

Fingerprints II

Einen ersten Einblick in Fingerabdrücke konnte bereits durch den Vortrag *Fingerprints I* vermittelt werden. In unserem Vortrag stellen wir vier verschiedene Verfahren zur Fingerabdruck-Verifikation vor.

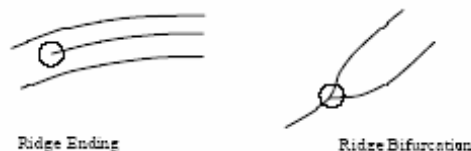
Extraktion von Minutien aus Binärbildern

In diesem Abschnitt wird ein On-Line Fingerabdruck-Verifikationssystem vorgestellt, welches die Aufgabe hat ein Bild eines Fingerabdrucks zu erstellen und dieses mit anderen in einer Datenbank gespeicherten zu vergleichen und zwar in Echtzeit. Dieser Prozess ist in zwei Phasen unterteilt:

- Off-Line Phase:** In dieser Phase wird ein Bild oder mehrere Bilder eines Fingerabdrucks erstellt und vom Merkmalerkennungsmodul weiterverarbeitet. Die erkannten Merkmale werden dann als Vorlage einer Datenbank abgelegt.
- On-Line Phase:** Es wird wieder ein Bild eines Fingerabdrucks erstellt, die Minutien extrahiert und an ein *Matching*-Modul weitergeleitet, welches die Minutien mit den Vorlagen in der Datenbank vergleicht.

1. Minutien-Extraktion

Die meisten Verfahren zur Minutien-Extraktion beschränken sich auf nur zwei Arten von Minutien, die Rillenendung und die Rillengabelung. Bevor die Minutien extrahiert werden können, muss das Bild noch etwas aufbereitet werden, um die Minutien-Detektion überhaupt durchführen zu können. Dazu sind die nun folgenden Schritte notwendig.



Bestimmung des Orientierungsfeldes

Das Bild des Fingerabdrucks wird in Blöcke der Größe $W \times W$ eingeteilt. Dann wird in jedem Pixel eines Blocks der Gradient G_x und G_y berechnet, diese dienen zur Bestimmung der Orientierung eines ganzen Blockes.

Rillendetektion

Da die Grauwerte der Rillen ihr lokales Maximum entlang der Richtung der Normalen der lokalen Rillen erreichen, können dadurch Pixel die zu den Rillen gehören, gefunden werden. Zuerst wird das Bild mit Hilfe zweier Masken gefaltet. Diese Masken sind in der Lage, das lokale Maximum entlang der Normalen zu finden. Wenn die Grauwerte eines Pixels (x,y) des gefalteten Bildes größer als ein Schwellwert ist, so wird dieser Pixel als Teil einer Rille angesehen. Die Pixel, die als Teil einer Rille erkannt wurden erhalten den Wert 1 und die anderen den Wert 0 (\Rightarrow Binärbild).

Ausdünnung

Die gefundenen Rillen werden auf die Breite von einem Pixel verdünnt.

Minutien-Detektion

Bei einem ausgedünnten Bild ist Minutien-Detektion sehr trivial. Man untersucht zu jedem Punkt (x,y) seine acht Nachbarn, wenn die Summe der Nachbarn eins ist, liegt eine Rillenendung vor, sollte die Summe größer als zwei sein, handelt sich hierbei um eine Rillen-Gabelung. Um jedoch eine fehlerhafte Erkennung von Minutien zu verringern, wird das Bild noch einmal gefiltert, um unterbrochene Rillen (*broken ridges*) zu verbinden und kurze Rillen (*spikes*), die kürzer sind als ein Schwellwert, zu entfernen.

