

Kapitel 2

Vorverarbeitung

Unter technischen Aspekten, handelt es sich bei der Handschrift um ein hoch-komplexes Signal zur Informationsübertragung. Jedes handgeschriebene Wort, oder auch nur ein handgeschriebener Buchstabe sind Unikate. Die genaue manuelle Reproduktion, selbst der eigenen Schrift, scheint ausgeschlossen. Dies fällt um so mehr auf, je eingehender die Handschrift mit maschinellen Methoden bearbeitet wird. Die Faktoren, die schließlich das Gesamtergebnis beeinflussen, sind dabei außerordentlich vielfältig.

Die größten Unterschiede sind sicherlich bei variierenden Schreibern auszumachen. Jede Handschrift verfügt über gewisse Eigenschaften, die sie unverwechselbar machen.

Doch auch unter der Voraussetzung, dass die von einem Handschrifterkennungssystem erfassten Handschriftdaten von nur einem Schreiber produziert werden, ergeben sich zahlreiche Randbedingungen, die auf das Resultat einwirken. Dies können sein:

- Der Zeitpunkt der Produktion. Insbesondere bei jüngeren Menschen kann sich das Schriftbild über einen längeren Zeitraum stark verändern.
- Die persönliche Verfassung des Schreibers. Eine verletzte Schreibhand z. B. kann großen Einfluß auf die Handschrift haben.
- Der Betreff und die Zielperson. Das Schriftbild einer für eigene Zwecke gedachten schnellen Notiz weicht möglicherweise ab von dem Schriftbild in handgeschriebenen Nachrichten für andere.
- Schreibgerät und Unterlage beeinflussen das Schriftbild.
- Der Kontext. Als Kontext läßt sich sowohl die weiter als auch enger gefasste Umgebung der aktuellen Schreibposition definieren. Die weitere Umgebung, und damit verbunden auch die Platzverhältnisse, können sich auf die Schrift auswirken. Im engeren Sinne kann eine Kontextabhängigkeit auch durch benachbarte Buchstaben bedingt sein.

- Datenerfassungsgerät. Als technische Komponente beeinflusst das benutzte Erfassungsgerät als unmittelbare Schnittstelle zwischen Mensch und Maschine wesentlich die Qualität der erfassten und zu verarbeitenden Daten (Abtastverfahren, Abtastrate, Auflösung).

Auf eine automatische Schrifterkennung wirken sich Veränderungen der genannten Einflußgrößen natürlich störend aus. Ein 'a' sollte stets als ein solches erkannt werden - und dies möglichst robust gegenüber äußeren Einflüssen. Der Erkennungsalgorithmus sollte so unabhängig wie möglich von den genannten Randbedingungen funktionieren. Die Eliminierung, oder zumindest die Minimierung irrelevanter Eigenschaften ist die Aufgabe der Vorverarbeitung. Die Vorverarbeitung betrifft in diesem Zusammenhang die in den folgenden Abschnitten beschriebene Neuabtastung und die Neigungs- und Größen-Normalisierung der Schrift.

2.1 Neuabtastung

Mit der Online-Erfassung von Handschriftdaten und der damit gegebenen zeitlichen Abfolge der Abtastvektoren ergeben sich einerseits Vorteile, wie beispielsweise die Unterstützung der Zeichensegmentierung durch den Stiftdruck, andererseits wird mit der Online-Abtastung auch unerwünschte und irrelevante Information erfasst.

Durch die Verwendung von WACOM-Boards für die Datenerfassung sind dies in expliziter Form zum einen die Stiftführung bei negativem Stiftdruck - also bei abgehobener Mine. Bei dem Verlauf dieser virtuellen Segmente (obere Zeile in Abb. 2.2) kann davon ausgegangen werden, dass diese keinen Beitrag für eine Unterscheidung verschiedener Buchstaben liefern.

Zum anderen ist durch die zeitlich äquidistante Abtastung der Rohdaten implizit die Schreibgeschwindigkeit in den Rohdaten enthalten. Die zeitlich konstante Abtastrate resultiert bei zwei optisch identischen Kurvenverläufen die mit unterschiedlicher Stiftgeschwindigkeit erzeugt wurden in deutlich abweichenden Abtastsequenzen.

