde die These der Arbeit in der begrenzten Betrachtung, in der sie durchgeführt wurde, bereits widerlegt und es müsste ein anderer Ansatz gefunden werden, um festzustellen, ob Häufungen von Minutien einen Einfluss auf die Qualität von Erkennungsalgorithmen haben oder eben nicht, egal auf welche Methode man Betrachtungen hinsichtlich dieser Fragestellung durchführt.

# Anhang

## Anhang A

Tabelle 1: Zuordnung der fingerID zu den Dateinamen (bzw. eindeutigeren Hashwerten)

fingerID	filename	filehash	xMedian	yMedian
1	D:\BScArbeit\Fingerabdrücke\DB1_A\1_6.bmp	E219C6701A4BE122C0300EEFC9BA05	319	230
2	D:\BScArbeit\Fingerabdrücke\DB1_A\10_6.bmp	6E1941A0FAC77D18F1FABAF6D02A15	310	225
3	D:\BScArbeit\Fingerabdrücke\DB1_A\100_1.bmp	B0D85F5A30FDDC89F048F68EACCCCE	265	290
4	D:\BScArbeit\Fingerabdrücke\DB1_A\11_4.bmp	D532FA187B28516687361F51F2E4D7	318	212
5	D:\BScArbeit\Fingerabdrücke\DB1_A\12_8.bmp	2DB3FE217626499629B4CF74738C62	299	235
6	D:\BScArbeit\Fingerabdrücke\DB1_A\13_6.bmp	66D5E04FABD1F5A6FE22B45C7B03E6	269	267
7	D:\BScArbeit\Fingerabdrücke\DB1_A\14_6.bmp	460B2C682FF059BBB7833BD193F059	283	198
8	D:\BScArbeit\Fingerabdrücke\DB1_A\15_1.bmp	42C16779CD5317B4B6C79484BC458C	307	160
9	D:\BScArbeit\Fingerabdrücke\DB1_A\16_2.bmp	E5A3A0EA48B42E9321889D5BA200BF	326	301
10	D:\BScArbeit\Fingerabdrücke\DB1_A\17_6.bmp	4666DBFBD90B1114A9C5544125D7D2	275	328
11	D:\BScArbeit\Fingerabdrücke\DB1_A\18_7.bmp	297E09F684061FEE1B0A056B11F3E4	305	333
12	D:\BScArbeit\Fingerabdrücke\DB1_A\19_5.bmp	7FD78A1C1C32DA47687FF0883104CD	277	197
13	D:\BScArbeit\Fingerabdrücke\DB1_A\2_1.bmp	D109A8F22EEB0958EBCBD04377ABBA	289	129
14	D:\BScArbeit\Fingerabdrücke\DB1_A\20_7.bmp	D3D511D1EBF9A9C487B91D427040B8	318	297
15	D:\BScArbeit\Fingerabdrücke\DB1_A\21_1.bmp	0CC81257CA332A47167C78354273D6	261	171
16	D:\BScArbeit\Fingerabdrücke\DB1_A\22_7.bmp	38934A95BA93D8B15AE53D89B8DF43	270	283
17	D:\BScArbeit\Fingerabdrücke\DB1_A\23_2.bmp	1C603DDB6F18B699C36CC8E636931	307	334
18	D:\BScArbeit\Fingerabdrücke\DB1_A\24_3.bmp	C81FBAE2F2307AC4D3DEBEB33CB647	261	255
19	D:\BScArbeit\Fingerabdrücke\DB1_A\25_3.bmp	E876513C35FC4519A70B5EDA6E1D04	271	271
20	D:\BScArbeit\Fingerabdrücke\DB1_A\26_1.bmp	69A493B79CFC54BE3F5F43CA3F17E5	298	175
21	D:\BScArbeit\Fingerabdrücke\DB1_A\27_3.bmp	0373C15D0B2FFF685B6F3E9C779213	275	293
22	D:\BScArbeit\Fingerabdrücke\DB1_A\28_5.bmp	A910CF219C76F2381CE957A7A057F2	376	238
23	D:\BScArbeit\Fingerabdrücke\DB1_A\29_1.bmp	F449B933CB900E686B166A94E6D05	295	143
24	D:\BScArbeit\Fingerabdrücke\DB1_A\29_8.bmp	1BAA9D40944E538DFF2529BB15C79	333	256
25	D:\BScArbeit\Fingerabdrücke\DB1_A\3_3.bmp	5EBE9BC0FAE04856C958E9CC794411	310	276
26	D:\BScArbeit\Fingerabdrücke\DB1_A\30_5.bmp	68E265532AE3CDDFDC96ECF58644C	297	305
27	D:\BScArbeit\Fingerabdrücke\DB1_A\31_4.bmp	8BA1B1B9722A799448540B67D8B149	268	264
28	D:\BScArbeit\Fingerabdrücke\DB1_A\32_2.bmp	8B9016C3F3CC3CD382E083390D0D93	277	249
29	D:\BScArbeit\Fingerabdrücke\DB1_A\33_6.bmp	91F4B4BEDF46B14BC3EDFFAE77C4EE	361	286
30	D:\BScArbeit\Fingerabdrücke\DB1_A\34_5.bmp	567ED1E3B3D14BBOFF0C7D47927A59	272	265
31	D:\BScArbeit\Fingerabdrücke\DB1_A\35_5.bmp	1D981464908F3E506198A52321BB19	365	291
32	D:\BScArbeit\Fingerabdrücke\DB1_A\36_7.bmp	F2C9E4707E481D8DC410E7C3B4EFA1	338	254