Zu jeder Minutie werden dann die folgenden Daten gespeichert:

- x/y-Koordinaten
- Richtung der Rille an der Stelle
- dazugehörige Rille

2. Minutien-Matching

Das hier vorgestellte Verfahren versucht Übereinstimmungen zu finden, indem es eine „Erkennung durch Ausrichtung“ (*recognition by alignment*) durchführt. Der *Matching*-Algorithmus wird dabei in 2 Abschnitte unterteilt:

- a) **Alignment Stage:** Transformationen, wie Verschiebung, Rotation und Größenänderung werden geschätzt und mit der Vorlage ausgerichtet.
- b) **Matching Stage:** Die Minutien der Eingabe und der Vorlage werden in Polygone in Polarkoordinatendarstellung umgewandelt. Mit Hilfe eines String *Matching* Algorithmus wird dann versucht eine Übereinstimmung der Polygone zu finden. Überführung eines Strings in einen anderen, die Anzahl der dabei durchgeführten Schritte (Einfügen, Löschen, Vertauschen) wird als „*edit distance*“ bezeichnet. Die „*edit distance*“ als Grad der Ähnlichkeit betrachtet, denn je höher die „*edit distance*“ ist, desto verschiedener sind die beiden Fingerabdrücke.

Wird während des *Matching* Prozesses eine nicht ganz exakte Übereinstimmung gefunden, wird geprüft, ob sich diese noch innerhalb einer viereckigen Region befindet. Die Größe der viereckigen Region wird zuvor mit Hilfe der „*edit distance*“ berechnet.

Extraktion von Minutien direkt aus Graustufenbildern

Andere Verfahren wandeln vor der Minutien-Erkennung das Eingabebild in ein Binärbild (2 Bit) um (siehe oben), dadurch gehen viele nützliche Informationen verloren. Auch ist dieser Umwandlungsprozess mit anschließender Verdünnung (*thinning*) sehr zeitaufwendig. Liegen die Eingabebilder zudem noch in schlechter Qualität vor, so sind die Ergebnisse der anderen Verfahren unzureichend.

Dieses Verfahren führt die Minutien-Erkennung auf einem Graustufenbild durch, indem es den Rillenlinien folgt. Mathematisch gesehen ist eine Rillenlinie eine Punktmenge, die lokale Maxima entlang einer Richtung bilden. Wenn man die Maxima aneinander reiht, erhält man eine polygonale Annäherung der originalen Rillenlinie.

Rillenverfolgung:

An einem lokalen Maximum einer Rillenlinie wird die Richtung der Rille (Tangentialrichtung) berechnet. Nun geht der Algorithmus μ Pixel in Richtung der Rille weiter, bestimmt wieder das lokale Maximum und die Tangentialrichtung. Dies wird solange fortgesetzt bis ein Stop-Kriterium eintritt. Es wird natürlich auch vom Startpunkt aus die Rille in entgegengesetzt der Startrichtung verfolgt.

Stop-Kriterien:

- Verlassen des zu untersuchenden Bereichs
- Ende einer Rille
- Schnittpunkte
- Zu starke Krümmung (deutet auf Fehler bei der Rillenverfolgung hin)

Minutien-Detektion

Die Detektion der Minutien basiert auf der Rillenverfolgung, dabei werden die gefundenen Rillenlinien und Minutien in eine Hilfsabbildung übertragen.

Nach Abschluss der Minutien-Detektion wurden die Rillenlinien auf die Hilfsabbildung übertragen und zudem die Minutien extrahiert. Der Algorithmus erkennt automatisch falsch erkannte Minutien und korrigiert diese. Im direkten Vergleich mit Verfahren, die auf Umwandlung in ein Binärbild (*binarization*) und Ausdünnung (*thinning*) beruhen, schneidet dieses Verfahren weit besser ab. Es hat eine geringe Rate von falsch erkannten Minutien und ist bei der Minutien-Extraktion im Durchschnitt zwei- bis achtmal schneller.

Korrelationsbasiertes Fingerabdruck-Identifikationssystem

Dieses System extrahiert keine Minutien aus den Bildern, sondern arbeitet direkt mit den Grauwerten des Bildes indem es aus dem Primär-Fingerabdruck kleine geeignete Regionen (*templates*) extrahiert, diese im Sekundärfingerabdruck zu lokalisieren versucht und dann die relativen Positionen zueinander vergleicht.