Um diese unerwünschten Effekte auszuschalten, bietet sich eine Neuabtastung (NA) der Rohdaten an [Wey96, Kas95]. Die Neuabtastung überführt die Folge von Abtastpunkten, die zu $n \cdot \Delta T$ zeitlich äquidistant abgetastet wurden in eine Folge räumlich äquidistanter Abtastpunkte:

$$\vec{v}(n \cdot \Delta T) = (x_{Anf}, y_{Anf}, x_{End}, y_{End}, \tilde{p})(n \cdot \Delta T) \xrightarrow{NA} (x_{Anf}, y_{Anf}, x_{End}, y_{End}, \tilde{p})(k) = \vec{v}(k) \quad (2.1)$$

Die Abtastvektoren $v(n \cdot \Delta T)$, die durch die abgetasteten Anfangs- und Endpunkte in x - und y -Richtung, wie auch durch den abgetasteten Stiftdruck \tilde{p} gegeben sind, werden mit

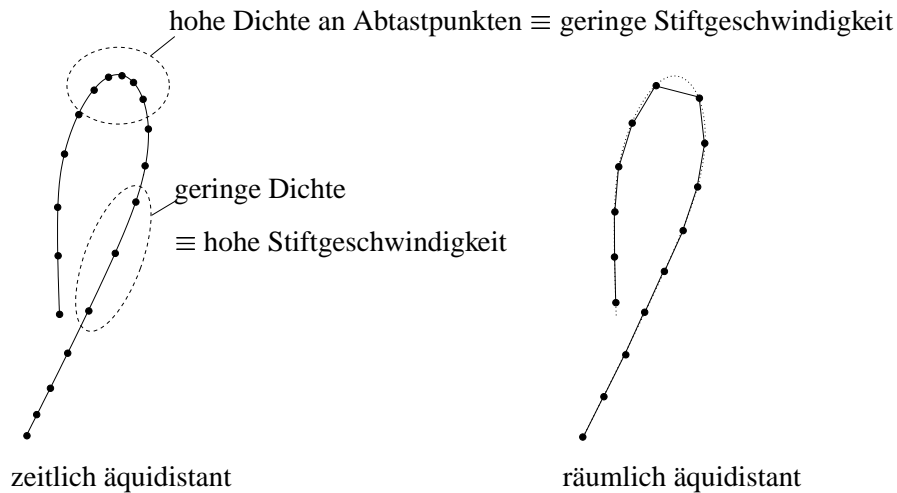


Abbildung 2.1: Neuabtastung - prinzipielle Wirkungsweise

der Neuabtastung in der Form zusammengefasst oder ggf. aufgeteilt, dass der Abstand zwischen Anfangs- und Endpunkten $\sqrt{(x_{End}(k) - x_{Anf}(k))^2 + (y_{End}(k) - y_{Anf}(k))^2}$ der Vektoren $\vec{v}(k)$ einem vorzugebenden, konstanten Wert \hat{l} entspricht. Abb. 2.1 verdeutlicht diese Vorgehensweise. Abb. 2.2 zeigt einige konkrete Beispiele, die mit verschiedenen Abtastvektorklängen $\hat{l} = 2, 5, 10, 20$ erzeugt wurden, zusammen mit der zugrunde liegenden Originalsequenz. Die obere Zeile in Abb. 2.2 gibt die auf oben beschriebene Weise neuabgetasteten Vektorsequenzen wieder. Die dünnen Linien zeigen die ebenfalls erfassten Stiftbewegungen zwischen den Segmenten bei abgehobener Mine.

Zum einen zeigt sich in Abb. 2.2 der z. T. chaotische Verlauf der virtuellen Segmente (bei abgehobenem Stift), zum anderen zeigt sich aber auch, dass bei steigender Abtastvektorklänge die Formeigenschaften der Zeichen doch stark verfälscht werden. Dies gilt um so mehr bei insgesamt eher kleinen Zeichen. Abhilfe für das erste Problem schafft eine lineare Interpolation der virtuellen Segmente mit Abtastvektoren der konstanten Länge \hat{l} .

