

Seminararbeit zur Projektgruppe CSI:PC<sup>2</sup>  
Fingerprint Recognition  
Algorithmen zur Feature Extraction – Teil 1

Pascal Deppe

31. Mai 2007



**UNIVERSITÄT PADERBORN**  
*Die Universität der Informationsgesellschaft*

# 0 Einleitung

---

Biometrische Merkmale zur Identifikation und Authentifizierung sind momentan ein aktuelles Thema in der Forschung. Nie zuvor wurde versucht, ein so hohes Maß an Sicherheit zu gewährleisten - sei es bei der Identifikation von Personen (Biometrischer Pass) oder Verifikation (z.B. Zugangskontrolle). Ein schon sehr lange betrachtetes Merkmal ist der Fingerabdruck, der für jeden Menschen einzigartig ist und relativ einfach entnommen werden kann. Diese Art der Authentifizierung ist mittlerweile schon im low-cost Bereich angekommen, wo z.B. Fingerprintsensoren in Notebooks eingebaut werden, um dem Benutzer eine Kennwortabfrage zu ersparen. Leider sind diese low-cost Sensoren auch nur das – billige Ware, denn die aufgenommene Qualität des Abdrucks lässt deutlich zu wünschen übrig. Dies macht sich dann bemerkbar, wenn sich der Benutzer selber vom Notebook aussperrt, da sein Fingerabdruck nicht richtig erkannt wurde.

Die Fingerabdruckerkennung ist seit je her auf gute Bildqualität angewiesen, denn nur unter optimalen Aufnahmebedingungen lassen sich die spezifischen Merkmale mit größter Präzision entnehmen und führen zu einem eindeutigen Ergebnis. Doch diese Voraussetzungen sind nur extrem selten gegeben, denn die Umstände, in denen ein Fingerabdruck entnommen wird, sind jedes Mal unterschiedlich. Um trotzdem einen gewissen Standard an Bildqualität einhalten zu können, ist es notwendig, dass das online eingelesene Fingerabdruckbild durch mehrere Verfahren aufgebessert wird.

Im Verlauf des Authentifizierungsprozesses gibt es mehrere Ansatzstellen, an denen es möglich ist, das Resultat durch zielgerichtete Manipulation zu verbessern. Dabei denke ich zum einen an einfache Bildbearbeitungswerkzeuge, um die Bildqualität zu verbessern so wie an spezielle Analysealgorithmen, die durch gezielte Messungen und Annäherungen Informationen aus dem vorhandenen Bild extrahieren und ggf. extrapolieren können, um z. B. unkenntliche Bereiche zu rekonstruieren oder aber zu ignorieren.

In Kapitel 1 beschäftige ich mich mit der eigentlichen Bildverbesserung, also um Methoden, das Ausgangsmaterial für spätere Bearbeitung optimal vorzubereiten. Dazu betrachte ich einfache Bildbearbeitungsmethoden sowie einen konkreten Algorithmus zum generellen Verbessern des Abdruckbildes.

Nach Abschließen der Bildoptimierung gehe ich in Kapitel 2 auf die eigentliche Feature Extraction ein. Dabei werfe ich zunächst einen Blick auf die verschiedenen Typen von Merkmalen, die sich aus einem Fingerabdruck entnehmen lassen und werde danach zwei Extraktionsalgorithmen vorstellen, die sich auf einen speziellen Typ festgelegt haben, nämlich den globalen Merkmalen.

Abschließen fasse ich meine Ergebnisse in Kapitel 3 noch einmal zusammen und gebe einen Ausblick auf die Entwicklung der nun schon mehrere Jahrzehnte existierenden Fingerabdruckerkennung.

# 1 Bildoptimierung

Schlechte Bildqualität ist ein fundamentales Problem beim Erkennen und Identifizieren eines Fingerabdruckes. Denn an gleich zwei wichtigen Stellen im Erkennungsprozess können Bildfehler zu unerwünschten Fehlern und somit zum falschen Ergebnis führen. Die erste Stelle ist dabei die eigentliche Merkmalerkennung. Durch Bildfehler, sei es durch unsaubere oder unscharfe Aufnahme, kann ein automatisierter Algorithmus in die Irre geführt werden und somit falsche oder inkorrekte Merkmale ermitteln. Werden nun diese fehlerhaften Merkmale zur Identifikation bzw. Verifikation des Benutzers mit einer Datenbank verglichen, ergeben sich Folgefehler: die zweite Stelle, an der sich eine schlechte Bildqualität bemerkbar macht, ist der eigentliche Vergleich mit den Templates in der Datenbank. Fehlerhafte Merkmale können dazu führen, dass identische Fingerabdrücke trotzdem zurückgewiesen werden, da einige oder zu viele Merkmale nicht oder nicht ganz übereinstimmen. Dies macht sich dann in der False Rejection Rate (FRR) bemerkbar. Anders herum kann es auch vorkommen, dass bei schlechter Bildqualität durch Zufall gewisse Merkmale extrahiert werden, die zu einem erfolgreichen Matching mit einem Datenbanktemplate führen, obwohl der eigentliche Fingerabdruck nicht in der Datenbank vorhanden ist. Dies zeigt sich dann in einer hohen False Acceptance Rate (FAR).