33	D:\BScArbeit\Fingerabdrücke\DB1_A\37_2.bmp	862AB792AA1425173D38390434C489	269	275
34	D:\BScArbeit\Fingerabdrücke\DB1_A\38_4.bmp	94FC060BCA9DE3447C31BAD529B186	281	299
35	D:\BScArbeit\Fingerabdrücke\DB1_A\39_4.bmp	F26DE7BED6B555B4B26B5A4A001C3E	263	257
36	D:\BScArbeit\Fingerabdrücke\DB1_A\4_1.bmp	C21013AE2585F1DC09A9570FD8B8FF	294	140
37	D:\BScArbeit\Fingerabdrücke\DB1_A\40_5.bmp	BA328F68D51042606E8A1ECD9FFB3	294	264
38	D:\BScArbeit\Fingerabdrücke\DB1_A\41_5.bmp	092E06758D4C8269758A69D9F75FBE	349	323
39	D:\BScArbeit\Fingerabdrücke\DB1_A\42_4.bmp	3EA700ACBF8FA624B42BEA1B02446F	246	318
40	D:\BScArbeit\Fingerabdrücke\DB1_A\43_6.bmp	B9A5923FF5BCC848CF91CFA6E15292	252	314
41	D:\BScArbeit\Fingerabdrücke\DB1_A\44_5.bmp	4A9CD47F28FDA1D30E13C4E60E22F	260	329
42	D:\BScArbeit\Fingerabdrücke\DB1_A\45_1.bmp	2FB90BE5872986FF6C5B27C89F0497	252	120
43	D:\BScArbeit\Fingerabdrücke\DB1_A\46_5.bmp	EC47089E72A023F460C3A279E935FC	419	265
44	D:\BScArbeit\Fingerabdrücke\DB1_A\47_1.bmp	F8854825D7B43803C66A89638AE58	255	155
45	D:\BScArbeit\Fingerabdrücke\DB1_A\48_5.bmp	7D6C0D96CA41DF993E410C1EEBDDCE	244	271
46	D:\BScArbeit\Fingerabdrücke\DB1_A\49_4.bmp	7482D7827933C925D914F310014618	300	210
47	D:\BScArbeit\Fingerabdrücke\DB1_A\5_6.bmp	ABE047571A577CB4E284609357CF17	261	279
48	D:\BScArbeit\Fingerabdrücke\DB1_A\50_6.bmp	3ED4243E904BF3479914057B9A8CCF	262	313
49	D:\BScArbeit\Fingerabdrücke\DB1_A\51_7.bmp	FB05690DE92B65133ED1B065888E34	275	302
50	D:\BScArbeit\Fingerabdrücke\DB1_A\52_6.bmp	0BEF33281AC4E0A267708F0919B0CF	293	251
51	D:\BScArbeit\Fingerabdrücke\DB1_A\53_1.bmp	DB97E0CD8E67B9B4E0360201E4D4E5	297	125
52	D:\BScArbeit\Fingerabdrücke\DB1_A\54_1.bmp	5E5B57B0D15CB126F9AB3660868BC4	313	157
53	D:\BScArbeit\Fingerabdrücke\DB1_A\55_1.bmp	E2A45DEB492904C33FBF940B0D28C6	232	165
54	D:\BScArbeit\Fingerabdrücke\DB1_A\56_6.bmp	096E38C66028B32BC52A2E6A94951C	334	335
55	D:\BScArbeit\Fingerabdrücke\DB1_A\57_6.bmp	BE61019826F854016613828C607775	278	212
56	D:\BScArbeit\Fingerabdrücke\DB1_A\58_8.bmp	14BBC6E78E9D18307F1AAAF7A2EAF79	330	240
57	D:\BScArbeit\Fingerabdrücke\DB1_A\59_1.bmp	7A12AC4D9618997E033C3781CDF172	307	170
58	D:\BScArbeit\Fingerabdrücke\DB1_A\6_1.bmp	650981BC299DBB0AA8779A7A4823CA	340	191
59	D:\BScArbeit\Fingerabdrücke\DB1_A\60_1.bmp	280A772ADBB4E53B0E603D8EF1804D	272	154
60	D:\BScArbeit\Fingerabdrücke\DB1_A\61_3.bmp	C8826BF92FB5320AA2C01414D89E26	304	268
61	D:\BScArbeit\Fingerabdrücke\DB1_A\62_6.bmp	3F66B30970F12766DE7656BABB804C	270	266
62	D:\BScArbeit\Fingerabdrücke\DB1_A\63_5.bmp	63369DB7315E62B115F1D578EFF2F6	275	276
63	D:\BScArbeit\Fingerabdrücke\DB1_A\64_5.bmp	B6CB00A89F7A60DB941C223BDAB98	266	218
64	D:\BScArbeit\Fingerabdrücke\DB1_A\65_5.bmp	4D7C82A14D1659C5ACDC41D9C759B2	376	213
65	D:\BScArbeit\Fingerabdrücke\DB1_A\66_1.bmp	9D5F426A90D04199D30B6643999439	208	157
66	D:\BScArbeit\Fingerabdrücke\DB1_A\67_5.bmp	DD99ED61A8709AE6D57AE8D17C0B57	338	223
67	D:\BScArbeit\Fingerabdrücke\DB1_A\68_4.bmp	1CFB7346D83FF3E236C8D7A3AC032D	272	283
68	D:\BScArbeit\Fingerabdrücke\DB1_A\69_3.bmp	0943F39363F0C511D5455E9E8846AA	280	210
69	D:\BScArbeit\Fingerabdrücke\DB1_A\7_6.bmp	9D37BB9CB664758327D0FB2B49FA29	285	276
70	D:\BScArbeit\Fingerabdrücke\DB1_A\70_5.bmp	32E6C9D94BC143F65A0E7293F5D8CA	341	261