Zur Lösung des zweiten Problems werden die realen Teilsequenzen (bei aufgesetztem Stift) zunächst nach lokalen Extremwerten in x - und y -Richtung durchsucht. Diese Extremwerte werden dann verwendet, um eine Unterteilung der realen Teilsequenzen in Fragmente vorzunehmen. Schließlich wird die Neuabtastung so durchgeführt, dass die verwendete Abtastvektorklänge für das aktuelle Fragment optimiert wird. D. h. die Anzahl der Abtastvektoren in einem Fragment, und damit deren Länge ist so zu wählen, dass sie der geforderten Solllänge \hat{l} möglichst weit angenähert wird. Die untere Zeile in Abb. 2.2 zeigt dazu einige Beispiele, ebenfalls bei variiertem Abtastvektorklänge. Es zeigt sich klar, dass die chaotisch geformten virtuellen Segmente linearisiert wurden, wie auch die Formerhaltung bei längeren Abtastvektoren verbessert wurde. So läßt sich nicht nur die Konsistenz der Daten verbessern, insgesamt wird mit der Neuabtastung auch eine Reduktion der Datenmenge um das zwei- bis dreifache erreicht.

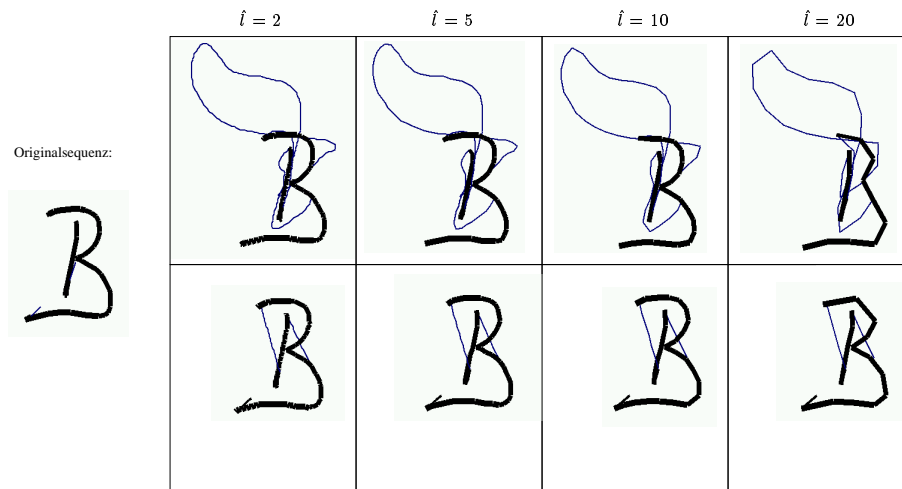


Abbildung 2.2: Neuabtastung mit verschiedenen Vektorlängen

2.2 Normalisierung

Neben der Abgleichung der Stiftgeschwindigkeit und der Linearisierung virtueller Segmente, sollten vor der eigentlichen Modellierung weitere Schritte zur Steigerung der Datenkonsistenz unternommen werden.

Für eine robuste Handschrifterkennung sind aufgrund der hohen Schriftvariabilität zwischen verschiedenen Schreibern große Mengen an Trainingsbeispielen erforderlich um möglichst alle Schrifttypen zu erfassen. Abgesehen von der Stiftgeschwindigkeit läßt sich die Handschrift unter Modellierungsaspekten anhand von vier wesentlichen Merkmalen charakterisieren:

- Schriftgröße,
- Schriftneigung (Slant),
- Zeilenneigung (Skew),
- Die Formeigenschaften der einzelnen Zeichen (Font).

Die Aufgabe der Handschriftnormalisierung ist die Reduktion dieser Freiheitsgrade, um auch mit einer relativ kleinen Trainingsdatenbasis und damit mit weniger komplexen Modellen eine robuste Erkennung zu realisieren. Während die Fontvarianz erst auf der Modellebene aufgelöst werden kann, ist es bereits in frühen Vorverarbeitungsschritten möglich, die Handschrift bezüglich Größe und Neigungswinkel zu normalisieren.

Die Normalisierung der Handschrift ist speziell in schreiberunabhängigen Systemen von Bedeutung. In schreiberabhängigen Systemen werden schreiberspezifische Charakteristika wie Schrifthöhe oder Slant-Winkel auf der Modellebene mittrainiert. Eine Normalisierung ist in

schreiberabhängigen Systemen daher nur erforderlich, wenn vorab bereits klar ist, dass die Testbedingungen von den Trainingsbedingungen abweichen.