Um es den Extraktionsalgorithmen möglichst leicht zu machen, ist es notwendig, fehlerhafte sowie überflüssige Bildinformationen zu entfernen und den verbleibenden Teil für eine effiziente und schnelle Bearbeitung vorzubereiten. Dabei werden gezielt Bildinformationen vernichtet, um dem Algorithmus eindeutigeren Vorlagen zu bieten. Dies geschieht zum Beispiel dadurch, dass

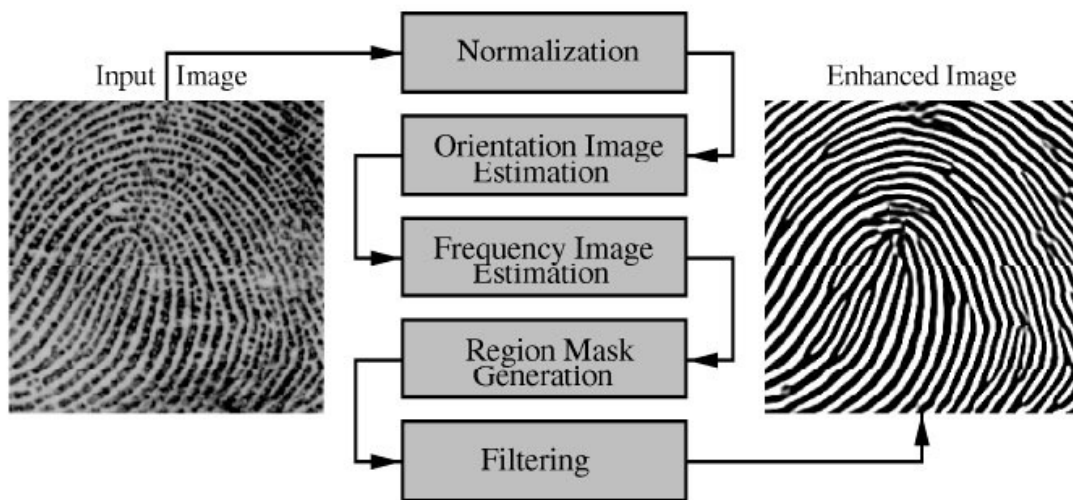
nun Graustufenbilder oder sogar nur schwarz-weiß Bilder erstellt werden (Binarisierung). Wo es bei Grauwerten noch gewissen Handlungsspielraum gibt, ist er im s/w Bild nicht mehr vorhanden. Somit kann durch Reduzieren von Bildinformationen die Erkennungsrate merkbar gesteigert werden.

## 1.1 Nutzung von Bildbearbeitungswerkzeugen

In diesem Abschnitt möchte ich nur ganz kurz auf die Verwendung von Bildbearbeitungswerkzeugen zur Bildverbesserung eingehen. Ein einfacher, aber trotzdem effektiver Weg ist es, die Helligkeit und den Kontrast der Aufnahme anzupassen. So ist es möglich, aus einem Graustufenbild die einzelnen Hell- und Dunkelwerte besser zu differenzieren. Wie auf Abbildung 1 zu erkennen ist, lässt sich somit aus dem aufgenommenen Graustufenbild (links) ein s/w Bild erzeugen (mitte), welches dennoch die vorhandenen Merkmale aufweist. Durch weitere Tricks kann man die schwarzen Ridge-Linien ausdünnen, so dass sie nur noch 1 Pixel breit sind (rechts). Das nun erhaltene Bild lässt sich leicht analysieren und es können mit einfachen Mitteln die Merkmale erkannt und extrahiert werden. Der Nachteil bei dieser sehr einfachen Variante ist, dass es nur bei sehr guter Bildqualität gelingt. Sollte der Fingerabdruck schon bei der



**Abbildung 1:** vom Graustufenbild über die Binarisierung zur Skelettierung – Vorbereitung des Bildes für die Feature Extraction



**Abbildung 2:** vom Ausgangsbild in fünf Schritten zum optimierten Ergebnis

Aufnahme verschmutzt sein, so ziehen sich diese Bildfehler in allen Schritten dieser Prozedur durch und produzieren damit fehlerhafte Merkmale.

Einen Weg, wie man trotz schlechtem Ausgangsbild hier weiterkommt, möchte ich nun etwas genauer betrachten: es ist ein schneller Algorithmus, der zwischen Aufnahme und Weiterverarbeitung geschaltet werden kann (also zwischen dem ersten und zweitem Bild von Abb. 1).

## 1.2 Fingerprint Image Enhancement Algorithm

1998 stellten Lin Hong, Yifei Wan und Anil Jain [1] einen Algorithmus vor, der die Bildqualität und somit die Erkennungsrate effizient steigern kann. Das Prinzip dieses Algorithmusses ist die Ausnutzung globaler Merkmale (Rillenfrequenz und -richtung), um einen Filter zu erstellen, der die Konturen des Bildes herausarbeitet und die unbrauchbaren Bereiche für die weitere Verarbeitung eliminiert. Abbildung 2 zeigt den schematischen Ablauf, der aus fünf einzelnen Punkten besteht, die ich nun näher erläutern will. Dabei beschränke ich mich auf die Arbeitsweise und vernachlässige die einzelnen

mathematischen Formeln, die in [1] nachgelesen werden können.

### 1.2.1 Normalisieren

Um eine vernünftige Ausgangslage zu schaffen, ist es notwendig, die Bildhelligkeit zu standardisieren. Dazu wird ein Prinzip ähnlich der Transformation einer Normalverteilung in eine Standardnormalverteilung angewandt: vom Ausgangsbild wird ein durchschnittlicher Grauwert ermittelt, welcher dem Mittelwert  $m$  einer Normalverteilung entspricht. Anschließend werden die Abweichungen von diesem Mittelwert bestimmt, was der Standardabweichung entspricht. Der eigentliche Schritt erfolgt nun durch transformieren der vorhandenen Grauwerte zu einem Standardwert, zum Beispiel 128 für den Mittelwert (Hälfte von



**Abbildung 3:** Ein Fingerabdruck vor und nach dem Normalisieren

# 1 Bildoptimierung

8 Bit) und einer geeigneten Standardabweichung. Abbildung 3 zeigt das Ergebnis nach einer Normalisierung.

## 1.2.2 Berechnen des Orientierungsfeldes

Im zweiten Schritt wird nun ein Orientierungsfeld berechnet, welches den Rillenverlauf wiedergeben soll. Dazu wird das Bild in gleichgroße quadratische Blöcke eingeteilt (beispielsweise  $16 \times 16$  Pixel). Nun wird für jeden Pixel in x- sowie in y-Richtung der Grauverlauf berechnet. Sobald dies geschehen ist, können die Werte für jeden Block zusammengefasst werden, um eine angenäherte Richtung zu berechnen, die orthogonal zum Fourier Spektrum verläuft. Da es durch Bildrausche und Störungen zu ungenauen Richtungsangaben kommen kann, lässt man noch einmal einen Tiefpassfilter über das Bild laufen. Das Ergebnis ist ein weichgezeichnetes Bild, bei dem nun die Grauverläufe sanfter ineinander übergehen. Damit lässt sich das Orientierungsfeld noch einmal nachjustieren, wie es in Abbildung 4 zu erkennen ist.