## Anhang B

In den folgenden Tabelle 2 bis Tabelle 6 beschreiben die Spalten mit den Werten immer die Anzahl der Minuten in dem entsprechenden Intervall. Also beschreibt z.B. „10<Minutiae<=20“ alle Minuten, welche sich im Abstand von 10 Pixeln zum Mittelpunkt bis zum Abstand von 20 Pixeln zum Mittelpunkt befinden usw.

Tabelle 2: Minutiendaten der 10-Pixel-Ringe um den Mittelpunkt (1/5)

finger	0<Minutiae<=10	10<Minutiae<=20	20<Minutiae<=30	30<Minutiae<=40	40<Minutiae<=50	50<Minutiae<=60	60<Minutiae<=70	70<Minutiae<=80
finger_1	4	4	12	4	4	10	8	8
finger_2	0	5	1	3	2	3	3	5
finger_3	1	1	4	2	1	1	1	10
finger_4	0	1	1	1	3	1	6	4
finger_5	0	1	2	6	5	4	5	11
finger_6	1	2	1	3	3	2	1	3
finger_7	0	0	3	0	0	0	0	2
finger_8	1	1	3	1	5	3	5	3
finger_9	0	1	1	3	0	2	3	4
finger_10	0	1	1	3	2	0	3	3
finger_11	1	1	2	1	4	2	1	6
finger_12	0	3	4	3	2	3	5	2
finger_13	1	0	2	5	0	3	3	5
finger_14	1	3	1	1	0	2	0	4
finger_15	1	2	3	3	2	6	6	1
finger_16	2	0	7	1	4	3	12	4
finger_17	1	3	3	2	6	4	5	2
finger_18	0	1	0	2	1	4	4	4
finger_19	0	1	0	1	1	3	4	4
finger_20	0	3	2	1	1	3	3	3
finger_21	0	0	2	0	2	3	2	4
finger_22	0	2	1	5	4	2	6	6
finger_23	1	0	0	3	4	2	4	6
finger_24	0	4	2	5	1	2	1	5
finger_25	2	4	2	1	1	7	4	2
finger_26	1	0	3	4	2	6	2	3
finger_27	0	0	3	2	4	5	2	4
finger_28	2	2	1	2	2	3	5	6
finger_29	0	0	1	0	0	2	2	2
finger_30	0	0	0	0	3	1	1	2
finger_31	0	0	0	1	4	2	1	1
finger_32	1	1	2	1	1	1	2	6
finger_33	1	2	2	4	1	3	1	2
finger_34	0	0	1	2	1	4	2	4
finger_35	1	1	5	2	3	2	1	1
finger_36	0	3	4	3	4	1	5	1
finger_37	2	2	1	2	1	1	4	1
finger_38	1	0	2	2	5	1	6	3
finger_39	1	0	2	2	3	1	4	0
finger_40	0	1	0	0	5	3	2	2
finger_41	2	2	0	2	2	2	1	4
finger_42	0	2	3	2	5	4	5	2
finger_43	1	1	0	0	4	2	4	1
finger_44	0	1	1	1	1	2	6	4
finger_45	0	1	2	3	1	2	1	1
finger_46	0	1	3	5	2	2	3	5
finger_47	0	3	3	1	2	1	0	2
finger_48	1	0	3	0	1	6	8	6
finger_49	1	1	3	5	3	2	3	4
finger_50	1	0	3	5	2	4	4	7
finger_51	1	0	3	3	4	0	4	5
finger_52	0	1	3	3	1	2	1	2
finger_53	1	0	3	1	2	5	6	5
finger_54	0	1	3	2	2	1	1	3
finger_55	0	2	3	1	2	3	4	3
finger_56	1	1	3	4	3	3	2	5
finger_57	0	2	3	2	2	5	5	1
finger_58	0	0	3	1	1	3	1	4
finger_59	0	1	3	0	2	5	9	0
finger_60	0	2	3	5	4	12	3	3
finger_61	2	0	3	4	2	4	1	3
finger_62	0	1	3	1	3	1	5	6
finger_63	0	1	3	2	1	4	1	5
finger_64	1	1	3	2	2	1	4	6
finger_65	1	0	3	4	2	4	2	7
finger_66	0	0	3	4	1	3	2	4
finger_67	0	0	3	3	2	0	1	1
finger_68	1	1	3	2	2	5	1	5
finger_69	1	1	3	2	2	3	0	4
finger_70	1	2	3	3	3	5	6	2

Tabelle 3: Minutiendaten der 10-Pixel-Ringe um den Mittelpunkt (2/5)

finger	80<Minutiae<=90	90<Minutiae<=100	100<Minutiae<=110	110<Minutiae<=120	120<Minutiae<=130	130<Minutiae<=140	140<Minutiae<=150
finger_1	8	10	4	10	6	8	4
finger_2	3	5	6	8	7	4	7
finger_3	0	7	5	3	4	5	1
finger_4	1	4	9	2	10	4	2
finger_5	6	10	9	5	3	7	2
finger_6	2	2	3	8	2	2	3
finger_7	5	4	2	4	3	2	7
finger_8	6	3	4	2	3	5	1
finger_9	1	5	4	4	4	0	3
finger_10	3	1	9	6	4	8	2
finger_11	6	4	8	5	4	4	3
finger_12	6	5	8	5	6	2	3
finger_13	6	4	7	2	2	0	1
finger_14	5	2	3	3	5	4	5
finger_15	4	6	4	3	5	1	3
finger_16	3	5	4	7	8	6	6
finger_17	4	3	2	1	1	3	0
finger_18	5	7	5	6	12	7	5
finger_19	5	3	2	3	7	7	6
finger_20	2	6	5	8	5	6	7
finger_21	5	7	5	2	6	6	5
finger_22	6	3	4	9	6	6	4
finger_23	3	5	5	3	5	2	1
finger_24	4	6	5	5	3	7	4
finger_25	6	4	7	9	2	10	6
finger_26	6	2	4	3	5	4	3
finger_27	7	4	5	9	5	4	8
finger_28	2	4	5	4	3	2	1
finger_29	3	4	2	4	3	2	2
finger_30	1	1	2	4	3	5	5
finger_31	2	0	3	3	6	6	3
finger_32	4	4	2	6	5	6	1
finger_33	4	9	7	2	6	2	2
finger_34	3	5	5	10	3	3	7
finger_35	5	5	3	4	4	3	2
finger_36	2	4	11	5	4	2	2
finger_37	3	6	5	6	6	4	2
finger_38	3	1	5	9	3	5	1
finger_39	5	3	3	4	3	1	1
finger_40	5	4	5	6	2	2	5
finger_41	5	4	1	3	6	8	1
finger_42	5	8	5	5	1	3	1
finger_43	1	2	4	6	5	7	2
finger_44	4	10	6	1	3	1	2
finger_45	1	4	2	8	4	8	5
finger_46	10	2	5	5	6	3	6
finger_47	2	0	4	3	7	1	3
finger_48	9	7	7	7	4	0	1
finger_49	2	4	5	2	3	3	5
finger_50	5	5	7	3	8	6	6
finger_51	9	0	3	8	3	1	3
finger_52	9	4	9	5	3	3	3
finger_53	3	5	6	8	5	4	1
finger_54	4	2	4	8	8	4	2
finger_55	3	6	3	4	4	2	2
finger_56	3	3	4	9	3	5	6
finger_57	4	1	7	5	3	6	1
finger_58	5	4	4	4	4	6	2
finger_59	5	2	5	4	6	2	2
finger_60	7	8	5	5	10	6	5
finger_61	5	4	1	5	5	3	2
finger_62	4	4	2	3	2	3	6
finger_63	2	3	3	10	5	2	4
finger_64	2	2	3	7	8	5	4
finger_65	8	2	4	3	4	3	2
finger_66	3	2	6	6	8	6	4
finger_67	3	1	4	3	10	4	7
finger_68	1	4	4	5	2	9	6
finger_69	4	1	3	4	4	6	3
finger_70	6	7	9	4	8	5	3