2.2.1 Zeilenneigung - Skew

Eine in der Vorverarbeitung korrigierbare Abweichung stellt im allgemeinen auch die Zeilenneigung dar. Zur Korrektur einer Zeilenneigung finden sich in der Literatur zwei prinzipielle Ansätze. Der erste Ansatz basiert auf der Berechnung linearer Regressionsgeraden [Sch95]. Die berechneten Regressionsgeraden approximieren obere und untere Wendepunkte der Trajektorie. Die Steigungen der ermittelten Geraden sind ein Maß für den Winkel der Zeilenneigung. Der zweite Ansatz stützt sich auf die Analyse von Richtungshistogrammen [Sun97] um den Zeilenneigungswinkel zu finden. Ist der Zeilenneigungswinkel θ_0 gefunden, läßt sich die Vektorsequenz durch eine Rotation um den Winkel $-\theta_0$ korrigieren.

Bei einem Abstand $r_v(k)$ der Abtastpunkte $\mathbf{v}(k) = (x, y)(k)$ zu einem frei wählbaren Referenzpunkt und einem Winkel $\varphi_v(k)$ des Radiusvektors zur x -Achse ergibt sich die rotierte Abtastsequenz $\mathbf{v}'(k) = (x'(k), y'(k)) = r_v(k)[\cos(\varphi_v(k) - \theta_0), \sin(\varphi_v(k) - \theta_0)]$. Sinnvoll ist es für die weitere Verarbeitung, den Rotationsradius einzuschränken. Als Referenzpunkt kann darum beispielsweise der Anfangspunkt des 0-ten Abtastvektors $\mathbf{v}_0(0)$ gewählt werden, um den sämtliche Vektoren $\vec{v}(k)$ gedreht werden.

Eine interessante Alternative zu den oben erwähnten Ansätzen bietet sich für die Normalisierung von Offline-Daten mit der Untersuchung von Integralprojektionen [Cot97]. Übertragen auf Online erfasste Handschriftdaten, läßt sich eine Projektion aus den Häufigkeiten der Abtastpunkte in den einzelnen Pixelreihen ermitteln. Betrachtet wird dazu ein hinreichend großer Bereich zwischen y' und y'' in y -Richtung der für die auszuführenden Rotationen genügend Raum bietet. Liegen in der i -ten Pixelreihe n_i Abtastpunkte, so läßt sich die Projektion, bzw. relative Häufigkeit

$$h(y) = \frac{n_y}{\sum_{i=y'}^{y''} n_i} \quad (2.2)$$

für die Pixelzeile y angeben.

Abb. 2.3 zeigt an einem Beispiel den Zusammenhang zwischen dem Zeilenneigungswinkel und den entsprechenden y -Projektionen. Horizontale Ausrichtungen spiegeln sich in Verteilungen mit geringer Varianz und klar erkennbaren Maxima wieder. Bei schief verlaufenden Zeilen sind die Verteilungen der Abtastpunkte deutlich flacher. Als Bewertungsmaß für die unterschiedlichen Verteilungen drängt sich damit die Entropie

$$H = - \sum_{i=y'}^{y''} h(i) \log h(i) \quad (2.3)$$

der Verteilung $h(i)$ auf.

Um schließlich die Zeilenneigung einer unbekanntenen Schriftprobe zu ermitteln, ist die Vek-