Template-Bestimmung im Primärfingerabdruck:

Im Primärfingerabdruck werden rechteckige Felder bestimmt, welche folgende Eigenschaften erfüllen sollen:

- Die Region sollte im Primärfingerabdruck einzigartig sein, damit es später nicht zu fehlerhaften Detektionen im Sekundärfingerabdruck kommt.
- Die Größe der *Templates* sollte weder zu groß noch zu klein sein: zu große *Templates* haben mit den Verzerrungen zu kämpfen, zu kleine bergen die Gefahr der Mehrfachdetektion

Um diese Eigenschaften zu erfüllen, ergeben sich folgende Arten die *Template*-Positionen zu bestimmen:

- Minutienbasiert: man bestimmt die Minutien-Positionen im Fingerabdruck und legt um diese die *Templates* fest. Vorteil: man übernimmt alle Nachteile der minutienbasierten Systeme (fehlerhafte Erkennung von Minutien)
- Gradientenrichtungs-basiert: man bestimmt zuerst diejenigen Gebiete im Fingerabdruck, in denen die Rillenrichtungen stark schwanken (*low coherence*) und legt diese als *Template*-Positionen fest. Vorteil: verrauschte Bilder haben ebenfalls niedrige Kohärenz-Werte.
- Korrelationsbasiert: man legt diejenigen Regionen als *Template* fest, welche über den gesamten Primärfingerabdruck die niedrigsten Korrelationswerte besitzen. Leider wird bei diesem Ansatz viel Rechenzeit benötigt. Um dies zu umgehen, bedient man sich zuerst einer der oben erwähnten Methoden um Anwärter für *Template*-Positionen zu erhalten und führt die Korrelationsmethode nur noch mit den Anwärtern durch.

Template Matching

Mittels Korrelation wird für jedes *Template* des Primärfingerabdrucks diejenige Region im Sekundärfingerabdruck bestimmt, die dem *Template* am ähnlichsten ist. Dadurch erhält man für jedes *Template* im Primärfingerabdruck genau eine entsprechende Position im Sekundärfingerabdruck.

Klassifikation der Templatepositionen

Dieser Abschnitt ist in zwei Phasen unterteilt: die elementaren Entscheidungen und die globale Entscheidung.

- Elementare Entscheidungen:
Für alle *Template*-Paarungen im Primärfingerabdruck werden die relativen Abstände und Winkel (*Relative Template Position* = RTP) zueinander ausgerechnet und mit den Abständen und Winkel der Paarungen im Sekundärfingerabdruck verglichen. Sind die beiden Ergebnisse (fast) gleich so wird die verglichene RTP als „passend“ bzw. als „nicht passend“ gekennzeichnet.
- Globale Entscheidung:
Für die globale Entscheidung wird zuerst ein Referenz-*Template* bestimmt, welches die meisten passenden RTPs besitzt. Ein Fingerabdruck wird als „passend“ bewertet, wenn das Referenz-*Template* mehr als kT „passende“ RTP besitzt. Die Schranke kT hängt von den erwünschten FAR bzw. TRR ab und wird in diesem System mittels bedingter Wahrscheinlichkeit bestimmt.

Ein neuronales Netzwerk zur Fingerabdruckererkennung

Dieses System setzt neben einem korrelationsbasierten Ansatz ein neuronales Netzwerk um zwei Fingerabdrücke zu verifizieren.

1. Die Fingerabdruckdatenbank des System

Die Datenbank enthält 200 Bilder von denen 100 Bilder von 20 Fingern stammen (von einem Finger wurden 5 unterschiedliche Bilder hinterlegt). Diese Bilder stellen die Datensätze dar, mit denen das neuronale Netz trainiert wurde. Die Bilder sind 512*464 Pixel groß und haben 256 Graustufen.