Tabelle 4: Minutiendaten der 10-Pixel-Ringe um den Mittelpunkt (3/5)

finger	150<Minutiae<=160	160<Minutiae<=170	170<Minutiae<=180	180<Minutiae<=190	190<Minutiae<=200	200<Minutiae<=210	210<Minutiae<=220
finger_1	0	2	0	0	0	0	0
finger_2	2	1	2	2	0	2	0
finger_3	2	2	3	1	0	0	0
finger_4	3	2	1	3	1	1	0
finger_5	4	2	1	2	0	0	0
finger_6	2	2	1	0	2	0	0
finger_7	4	1	3	0	0	1	0
finger_8	3	3	1	0	1	0	0
finger_9	2	3	1	3	1	0	1
finger_10	1	0	0	1	0	0	0
finger_11	2	1	1	1	0	0	2
finger_12	2	4	0	0	0	0	0
finger_13	1	0	0	0	0	0	0
finger_14	6	3	0	0	0	0	0
finger_15	1	1	0	0	0	0	0
finger_16	4	3	2	2	0	1	0
finger_17	0	1	0	0	0	0	0
finger_18	10	8	5	4	2	0	4
finger_19	7	8	1	4	2	3	0
finger_20	4	4	2	2	1	1	0
finger_21	4	1	6	4	1	1	3
finger_22	6	3	1	3	2	1	4
finger_23	2	1	0	0	0	0	0
finger_24	4	1	2	0	1	0	0
finger_25	4	2	2	3	2	2	0
finger_26	1	1	1	0	0	0	0
finger_27	3	1	2	0	0	0	2
finger_28	1	0	0	1	0	0	0
finger_29	0	2	2	0	0	0	0
finger_30	3	2	0	4	0	1	0
finger_31	3	5	7	3	2	1	1
finger_32	3	4	5	2	2	1	0
finger_33	5	4	3	2	0	1	2
finger_34	10	4	3	0	2	0	0
finger_35	0	1	1	1	1	2	0
finger_36	1	1	0	0	0	0	0
finger_37	1	3	4	0	2	2	1
finger_38	3	2	0	1	2	1	1
finger_39	8	1	1	2	0	0	0
finger_40	5	2	0	1	1	0	1
finger_41	2	5	2	1	0	0	0
finger_42	0	0	0	0	0	0	0
finger_43	7	6	4	2	2	5	0
finger_44	0	0	0	0	0	0	0
finger_45	5	3	3	2	2	2	2
finger_46	5	0	1	0	0	0	2
finger_47	7	2	1	2	1	2	1
finger_48	0	1	1	0	0	2	1
finger_49	3	2	1	0	3	0	0
finger_50	3	3	1	4	4	0	0
finger_51	0	0	0	0	0	0	0
finger_52	1	1	0	0	0	0	0
finger_53	3	1	0	0	0	0	0
finger_54	0	2	2	0	0	0	0
finger_55	2	1	1	2	1	1	0
finger_56	3	2	2	0	1	0	1
finger_57	6	2	0	0	0	0	0
finger_58	2	3	2	0	0	0	0
finger_59	1	1	2	0	0	0	0
finger_60	3	8	5	3	5	0	0
finger_61	4	4	3	3	2	2	0
finger_62	3	7	3	2	0	2	0
finger_63	3	5	3	4	5	1	0
finger_64	3	4	3	2	0	0	0
finger_65	0	4	2	1	0	0	0
finger_66	4	3	1	2	2	1	0
finger_67	5	1	1	3	0	0	0
finger_68	9	5	6	5	4	1	1
finger_69	2	3	2	1	2	2	1
finger_70	5	4	1	2	0	1	2

Tabelle 5: Minutiendaten der 10-Pixel-Ringe um den Mittelpunkt (4/5)