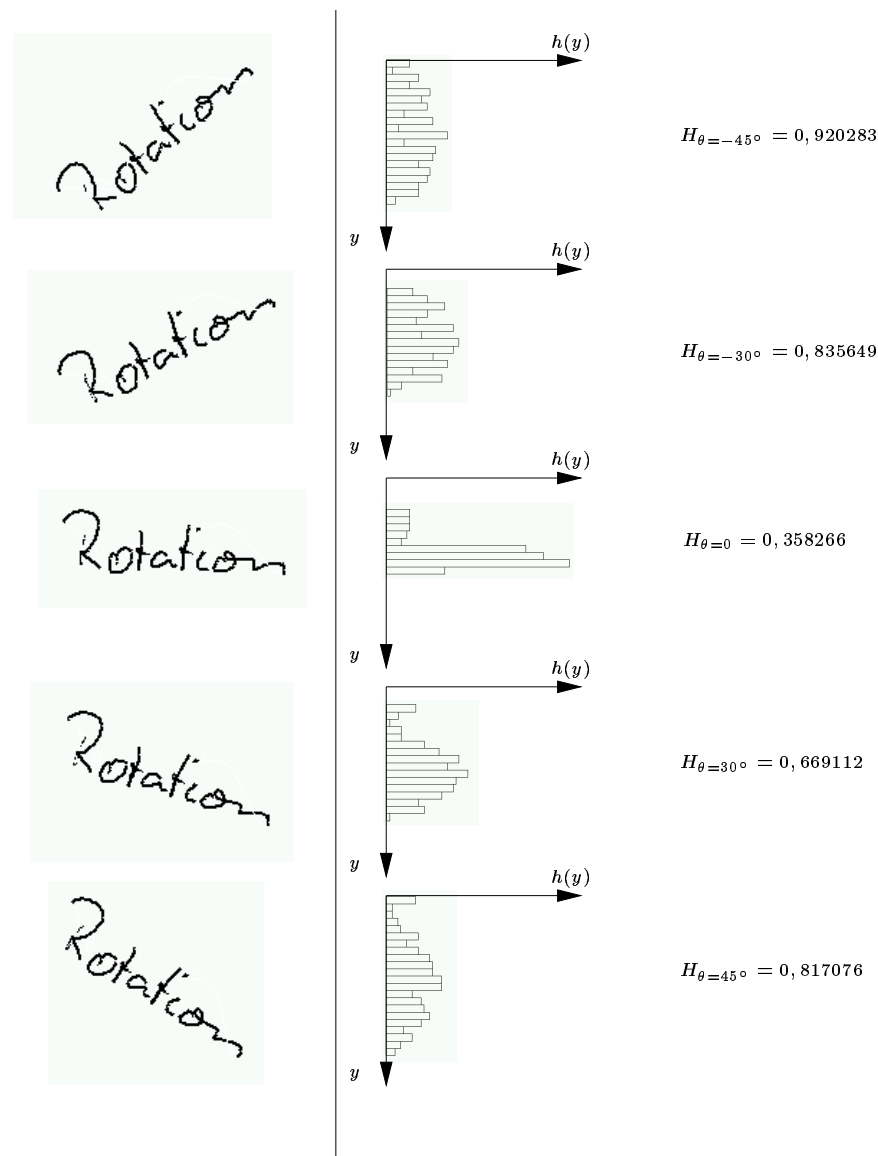


Abbildung 2.3: Zeilenrotation

torfolge in einem bestimmten Intervall jeweils um einen kleinen Betrag $\Delta\theta$ zu drehen. Zu jedem Drehwinkel wird die Projektion ermittelt und die zugehörige Entropie. Abb. 2.4 zeigt dazu das Beispiel einer schiefen Schriftprobe und den Entropieverlauf der y -Projektion in Abhängigkeit des Rotationswinkels. Der Neigungswinkel von ca. -18° lässt sich klar am Minimum des Entropieverlaufes ablesen. Die abschließende Zeilenneigungskorrektur kann durch eine Rotation der Originalsequenz um $+18^{\circ}$ vorgenommen werden. Bei einer hinreichend großen Anzahl von Abtastpunkten erweist sich dieses Verfahren als recht präzise. Es lässt sich dann eine Genauigkeit von $\pm 1^{\circ}$ erreichen.