Bemerkung: Die Richtungsvektoren benachbarter Blöcke ändern sich nicht stark, sofern kein Singular Point in der Nähe liegt.

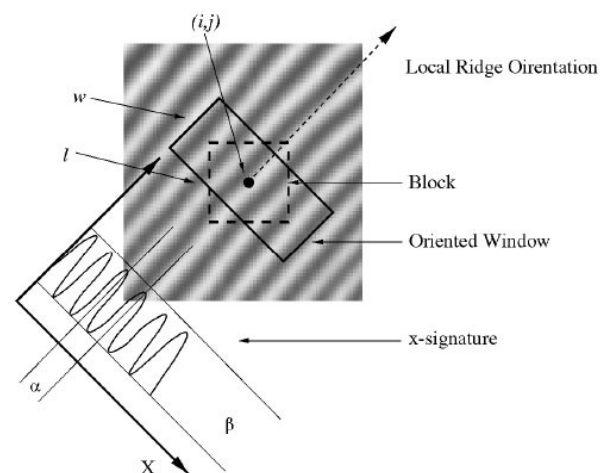


**Abbildung 4:** Das Orientierungsfeld – vor und nach einem Tiefpassfilter

## 1.2.3 Berechnen der Rillenfrequenzen

Um die Struktur des Fingerabdrucks hinreichend beschreiben zu können, reicht es nicht, einfach nur die Richtungen wieder zu geben. Es muss zusätzlich noch die Rillenfrequenz berücksichtigt werden, die stark von der Auflösung abhängt.

Das Ermitteln der Rillenfrequenz geschieht wieder Block für Block. Dabei wird ausgenutzt, dass der jeweilige Richtungsvektor schon bekannt ist. Zunächst wird ein Fenster um den Blockmittelpunkt gewählt, welches senkrecht zum Richtungsvektor liegt und die gleiche Höhe hat wie der Block – jedoch doppelte Breite. In diesem Fenster wird nun die Rillenfrequenz ermittelt, indem sie durch eine Sinusfunktion angenähert wird (siehe Abbildung 5).



**Abbildung 5:** Annäherung der Rillenfrequenz durch eine Sinusfunktion mit Intervalllänge  $\alpha$

Dazu wird der Helligkeitsverlauf entlang der breiten Achse ermittelt, welcher einer Sinusfunktion ähneln sollte. Lassen sich eindeutige Peaks bestimmen, so kann die Frequenz berechnet werden. Eine gültige Frequenz liegt zwischen  $1/3$  und  $1/25$ , eine Bildauflösung von 500 dpi vorausgesetzt. Liegt die errechnete Frequenz nicht in diesem Intervall, kann davon ausgegangen werden, dass es

# 1 Bildoptimierung

---

kein Rillenmuster war, welches in dem entsprechenden Block untersucht wurde. Daher wird die Frequenz auf den Wert  $-1$  gesetzt. Ein Tiefpassfilter kann auch hier helfen, die Bildqualität in unkenntlichen Blöcken soweit zu verbessern, dass eine Frequenzerkennung möglich wird.

## 1.2.4 Erstellen einer Regionsmaske

An dieser Stelle gilt es nun, die einzelnen Blöcke zu bewerten hinsichtlich ihrer weiteren Gebräuchlichkeit. Es muss ein Verfahren her, welches ein Einteilen der Blöcke in brauchbares und unbrauchbares Material aufteilen. Dazu wird nun der Helligkeitsverlauf der einzelnen orientierten Fenster betrachtet: es werden drei Parameter berechnet. 1. die Amplitude der Rillenfrequenz – dies geschieht durch Berechnung der Helligkeitsunterschiede zwischen Ridge und Valley. 2. die Frequenz – sie ergibt sich aus dem Kehrwert der Distanz zweier Peaks (dunkelste Stellen zweier Ridges). 3. die Varianz – sie wird durch Aufsummieren des Quadrates der Helligkeitsabweichung zum Mittelwert des Fensters, geteilt durch die Fensterlänge, bestimmt. Diese drei Werte werden nun in ein dreidimensionales Koordinatensystem eingetragen. Durch empirische Ermittlung konnte das Autorenteam sechs Cluster ausmachen, von denen vier Cluster eine Menge von brauchbarem Material indizierten und zwei Cluster, die unbrauchbare Stellen markierten.

Es wird nun ein Array erstellt, das für jede x-y-Koordinate entweder 1 (für brauchbarer Bereich) oder 0 (für unbrauchbarer Bereich) speichert. Durch Aufsummation der Werte lässt sich die Anzahl der brauchbaren Bereiche ermitteln. Unterschreitet dieser Wert eine vorher festgelegte Schwelle von z.B. 40, so kann der Fingerabdruck zurück

gewiesen werden, da die Aufnahme zu wenig verwertbare Informationen enthält.

## 1.2.5 Anwenden des Filters

Im fünften und letzten Schritt wird nun der eigentliche Filter angewandt. Benutzt wird ein Gaborfilter als Bandpassfilter. Der Gaborfilter ist ein richtungs- und frequenzabhängiger Filter, der exakt unseren Anforderungen entspricht, um die Wellenstruktur der Ridges/Valleys herauszuarbeiten und dabei unklare Bereiche säubert.