finger	220<Minutiae<=230	230<Minutiae<=240	240<Minutiae<=250	250<Minutiae<=260	260<Minutiae<=270	270<Minutiae<=280	280<Minutiae<=290
finger_1	0	0	0	0	0	0	0
finger_2	0	0	0	0	0	0	0
finger_3	0	0	0	0	0	0	0
finger_4	2	0	0	0	0	0	0
finger_5	0	0	0	0	0	0	0
finger_6	1	2	2	2	0	0	0
finger_7	0	0	0	0	0	0	0
finger_8	0	0	0	0	1	0	0
finger_9	0	1	0	0	0	0	0
finger_10	0	0	0	0	0	0	0
finger_11	0	0	0	0	0	0	0
finger_12	0	0	0	0	0	0	0
finger_13	0	0	0	0	0	1	0
finger_14	0	0	0	0	0	0	0
finger_15	0	0	1	1	0	0	0
finger_16	0	0	0	0	0	0	0
finger_17	0	0	0	0	0	0	0
finger_18	0	0	0	0	0	0	0
finger_19	1	4	0	0	0	0	0
finger_20	0	0	0	1	1	0	0
finger_21	2	1	1	0	0	2	0
finger_22	1	0	0	0	0	0	0
finger_23	0	0	0	1	1	0	1
finger_24	0	0	0	0	0	0	0
finger_25	0	0	0	0	0	0	0
finger_26	1	0	0	0	0	0	0
finger_27	3	0	0	0	0	0	0
finger_28	0	0	0	0	0	0	0
finger_29	0	0	0	0	0	0	0
finger_30	1	2	0	0	0	0	0
finger_31	1	0	0	0	0	0	0
finger_32	0	0	0	0	0	0	0
finger_33	1	3	0	0	0	0	0
finger_34	0	0	0	0	0	0	0
finger_35	1	0	0	0	0	0	0
finger_36	0	0	0	0	0	0	0
finger_37	0	0	0	0	0	0	0
finger_38	1	0	0	0	0	0	0
finger_39	0	0	0	0	0	0	0
finger_40	0	0	0	0	0	0	0
finger_41	0	0	0	0	0	0	0
finger_42	0	0	0	0	0	0	0
finger_43	2	0	0	0	0	0	0
finger_44	0	0	0	0	0	0	0
finger_45	1	1	0	0	0	0	0
finger_46	1	0	0	0	0	0	0
finger_47	0	0	0	0	0	0	0
finger_48	0	4	0	0	0	0	0
finger_49	0	0	0	0	0	0	0
finger_50	0	0	0	0	0	0	0
finger_51	0	0	0	0	0	0	0
finger_52	0	0	0	0	0	0	0
finger_53	0	0	0	0	0	0	0
finger_54	2	0	0	0	0	0	0
finger_55	0	0	0	0	0	0	0
finger_56	0	0	0	0	0	0	0
finger_57	0	0	0	0	0	0	1
finger_58	0	0	1	1	1	1	0
finger_59	0	0	0	0	0	0	0
finger_60	0	0	0	0	0	0	0
finger_61	0	0	0	0	0	0	0
finger_62	2	0	0	0	0	0	0
finger_63	1	0	0	0	0	0	0
finger_64	0	0	0	0	0	0	0
finger_65	0	0	0	0	0	1	1
finger_66	0	0	0	0	0	0	0
finger_67	0	0	0	0	0	0	0
finger_68	1	0	0	0	0	0	0
finger_69	0	0	0	0	0	0	0
finger_70	0	2	4	1	1	0	0

Tabelle 6: Minutiendaten der 10-Pixel-Ringe um den Mittelpunkt (5/5)

finger	290<Minutiae<=300	300<Minutiae<=310	310<Minutiae<=320
finger_1	0	0	0
finger_2	0	0	0
finger_3	0	0	0
finger_4	0	0	0
finger_5	0	0	0
finger_6	0	0	0
finger_7	0	0	0
finger_8	0	0	0
finger_9	0	0	0
finger_10	0	0	0
finger_11	0	0	0
finger_12	0	0	0
finger_13	0	0	0
finger_14	0	0	0
finger_15	0	0	0
finger_16	0	0	0
finger_17	0	0	0
finger_18	0	0	0
finger_19	0	0	0
finger_20	0	0	0
finger_21	0	0	0
finger_22	0	0	0
finger_23	1	1	0
finger_24	0	0	0
finger_25	0	0	0
finger_26	0	0	0
finger_27	0	0	0
finger_28	0	0	0
finger_29	0	0	0
finger_30	0	0	0
finger_31	0	0	0
finger_32	0	0	0
finger_33	0	0	0
finger_34	0	0	0
finger_35	0	0	0
finger_36	0	0	0
finger_37	0	0	0
finger_38	0	0	0
finger_39	0	0	0
finger_40	0	0	0
finger_41	0	0	0
finger_42	0	0	0
finger_43	0	0	0
finger_44	0	0	0
finger_45	0	0	0
finger_46	0	0	0
finger_47	0	0	0
finger_48	0	0	0
finger_49	0	0	0
finger_50	0	0	0
finger_51	0	0	0
finger_52	0	0	0
finger_53	0	0	0
finger_54	0	0	0
finger_55	0	0	0
finger_56	0	0	0
finger_57	0	0	0
finger_58	0	0	0
finger_59	0	0	0
finger_60	0	0	0
finger_61	0	0	0
finger_62	0	0	0
finger_63	0	0	0
finger_64	0	0	0
finger_65	0	0	0
finger_66	0	0	0
finger_67	0	0	0
finger_68	0	0	0
finger_69	0	0	0
finger_70	0	0	0

## Anhang C

In den folgenden Tabelle 7 und Tabelle 8 beschreiben die Spalten mit den Werten immer die Anzahl der Minuten in dem entsprechenden Intervall. Also beschreibt z.B. „20<Minutiae<=40“ alle Minuten, welche sich im Abstand von 20 Pixeln zum Mittelpunkt bis zum Abstand von 40 Pixeln zum Mittelpunkt befinden usw.

Tabelle 7: Minutiendaten der 20-Pixel-Ringe um den Mittelpunkt (1/2)