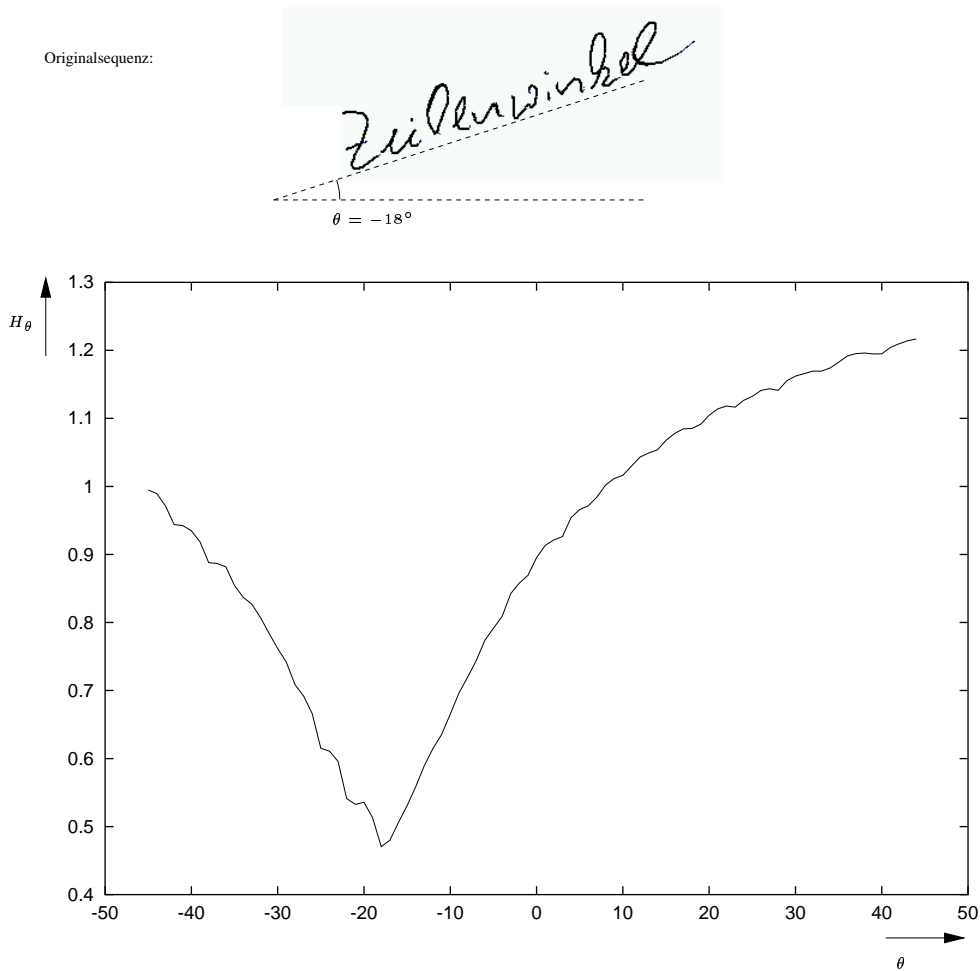


Abbildung 2.4: Entropie der Integralprojektion als Funktion des Zeilenneigungswinkels

2.2.2 Schriftneigung - Slant

Für die Normalisierung der Schriftneigung (Online und Offline) wird bisher fast ausschließlich auf die Analyse von Richtungshistogrammen gesetzt [Sun97]. In Anlehnung daran wird in [Sch95] ein Verfahren zur Slant-Minimierung beschrieben, welches eine mittlere Segmentorientierung verwendet. Die Ergebnisse in [Sch95], sowie eigene Untersuchungen zeigen jedoch, dass eine mittlere Orientierung, oder auch das Maximum der Orientierungsverteilung keine stabile Slant-Abschätzung erlaubt. Ein Grund dafür ist sicherlich, dass die verwendeten Vektorsequenzen, bedingt durch ihre teilweise begrenzten Längen, keine aussagekräftige Stichprobe für das Aufstellen von Richtungshistogrammen darstellen. Auf eine Normalisierung der Schriftneigung muß in der Vorverarbeitung daher oftmals verzichtet werden.

Betrachtet man das Problem der Schriftneigung näher, läßt sich wiederum die Frage stellen, was überhaupt ein eindeutiges Merkmal für aufrechte oder geneigte Schrift darstellt. Dies sind zweifellos die näherungsweise vertikalen Stiftbewegungen bei aufgesetzter Mine (pos. Stiftdruck). Diese Bewegungen entziehen sich allerdings einer direkten Analyse durch Rich-

tungshistogramme.

Eine indirekte Methode wäre wiederum die Projektion der Vektorsequenz. Nun erfolgt die Projektion jedoch auf die x -Achse. Schrift mit exakt vertikaler Ausrichtung der einzelnen Segmente müßte dabei durch Verteilungen auffallen, die besonders bei auftretenden Oberlängen zu hohen Peaks in den Projektionen führen. Geneigte Zeichen hingegen, die teilweise noch benachbarte Buchstaben überdecken, sollten im Gegensatz dazu eher verschmierte Maxima in den Projektionen ausbilden.