Die beiden benötigten Parameter haben wir soeben Block für Block bestimmt und können nun den Gaborfilter anwenden. Das einzige, was der Gaborfilter noch als Eingabe benötigt ist die Fenstergröße, in der der Filter aktiv werden darf. Dazu ist zu sagen, dass ein Kompromiss gemacht werden muss: je größer das Fenster, umso besser entfernt er Verunreinigungen des Musters, jedoch erzeugt so auch leichter falsche Rillen. Je kleiner das Filterfenster gemacht wird, umso geringer produziert er neue Fehler, und die Herausarbeitung der eigentlichen Struktur fällt schwächer aus. Ein vom Autorenteam empirisch ermittelter Wert mit den besten Resultaten ist 4.0 in x- und y-Richtung bei einer Gaborfiltergröße von 11.

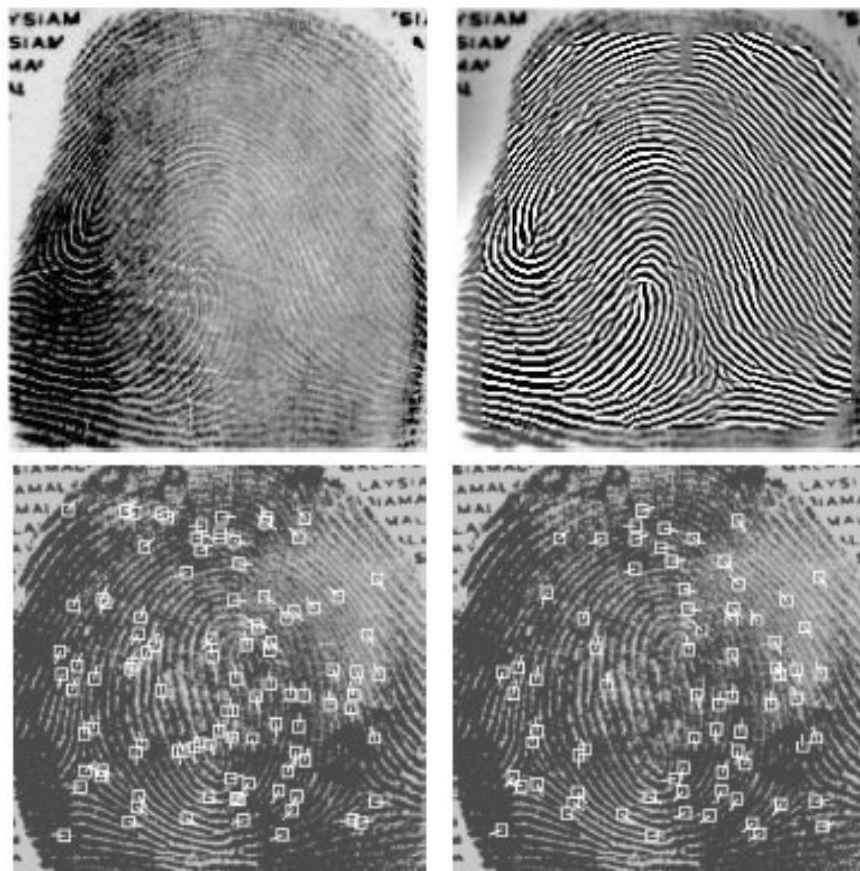
## 1.2.6 Auswertung und Beurteilung

Um eine qualitative Aussage über die Performance ihres Algorithmusses zu machen, benutzten die Autoren aus [1] einen *Goodness Index* (GI). Dieser GI sagt aus, wie viel Prozent der von einem menschlichen Experten erkannten Minutien von dem Algorithmus erkannt werden. Vor dem Einsatz des Algorithmusses wurden durch Tests ein GI-Mittelwert von 0,24 und eine Varianz von 0,05 ermittelt. Mit Hilfe des neuen Algorithmusses stieg der Mittelwert auf 0,39 und die Varianz fiel auf 0,04. Dies

# 1 Bildoptimierung

lassen auch die Bilder in Abbildung 6 andeuten: die schwachen Linien wurde sehr gut verstärkt und falsche/inkorrekte Minutien wurde erst gar nicht produziert. Laut den Autoren arbeitet dieser Algorithmus sehr schnell und effizient. Er

ist einfach in einen Erkennungsprozess zu integrieren und verbessert die Bildqualität und damit auch die Erkennungsrate deutlich.



**Abbildung 6:** links jeweils ohne und rechts mit Anwendung des Algorithmusses

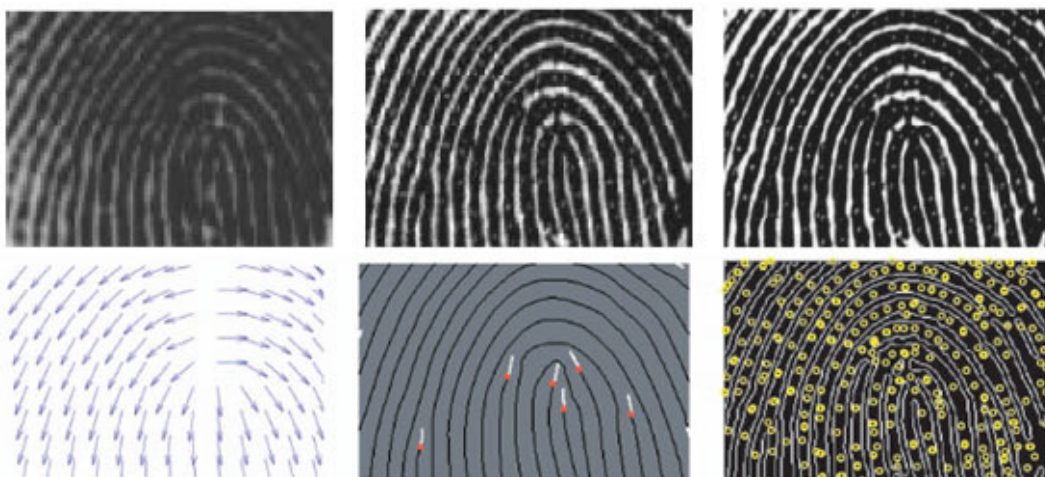
## 2 Feature Extraction

Nun sind wir an der Stelle angelangt, wo wir ein grafisch optimiertes Bild vor uns haben und uns nun an die eigentliche Merkmalerkennung machen können. Es gibt mehrere Typen von Merkmalen, die aus einem Fingerabdruck ermittelt werden können und die dann mehr oder weniger gut mit den gespeicherten Templates aus der Datenbank verglichen werden können. Diese Typen werde ich zunächst einmal vorstellen.