finger	0<Minutiae<=20	20<Minutiae<=40	40<Minutiae<=60	60<Minutiae<=80	80<Minutiae<=100	100<Minutiae<=120	120<Minutiae<=140
finger_1	8	16	14	16	18	14	14
finger_2	5	4	5	8	8	14	11
finger_3	2	6	2	11	7	8	9
finger_4	1	2	4	10	5	11	14
finger_5	1	8	9	16	16	14	10
finger_6	3	4	5	4	4	11	4
finger_7	0	3	0	2	9	6	5
finger_8	2	4	8	8	9	6	8
finger_9	1	4	2	7	6	8	4
finger_10	1	4	2	6	4	15	12
finger_11	2	3	6	7	10	13	8
finger_12	3	7	5	7	11	13	8
finger_13	1	7	3	8	10	9	2
finger_14	4	2	2	4	7	6	9
finger_15	3	6	8	7	10	7	6
finger_16	2	8	7	16	8	11	14
finger_17	4	5	10	7	7	3	4
finger_18	1	2	5	8	12	11	19
finger_19	1	1	4	8	8	5	14
finger_20	3	3	4	6	8	13	11
finger_21	0	2	5	6	12	7	12
finger_22	2	6	6	12	9	13	12
finger_23	1	3	6	10	8	8	7
finger_24	4	7	3	6	10	10	10
finger_25	6	3	8	6	10	16	12
finger_26	1	7	8	5	8	7	9
finger_27	0	5	9	6	11	14	9
finger_28	4	3	5	11	6	9	5
finger_29	0	1	2	4	7	6	5
finger_30	0	0	4	3	2	6	8
finger_31	0	1	6	2	2	6	12
finger_32	2	3	2	8	8	8	11
finger_33	3	6	4	3	13	9	8
finger_34	0	3	5	6	8	15	6
finger_35	2	7	5	2	10	7	7
finger_36	3	7	5	6	6	16	6
finger_37	4	3	2	5	9	11	10
finger_38	1	4	6	9	4	14	8
finger_39	1	4	4	4	8	7	4
finger_40	1	0	8	4	9	11	4
finger_41	4	2	4	5	9	4	14
finger_42	2	5	9	7	13	10	4
finger_43	2	0	6	5	3	10	12
finger_44	1	2	3	10	14	7	4
finger_45	1	5	3	2	5	10	12
finger_46	1	8	4	8	12	10	9
finger_47	3	4	3	2	2	7	8
finger_48	1	3	7	14	16	14	4
finger_49	2	6	5	7	6	7	6
finger_50	1	9	6	11	10	10	14
finger_51	1	3	4	9	9	11	4
finger_52	1	5	3	3	13	14	6
finger_53	1	2	7	11	8	14	9
finger_54	1	4	3	4	6	12	12
finger_55	2	3	5	7	9	7	6
finger_56	2	6	6	7	6	13	8
finger_57	2	4	7	6	5	12	9
finger_58	0	1	4	5	9	8	10
finger_59	1	2	7	9	7	9	8
finger_60	2	9	16	6	15	10	16
finger_61	2	8	6	4	9	6	8
finger_62	1	3	4	11	8	5	5
finger_63	1	7	5	6	5	13	7
finger_64	2	5	3	10	4	10	13
finger_65	1	4	6	9	10	7	7
finger_66	0	8	4	6	5	12	14
finger_67	0	4	2	2	4	7	14
finger_68	2	3	7	6	5	9	11
finger_69	2	3	5	4	5	7	10
finger_70	3	6	8	8	13	13	13

Tabelle 8: Minutiendaten der 20-Pixel-Ringe um den Mittelpunkt (2/2)

finger	160<Minutiae<=180	180<Minutiae<=200	200<Minutiae<=220	220<Minutiae<=240	240<Minutiae<=260	260<Minutiae<=280	280<Minutiae<=300
finger_1	2	0	0	0	0	0	0
finger_2	3	2	2	0	0	0	0
finger_3	5	1	0	0	0	0	0
finger_4	3	4	1	2	0	0	0
finger_5	3	2	0	0	0	0	0
finger_6	3	2	0	3	4	0	0
finger_7	4	0	1	0	0	0	0
finger_8	4	1	0	0	0	1	0
finger_9	4	4	1	1	0	0	0
finger_10	0	1	0	0	0	0	0
finger_11	2	1	2	0	0	0	0
finger_12	4	0	0	0	0	0	0
finger_13	0	0	0	0	0	1	0
finger_14	3	0	0	0	0	0	0
finger_15	1	0	0	0	2	0	0
finger_16	5	2	1	0	0	0	0
finger_17	1	0	0	0	0	0	0
finger_18	13	6	4	0	0	0	0
finger_19	9	6	3	5	0	0	0
finger_20	6	3	1	0	1	1	0
finger_21	7	5	4	3	1	2	0
finger_22	4	5	5	1	0	0	0
finger_23	1	0	0	0	1	1	2
finger_24	3	1	0	0	0	0	0
finger_25	4	5	2	0	0	0	0
finger_26	2	0	0	1	0	0	0
finger_27	3	0	2	3	0	0	0
finger_28	0	1	0	0	0	0	0
finger_29	4	0	0	0	0	0	0
finger_30	2	4	1	3	0	0	0
finger_31	12	5	2	1	0	0	0
finger_32	9	4	1	0	0	0	0
finger_33	7	2	3	4	0	0	0
finger_34	7	2	0	0	0	0	0
finger_35	2	2	2	1	0	0	0
finger_36	1	0	0	0	0	0	0
finger_37	7	2	3	0	0	0	0
finger_38	2	3	2	1	0	0	0
finger_39	2	2	0	0	0	0	0
finger_40	2	2	1	0	0	0	0
finger_41	7	1	0	0	0	0	0
finger_42	0	0	0	0	0	0	0
finger_43	10	4	5	2	0	0	0
finger_44	0	0	0	0	0	0	0
finger_45	6	4	4	2	0	0	0
finger_46	1	0	2	1	0	0	0
finger_47	3	3	3	0	0	0	0
finger_48	2	0	3	4	0	0	0
finger_49	3	3	0	0	0	0	0
finger_50	4	8	0	0	0	0	0
finger_51	0	0	0	0	0	0	0
finger_52	1	0	0	0	0	0	0
finger_53	1	0	0	0	0	0	0
finger_54	4	0	0	2	0	0	0
finger_55	2	3	1	0	0	0	0
finger_56	4	1	1	0	0	0	0
finger_57	2	0	0	0	0	0	1
finger_58	5	0	0	0	2	2	0
finger_59	3	0	0	0	0	0	0
finger_60	13	8	0	0	0	0	0
finger_61	7	5	2	0	0	0	0
finger_62	10	2	2	2	0	0	0
finger_63	8	9	1	1	0	0	0
finger_64	7	2	0	0	0	0	0
finger_65	6	1	0	0	0	1	1
finger_66	4	4	1	0	0	0	0
finger_67	2	3	0	0	0	0	0
finger_68	11	9	2	1	0	0	0
finger_69	5	3	3	0	0	0	0
finger_70	5	2	3	2	5	1	0

## Anhang D

In der folgenden Tabelle 9 beschreiben die Spalten mit den Werten immer die Anzahl der Minutien in dem entsprechenden Intervall. Also beschreibt z.B. „40<Minutiae<=80“ alle Minutien, welche sich im Abstand von 40 Pixeln zum Mittelpunkt bis zum Abstand von 80 Pixeln zum Mittelpunkt befinden usw.