2. Vorverarbeitung

- Tiefpassfilterung
Rauschen wird durch einfachen Boxfilter entfernt
- Bestimmung des Arbeitsbereichs
 - mittels Kantenerkennung wird eine Box um die Fingerabdrücke gelegt
 - Bestimmung der Boxmitte bestimmt
 - Extraktion einer 65*65 Pixel großen (Arbeits-)Region aus Referenzbild
 - Extraktion einer 105*105 Pixel großen (Fenster-)Region aus Testbild
- Ausrichtung
 - Korrelation (Bestimmung der Position in der Arbeitsregion in der Fensterregion am besten zueinander passen)
 - benötigt sehr viel CPU-Zeit
- Kompression und Normalisierung
 - die 64*64 Pixel großen Regionen werden mittels Gaußfilter geglättet und auf 32*32 Pixel Regionen reduziert (Breite entspricht ungefähr 10 Rillen im Fingerabdruck)
 - Normalisierung der Grauwerte auf Fließkommawerte von 0 bis 1.

3. Neuronale Entscheidungsphase

Als Eingabe dienen die beiden 32*32Pixel großen Regionen. Die Ausgabe wird in Form einer Wahrscheinlichkeit p ausgegeben, welche angibt, wie gut die beiden Eingabebilder zueinander passen.

Faltung mit erlernten Filtern

Die beiden Eingabebilder werden mit zwei 7*7 Pixel großen Filtermatrizen so gefaltet, dass sich für jedes Bild zwei 6*6 Felder große Resultatarrays ergeben. Die Gewichte in den Filtern entsprechen den Lernparametern (49*2 Parameter).

Abschließend berechnet man für jeden Filter die quadratische Differenz der beiden Faltungsergebnisse für die Eingangsbilder.

Entscheidung

Aufgrund der ermittelten quadratischen Differenzen berechnet nun das neuronale Netz die Wahrscheinlichkeit, ob die beiden Fingerabdrücke zueinander passen. In der Trainingsphase wird das berechnete Ergebnis mit dem Sollwert verglichen, um dann mit Hilfe der Differenz der beiden Werte die Gewichte in den Filtern und im neuronalen Netz anzupassen.

Ergebnisse beim Fingerprint Verification Contest 2000

Methode	Platz (von 11)	Gesamtzeit (Sekunden)	EER Fehlerrate
Bestes Verfahren (minutienbasiert)	1	4,3	1,7%
Minutienbasiertes Verfahren (Grau)	3	0,4	5,1%
Korrelationsbasierter Ansatz	7	13,1	15,2%

Schlussfolgerungen:

Minutienbasierte Systeme sind deutlich schneller als korrelationsbasierte. Sie haben geringere Fehlerraten, wenn die Fingerabdrücke nicht verrauscht sind. Der obige Wert von 15% Fehlerrate bei dem korrelationsbasierten Ansatz ergibt sich zudem aus der fehlenden Unterstützung bei gedrehten Eingangsbildern.

Quellen

- [1] Jain, Houg, Bolle, „On-line Fingerprint Verification“, IEEE Transactions On Pattern Analysis And Machine Intelligence, Vol. 19, No. 4, January 1997
- [2] Maio, Maltoni, „Direct Gray-Scale Minutiae Detection In Fingerprints“, IEEE Transactions On Pattern Analysis And Machine Intelligence, Vol. 19, No. 1, January 1997
- [3] Bazen, Verwaaijen, Gerez, Veelenturf, van der Zwaag, „A Correlation-Based Fingerprint Verification System“, ProRISC 2000 Workshop On Circuits, Systems and Signal Processing, Veldhoven, The Netherlands, November 2000
- [4] Baldi, Chauvin, „Neural Networks For Fingerprint Recognition“, Neural Computation 5, 402– 418 (1993) © 1993 Massachusetts Institute Of Technology
- [5] Maio, Maltoni, Cappelli, Wayman, Jain, „FVC2000: Fingerprint Verification Competition“, IEEE Transactions On Pattern Analysis And Machine Intelligence, Vol. 24, No. 3, March 2002
- [6] Maltoni, „Chapter 4, Fingerprint Matching“, Handbook Of Fingerprint Recognition, Springer, 2003