### 2.1 Feature Typen

Abbildung 2 zeigt drei Aufnahmen eines Fingerabdruck in drei verschiedenen Auflösungen. Links ist die Auflösung am geringsten, nämlich 380 dpi. Wie unschwer zu erkennen ist, ist der Abdruck relativ unscharf und man kann keine Details des Abdruckes erkennen. Einzig die Linienführung lässt sich erkennen, welches ausreichend ist, um Level 1 Merkmale zu extrahieren. Level 1 Merkmale sind globale Merkmale, also Merkmale, die sich auf große Flächen bzw. grobe Details beziehen. Dies sind z.B. die Linien, aus denen sich dann ein Orientierungsfeld ermitteln lässt. (Siehe Bild unten links in Abbildung 7). Mit höherer Auflösung steigt natürlich auch die Bildqualität und der Detailgrad. Der Abdruck in der Mitte wurde mit 500 dpi (FBI-Standard) aufgenommen. Hier

erkennt man schon sehr deutlich die Ridges und Valleys, sowie kleinere Details (= Minutien), mit denen die Gabelungen, Enden, o.ä. der Ridges gemeint sind. Diese Merkmale gehören zum Level 2; es sind lokale Merkmale, die sich nur auf einen sehr kleinen Raum beziehen. Es wurden bisher 18 verschiedene Minutientypen entdeckt. Doch die häufigsten und am eindeutigsten zu erkennenden Minutien sind das Ridgeending und die Bifurcation. Ein Fingerabdruck bei dieser Auflösung weist meistens 40-100 Minutien auf, von denen in Deutschland 12 Übereinstimmungen ausreichen, um eine Übereinstimmung der Abdrücke zu bestätigen. Der letzte Typ sind die Level 3 Merkmale. Diese sind erst ab einer Auflösung von 1000 dpi gut sichtbar: es sind die Poren, die auf den Ridges liegen (rechtes Bild). Nur bei einer sehr guten Bildqualität ist es möglich, diese Merkmale sauber zu erkennen und zu extrahieren. Da sich Poren jedoch öffnen und schließen können, wird diese Identifikationsmethode nur optional genutzt. Hauptsächlich werden die Level 1 und Level 2 Merkmale zum Datenbankvergleich genutzt. Dabei können die Level 1 Merkmale als Klassifikation eines Fingerabdruckes dienen (z.B. anhand eines



**Abbildung 7:** v.l.: Level 1 – 380 dpi, Level 2 – 500 dpi, Level 3 – 1000 dpi



## 2 Feature Extraction

Orientierungsvektors) und die Level 2 Merkmale werden letztendlich zum genauen Vergleich heran gezogen. In dieser Ausarbeitung befaße ich mich nun mit zwei Extraktionsalgorithmen für Level 1 Merkmale. Dabei betrachte ich einen sehr neuen Algorithmus (2.2), der sehr genau das Orientierungsfeld sowie Singularitätspunkte (Core und Delta) berechnen kann, sowie einen Algorithmus (2.3), der Helligkeitswerte in fest eingeteilten Segmenten vergleicht.

### 2.2 FOMFE

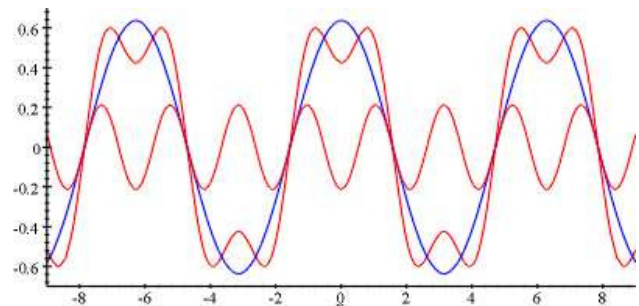
Im April 2007 veröffentlichten Yi Wang, Jianku Hu und Damien Phillips einen Algorithmus für ein Fingerabdruck-Orientierungs-Modell basierend auf einer zweidimensionalen Fouriererweiterung (**F**ingerprint **O**rientation **M**odel Based on **2D** **F**ourier **E**xpansion) [2]. Mit Hilfe dieses Algorithmusses ist es möglich, ein sehr genaues Orientierungsfeld eines Fingerabdruckes zu berechnen. Dadurch ist der Algorithmus laut den Autoren als einziger im Stande, voll automatisch Singular Points (SPs) zu ermitteln, ohne vorherige Kenntnis davon zu haben. SPs sind entweder ein Delta-Point oder ein Core-Point.

Zunächst werde ich die Grundlagen für diesen Algorithmus betrachten. Dabei vernachlässige ich wieder die einzelnen Formel – sie können in der Quelle [2] nachgelesen werden.