Tabelle 9: Minutiendaten der 40-Pixel-Ringe um den Mittelpunkt

finger	0<Minutiae<=40	40<Minutiae<=80	80<Minutiae<=120	120<Minutiae<=160	160<Minutiae<=200	200<Minutiae<=240	240<Minutiae<=280
finger_1	12	15	16	9	1	0	0
finger_2	9	13	22	20	5	2	0
finger_3	8	13	15	12	6	0	0
finger_4	3	14	16	19	7	3	0
finger_5	9	25	30	16	5	0	0
finger_6	7	9	15	9	5	3	4
finger_7	3	2	15	16	4	1	0
finger_8	6	16	15	12	5	0	1
finger_9	5	9	14	9	8	2	0
finger_10	5	8	19	15	1	0	0
finger_11	5	13	23	13	3	2	0
finger_12	10	12	24	13	4	0	0
finger_13	8	11	19	4	0	0	1
finger_14	6	6	13	20	3	0	0
finger_15	9	15	17	10	1	0	2
finger_16	10	23	19	24	7	1	0
finger_17	9	17	10	4	1	0	0
finger_18	3	13	23	34	19	4	0
finger_19	2	12	13	27	15	8	0
finger_20	6	10	21	22	9	1	2
finger_21	2	11	19	21	12	7	3
finger_22	8	18	22	22	9	6	0
finger_23	4	16	16	10	1	0	2
finger_24	11	9	20	18	4	0	0
finger_25	9	14	26	22	9	2	0
finger_26	8	13	15	13	2	1	0
finger_27	5	15	25	20	3	5	0
finger_28	7	16	15	7	1	0	0
finger_29	1	6	13	7	4	0	0
finger_30	0	7	8	16	6	4	0
finger_31	1	8	8	18	17	3	0
finger_32	5	10	16	15	13	1	0
finger_33	9	7	22	15	9	7	0
finger_34	3	11	23	23	9	0	0
finger_35	9	7	17	9	4	3	0
finger_36	10	11	22	9	1	0	0
finger_37	7	7	20	13	9	3	0
finger_38	5	15	18	12	5	3	0
finger_39	5	8	15	13	4	0	0
finger_40	1	12	20	14	4	1	0
finger_41	6	9	13	17	8	0	0
finger_42	7	16	23	5	0	0	0
finger_43	2	11	13	21	14	7	0
finger_44	3	13	21	6	0	0	0
finger_45	6	5	15	22	10	6	0
finger_46	9	12	22	20	1	3	0
finger_47	7	5	9	18	6	3	0
finger_48	4	21	30	5	2	7	0
finger_49	8	12	13	14	6	0	0
finger_50	10	17	20	23	12	0	0
finger_51	4	13	20	7	0	0	0
finger_52	6	6	27	10	1	0	0
finger_53	3	18	22	13	1	0	0
finger_54	5	7	18	14	4	2	0
finger_55	5	12	16	10	5	1	0
finger_56	8	13	19	17	5	1	0
finger_57	6	13	17	16	2	0	0
finger_58	1	9	17	14	5	0	4
finger_59	3	16	16	11	3	0	0
finger_60	11	22	25	24	21	0	0
finger_61	10	10	15	14	12	2	0
finger_62	4	15	13	14	12	4	0
finger_63	8	11	18	14	17	2	0
finger_64	7	13	14	20	9	0	0
finger_65	5	15	17	9	7	0	1
finger_66	8	10	17	22	8	1	0
finger_67	4	4	11	26	5	0	0
finger_68	5	13	14	26	20	3	0
finger_69	5	9	12	15	8	3	0
finger_70	9	16	26	21	7	5	6

---

## Abbildungsverzeichnis

---

Abbildung 1: Zeilenweise von links nach rechts: Papillarlinienende, einfache Gabelung, zweifache Gabelung, dreifache Gabelung, einfacher Wirbel, zweifacher Wirbel, seitliche Berührung; Haken, Punkt, Intervall, X-Linie, einfache Brücke, zweifache Brücke, fortlaufende Linie .....	5
Abbildung 2: Papillarlinienendung und Gabelung .....	5
Abbildung 3: Grundaufbau eines FTR-Sensors mit Prisma; Grundaufbau eines FTR-Sensors mit Prismenfolie .....	6
Abbildung 4: Grundaufbau eines optischen Sensors ohne FTR .....	6
Abbildung 5: Grundaufbau eines kapazitiven Sensors .....	7
Abbildung 6: Grundaufbau eines thermischen Sensors .....	7
Abbildung 7: Grundaufbau eines RF-Sensors .....	7
Abbildung 8: Grundaufbau eines Ultraschallsensors .....	8
Abbildung 9: BioAPI Architektur .....	9
Abbildung 10: secunet biomiddle Architektur [SECFS] .....	9
Abbildung 11: Minutenvergleich innerhalb eines Fingerabdrucks, bzw. Darstellung zweier Bilder des gleichen Fingerabdrucks (P = Prüffingerabdruck, G = Galeriefingerabdruck) .....	11
Abbildung 12: Fehlerkurven bei biometrischen Verfahren .....	13
Abbildung 13: Veranschaulichung einer möglichen Verteilung von Minuten .....	17
Abbildung 14: Anwendungsoberfläche des Fingerprint Statistic Tools .....	22
Abbildung 15: Klasse <FingerprintStatisticTool> .....	23
Abbildung 16: Auszug der PreMatch Datenbanktabelle .....	24
Abbildung 17: Beispiel für die Berechnungen des Medians und der Auswahl eines Repräsentanten....	24
Abbildung 18: Klasse <PreMatching> .....	25
Abbildung 19: Klasse <DatabaseManager> .....	25
Abbildung 20: Minutia Record Format des ISO Standard 19794-2 [ISO19794-2, S. 21] .....	26
Abbildung 21: Klasse <GetFPData> .....	27
Abbildung 22: Datenbanktabelle mit allen gewählten Repräsentanten .....	27
Abbildung 23: Beispiel für Finger-Datenbanktabelle .....	28
Abbildung 24: Verteilung der Minuten in 10-Pixel-Ringen um den Mittelpunkt .....	30
Abbildung 25: Verteilung der Minuten in 20-Pixel-Ringen um den Mittelpunkt .....	31
Abbildung 26: Verteilung der Minuten in 40-Pixel-Ringen um den Mittelpunkt .....	32
Abbildung 27: Verteilung der Minuten pro Radius .....	34
Abbildung 28: Verteilung der Minuten aus allen 70 Fingern.....	35

---

## Tabellenverzeichnis

---

Tabelle 1: Zuordnung der fingerID zu den Dateinamen (bzw. eindeutigeren Hashwerten) .....	37
Tabelle 2: Minutiendaten der 10-Pixel-Ringe um den Mittelpunkt (1/5) .....	38
Tabelle 3: Minutiendaten der 10-Pixel-Ringe um den Mittelpunkt (2/5) .....	39
Tabelle 4: Minutiendaten der 10-Pixel-Ringe um den Mittelpunkt (3/5) .....	40
Tabelle 5: Minutiendaten der 10-Pixel-Ringe um den Mittelpunkt (4/5) .....	41
Tabelle 6: Minutiendaten der 10-Pixel-Ringe um den Mittelpunkt (5/5) .....	42
Tabelle 7: Minutiendaten der 20-Pixel-Ringe um den Mittelpunkt (1/2) .....	43
Tabelle 8: Minutiendaten der 20-Pixel-Ringe um den Mittelpunkt (2/2) .....	44
Tabelle 9: Minutiendaten der 40-Pixel-Ringe um den Mittelpunkt .....	45

---

## Literaturverzeichnis

---

- [BSI04] Bundesamt für Sicherheit in der Informationstechnik. *Studie: Evaluierung biometrischer Systeme, Fingerabdrucktechnologien – BioFinger*, Version 1.1. Bonn, August 2004.
- [FVC] International Fingerprint Verification Competition (FVC)  
<https://biolab.csr.unibo.it/FVCOnGoing/UI/Form/Home.aspx>
- [HEI10] heise online News. *Zahlen per Fingerabdruck: Zwischen Alltagstauglichkeit und Datenschutzbedenken*. Januar 2010.  
<http://www.heise.de/newsticker/meldung/Zahlen-per-Fingerabdruck-Zwischen-Alltagstauglichkeit-und-Datenschutzbedenken-909472.html>
- [HEN03] International Biometric Group. *The Henry Classification System*. 2003.
- [HSK10] O. Henniger, D. Scheuermann, T. Kniess. *On security evaluation of fingerprint recognition systems*. In Online Proc. of the Internat. Biometric Performance Testing Conf., Gaithersburg, Maryland, USA, 2010.
- [ISO19784-1] ISO/IEC FDIS 19784-1:2005(E). *Information technology - Biometric application programming interface - Part 1: BioAPI specification*. 2005.
- [ISO19794-2] ISO/IEC 19794-2:2005(E). *Information technology - Biometric data interchange formats - Part 2: Finger minutiae data*. 2005.
- [ISO19794-3] ISO/IEC FDIS 19794-3:2006(E). *Information technology - Biometric data interchange formats - Part 3: Finger pattern spectral data*. 2006.
- [L1] L-1 Identix, Teil der BioEngine®  
<http://www.l1id.com/pages/100-bioengine-sdk>
- [LF2] Bundesamt für Sicherheit in der Informationstechnik, Secunet Security Networks AG, Fraunhofer SIT. *Studie: Life-Finger 2*. 2008/9.
- [NBIS] NIST Biometric Image Software, enthält den NIST Bozorth3 Algorithmus.  
<http://www.nist.gov/itl/iad/ig/nbis.cfm>
- [NIST04-1] Watson, Craig I. and Garris, Michael D. and Tabassi, Elham and Wilson, Charles L. and McCabe, R. Michael and Janet, Stanley and Ko, Kenneth. National Institute of Standards and Technology. *User's Guide to Export Controlled Distribution of NIST Biometric Image Software (NBIS-EC)*. September 2004.
- [NIST04-2] Watson, Craig I. and Garris, Michael D. and Tabassi, Elham and Wilson, Charles L. and McCabe, R. Michael and Janet, Stanley and Ko, Kenneth. National Institute of Standards and Technology. *User's Guide to NIST Biometric Image Software (NBIS)*. Oktober 2004.
- [SEC07] secunet Security Networks AG, F. Steffen. *Modulare Biometriearchitektur mit secunet biomiddle*, Version 1.0.0. 2007

- 
- [SEC09] secunet Security Networks AG, Michael Schwaiger. *Dokumentation der SOAP-Schnittstelle*, Version 2.0. Dezember 2009.
- [SECFS] secunet Security Networks AG. secunet biomiddle Factsheet BM\_V 1\_12 / 10\_D. *Middleware für biometrische Systeme und elektronische Ausweise*.
- [SIEM] Siemens SmartMatcher Algorithmus.  
<https://www.cee.siemens.com/web/austria/en/temp/biometrics/our/Pages/smartmatcher.aspx>
- [TT06] TeleTrust Deutschland e.V. *Bewertungskriterien zur Vergleichbarkeit biometrischer Verfahren – Kriterienkatalog*, Version 3.0. August 2006.

Weitere Quellen:

- Duden - Die deutsche Rechtschreibung. 4. Auflage.
- Fingerprint algorithms:  
[http://fingerchip.pagesperso-orange.fr/biometrics/types/fingerprint\\_algo.htm](http://fingerchip.pagesperso-orange.fr/biometrics/types/fingerprint_algo.htm)