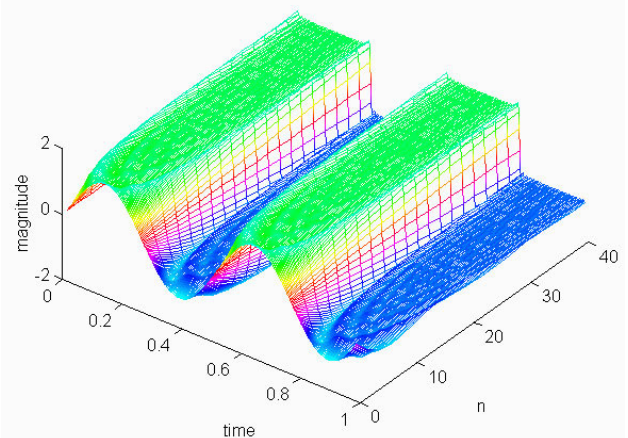
#### 2.2.1 Prinzip

Das Prinzip ist der Versuch einer mathematischen Annäherung des Fingerabdrucks durch eine zweidimensionale Fourierreihe.

Eine Fourierreihe ist eine beliebig fein aufgelöste Überlagerung von Sinus- und Cosinusfunktionen mit unterschiedlichen Frequenzen. Abbildung 8 zeigt eine eindimensionale Überlagerung zweier Wellen (blau mit großer Amplitude und rot



**Abbildung 8:** Überlagerung von Sinusfunktionen bei einem Fourierspektrum



**Abbildung 9:** Dreidimensionale Darstellung eines Fourierspektrums mit zwei Variablen

mit kleiner Amplitude). Das Ergebnis (rot mit großer Amplitude) ähnelt einem leichten Rechteckmuster. Durch beliebig viele Überlagerungen mit unterschiedlichen Amplituden bzw. Wellenlängen kann somit jede Form von Kurve angenähert werden.

Da der Fingerabdruck jedoch zweidimensional ist, muss die Fourierreihe um eine Dimension erweitert werden. Das Ergebnis ist eine dreidimensionale Beschreibung der Struktur, wie es Abbildung 9 für einen Ausschnitt darstellt. (Die dritte Dimension wird beim Fingerabdruckbild die Helligkeitsabstufung sein.)

## 2 Feature Extraction

### 2.2.2 Anwendung

Der eigentliche Durchlauf erfolgt in zwei Schritten. Der erste Schritt ist die Trainingsphase. Hier werden aufgrund einer groben Analyse des Graubildes zwei komplexe Koeffizientenmatrizen für die Fourierexpansion berechnet. Einmal für den Sinus- und einmal für den Cosinusanteil. Dieser aufwendige Schritt benötigt zwar etwas Zeit, jedoch lassen sich diese Matrizen wiederverwenden.

Im zweiten Schritt wird mit Hilfe der berechneten Matrizen das Orientierungsfeld berechnet. Aus Gründen der guten Verarbeitbarkeit werden die extrahierten Informationen in eine neue zweidimensionale Vektormatrix gespeichert, die jeweils den Sinus- und Cosinusanteil der eigentlichen Orientierungsvektoren repräsentiert. Damit kann nun für jeden einzelnen Pixel eine *charakteristische Matrix A* ermittelt werden. Anhand dieser Matrix lassen sich nun die SPs lokalisieren:

Ist die Determinante von  $A$  gleich 0, so gibt es entweder keinen kritischen Punkt in der Nachbarschaft des untersuchten Pixels, oder alle Punkte in der Nachbarschaft sind kritisch. Was heißt kritisch? Kritisch ist ein Punkt dann, wenn er die Orientierungsvektoren in seiner Umgebung stark beeinflusst. Das ist zum Beispiel der Fall bei einem Core-Point, denn dort existiert eine große Krümmung im Orientierungsfeld und die einzelnen Orientierungsvektoren haben eine hohe Abweichung zu ihren Nachbarn.

Ist die Determinante von  $A$  negativ, so befindet sich in der Informationen haltenden Matrix einen Sattelpunkt (siehe Abbildung 10 ganz links). Daraus folgt, dass im Ursprungsbild an dieser Stelle Delta-Point liegt.

Ist die Determinante von  $A$  positiv, so befindet sich in der Informationen haltenden Matrix kein Sattelpunkt und im Ursprungsbild an dieser Stelle ein Core-Point.

Zusatz: Die Koeffizientenmatrix lässt sich noch weiter Benutzen: und zwar für die Indexierung von Fingerprints. Weitere Infos gibt es unter [2].

### 2.2.3 Auswertung und Beurteilung

Zur Auswertung ziehe ich die Ergebnisse heran:

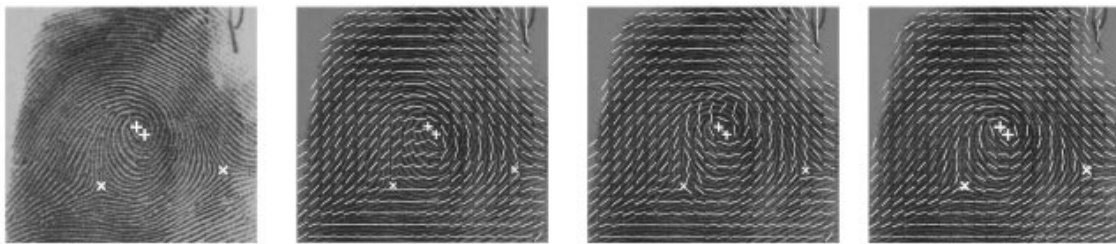
In Abbildung 11 sieht man den Erfolg von FOMFE bei der sehr genauen Ermittlung des Orientierungsfeldes sowie dem Ermitteln der SPs. Die beiden anderen Algorithmen waren nicht sehr erfolgreich – außerdem mussten dort die SPs schon zu Beginn angegeben werden, damit die Algorithmen in der Lage waren, ein mehr oder weniger genaues Orientierungsfeld zu berechnen.

In Abbildung 12 ist der Unterschied noch deutlicher. Hier ist der Fingerabdruck stark verschmutzt. Trotzdem schafft es FOMFE, eine sehr gute Annäherung an die eigentliche Linienführung zu ermitteln. Im Gegensatz dazu produziert der Combialgorithmus ein einziges Wirrwarr sowie lauter falsche SPs.

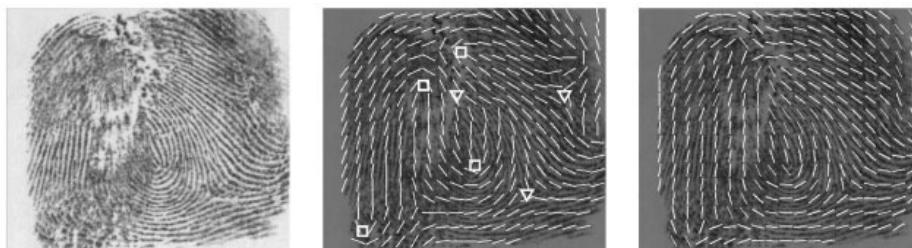


Abbildung 10: Vektorfelder (1) Sattel → (2) Delta, (3) und (5) kein Sattelle → (4) und (6) Core

## 2 Feature Extraction



**Abbildung 11:** v.l. (1) Fingerabdruck mit Markierung von Core-Points (+) und Delta-Points(x) von einem menschlichen Experten, (2) Grobes Orientierungsfeld durch Grauwertanalyse, (3) Combialgorithmus – Grauwertanalyse + Punktladungsmodell (hat Probleme, da er nur einen einzigen Punkt verarbeiten kann – daher Fehler im Orientierungsfeld in der Mitte), (4) FOMFE



**Abbildung 12:** v.l. (1) verschmutzter Fingerabdruck, (2) Combialgorithmus – produziert Wirrwarr, (3) FOMFE

Der Erfolg spricht für sich.

Um es noch ein wenig genauer zu beurteilen, werfen wir einen Blick auf ein ROC-Diagramm:

Dieses Diagramm stellt die FRR der FAR gegenüber. Gegenüber dem groben Erkennungsalgorithmus (Coarse) ist die Combi-I Versoin um die Hälfte besser. Doch FOMFE lässt alles hinter sich – er ist eindeutig die Nummer eins in diesem Vergleich! Doch woran liegt das? Durch die genaue Bestimmung des Orientierungsfeldes lassen sich die SPs sehr genau bestimmen. Dadurch ist es möglich, die Templates beim Matching genauer aneinander auszurichten. Somit können Minutien genauer verglichen werden, da eine Verschiebung zwischen einem zusammengehörenden Minutienpaar in verschiedenen Bildern jetzt nur noch durch unterschiedlich festen Druck des Fingers auf den Sensor entstehen können und nicht zusätzlich

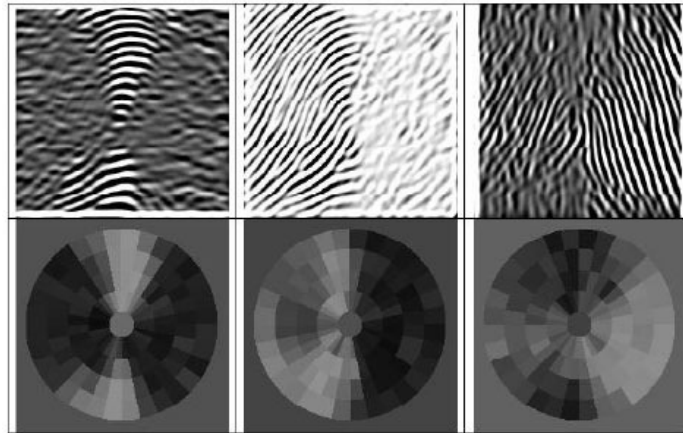
noch durch ungenaue Ausrichtung der beiden Bilder aneinander.

### 2.3 Gaborfilter

Ein anderer Ansatz zur Feature Extraction von globalen Merkmalen ist der Gaborfilter.

Die Voraussetzung für den Erfolg des Filters ist die ermittelte Position des Core-Punktes des Abdrucks. Denn dieser Punkt wird als Zentrum einer Schablone in Form einer Dartscheibe gewählt (siehe Abb. 13), die aus 16 Stücken mit jeweils fünf Segmenten besteht.

Der Gaborfilter an sich ist ein richtungssensitiver Bandpassfilter, der auf kleine Bildausschnitte angewandt werden kann. Er basiert auf einer Fouriertransformation, die jedoch nur in einem gewissen Bereich, einem Fenster angewandt wird. In diesem Fall wird als Fensterfunktion die Gaußfunktion genutzt. Somit ist es möglich, gegenüber



**Abbildung 13:** 3 Gaborfilter – oben die überlagerten Wellenmuster, unten die Auswertung der Übereinstimmung in Graustufen

der Fourieranalyse, die nur globale Merkmale beschreiben kann, lokale Merkmale zu beschreiben, denn die Fourieranalyse wird nur in einem kleinen Fenster durchgeführt. Weiterhin benötigt der Filter zwei Angaben: 1. die Richtung und 2. die Frequenz.

### 2.3.1 Anwenden des Filters

Um nun die Merkmale zu extrahieren, wird dieser Filter wie folgt angewandt: die beiden Parameter für den Filter werden zuerst festgelegt: Richtung ist z.B. Norden ( $0^\circ$ ) und die Frequenz eine feste Wellenlänge, die empirisch ermittelt wurde. Für jeden der 80 Segmente der Dartscheibenschablone wird nun der Filter angewendet. Das Resultat ist entweder eine Übereinstimmung von erzeugtem Wellen- und Rillenmuster oder aber ein unklares Muster. Die Übereinstimmung wird nun in Graustufen dargestellt, wobei ein heller Wert einer hohen Übereinstimmung entspricht. Nun wird der Richtungsparameter für den Filter angepasst: er wird um  $22.5^\circ$  gedreht, was genau einem Kuchenstück der Schablone entspricht. Anschließend wird wieder der Filter angewendet und die Übereinstimmungen von erzeugtem Wellenmuster mit dem Rillenmuster verglichen. Nach acht Drehungen hat der

Filter alle Richtungen abgedeckt und es stehen nun  $8 \times 80$  Grauwerte als extrahiertes Merkmal zur Verfügung, die anschließend zum Vergleich mit der Datenbank genutzt werden können.

## 2 Feature Extraction

### 2.3.2 Auswertung und Beurteilung

Wie in Abbildung 14 zu erkennen ist, liegt die Genuine Accept Rate (GAR) für diesen Algorithmus über 90% schon bei sehr geringen False Accept Rates (FAR) von unter 3%. Dem gegenüber steht ein etwas schlechteres Resultat bei den lokalen Minutien. Da Minutien sehr klein sind, können diese durch z.B.

unterschiedlich fest Aufdrücken des Fingers auf den Sensor deutlich mehr verzerrt werden, wie eher globale Merkmale. So liegt GAR beim Minutienbasierten Matching deutlich unter 90% bei einer FAR von 10%.

Festzuhalten ist, dass trotz Level 1 Merkmal (eher grobe Auflösung) ein gutes Matching machbar ist.

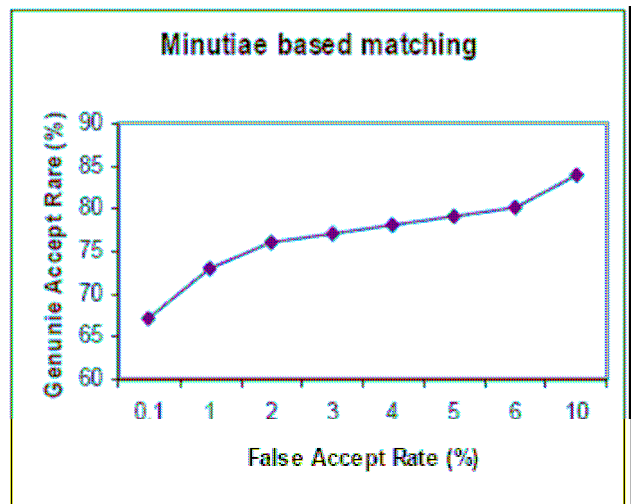
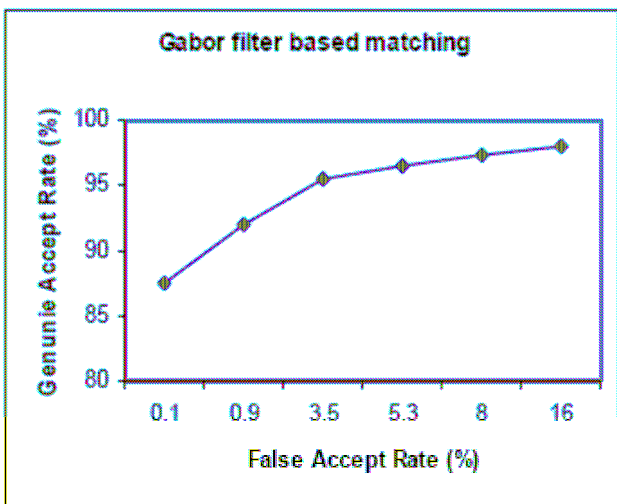


Abbildung 14: ROC-Diagramme für Matchingvergleich von Gaborfilter und Minutien

## 3 Fazit und Ausblick

---

Wie man sieht, bieten sich viele Möglichkeiten, um den Erkennungsprozess zu optimieren – sei es aus zeittechnischen Gründen oder genauere Feature Extraction. Dabei sind einfache Bildbearbeitungsoperationen genau so wichtig wie komplexe mathematische Modelle. Jedoch bleibt die Feature Extraction weitem schwierig – denn das Verständnis und die Fähigkeit, die ein menschlicher Experte vorweisen kann, lässt sich (noch) nicht durch Computerprogramme ersetzen. Die Forschung hat natürlich schon einiges auf dem Gebiet der Fingerabdruckererkennung geleistet, doch

sie ist noch lange nicht am Ende. Optimierungsbedarf besteht zu jeder Zeit. Durch den aktuellen Stand der Technik gibt es auch aus hardwaretechnischer Sicht Neuerungen: bessere Bildauflösungen durch besser aufgelöste Kamerachips bzw. Aufnahmeverfahren erlauben eine höhere Bildqualität und detailliertere Merkmalsbetrachtung (man denke an Level 3 Merkmale, für die es Kamerasensoren gibt, die weit über 1000 dpi auflösen können). Vielleicht gelingt es uns in der Projektgruppe ja, ein wenig an der Performanceschraube zu drehen und ein repräsentatives Ergebnis zu erzielen.

- [1] Lin Hong, Yifei Wan, Anil Jain:  
*Fingerprint Image Enhancement: Algorithm and Performance Evaluation*  
IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE  
INTELLIGENCE, VOL. 20, NO. 8, AUGUST 1998  
pp. 777 - 789
- [2] Yi Wang, Jiankun Hu, Damien Phillips:  
*A Fingerprint Orientation Model Based on 2D Fourier Expansion (FOMFE) and Its  
Application to Singular-Point Detection and Fingerprint Indexing*  
IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE  
INTELLIGENCE, VOL. 29, NO. 4, APRIL 2007  
pp. 573 - 585
- [3] Andreas Uhl:  
Skript zur Vorlesung „*Bildbearbeitung: Biometrische Verfahren*“  
Universität Salzburg, SS 2006
- [4] Muhammad Umer Munir, Dr. Muhammad Younas Javed:  
*Fingerprint Matching using Gabor Filters*, 2004

### *Allgemein + Bilder*

- [5] <http://www.leinroden.de/dpg2001/anzeige1.html>
- [6] <http://www.stanford.edu/~phartke/UDPages/fourier/index.htm>
- [7] Anil K. Jain, Yi Chen, Meltem Demirkus:  
*Pores and Ridges: High-Resolution Fingerprint Matching Using Level 3 Features*  
IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE  
INTELLIGENCE, VOL. 29, NO. 1, JANUARY 2007  
pp. 15 - 27