Diese Überlegungen werden anhand von Abb. 2.5 bekräftigt. Dort ist der Entropieverlauf in

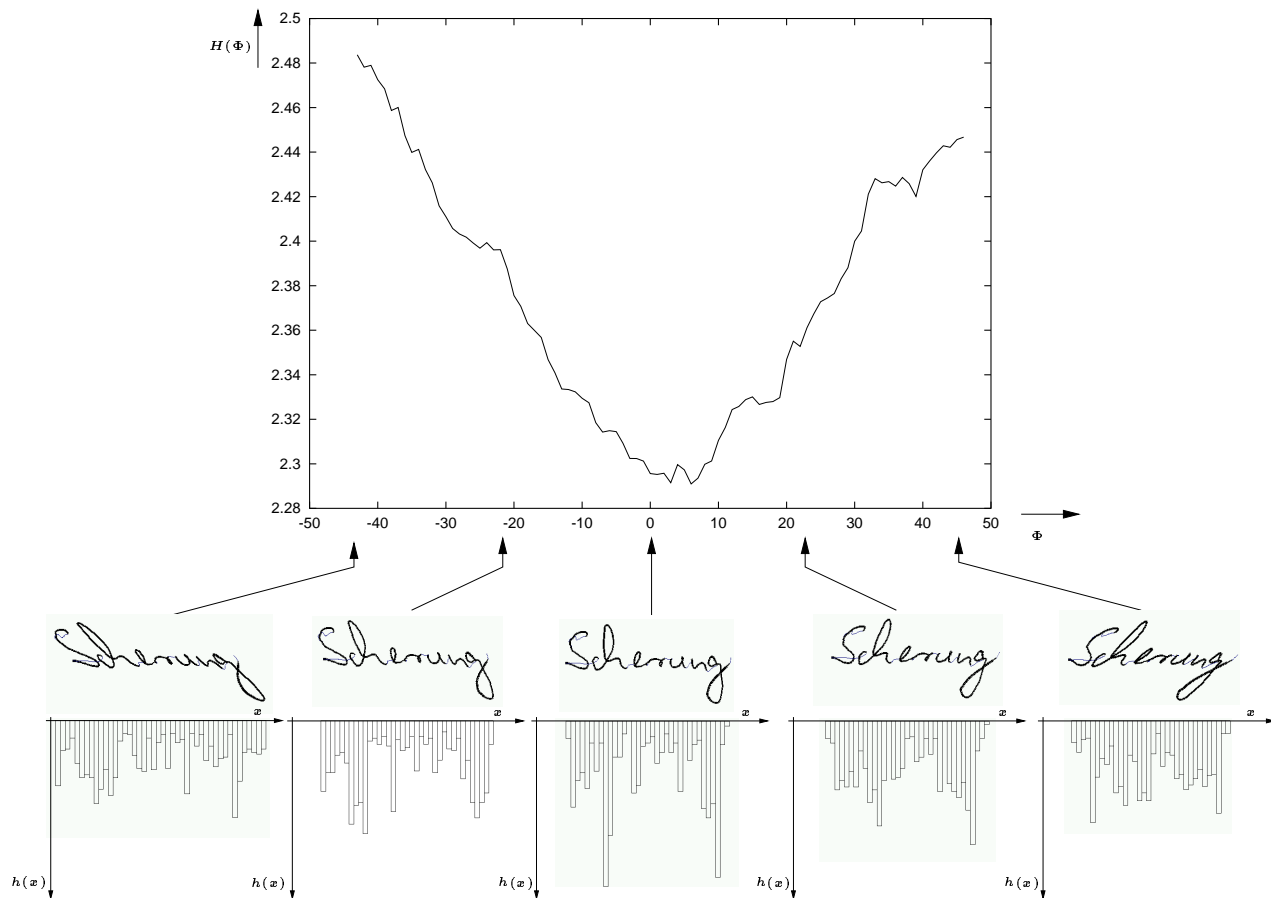


Abbildung 2.5: Entropieverlauf als Funktion des Scherungswinkels

Abhängigkeit des Schriftneigungswinkels gezeigt, wie auch das Schriftbild mit zugehörigen x -Projektionen für einige ausgewählte Neigungswinkel. Es zeigt sich ein Entropieminimum bei vollständig aufgerichteter Schrift. Wenngleich das Minimum in absoluten Werten nicht so deutlich von den Werten bei extremer Neigung abweicht, wie es bei der Zeilenrotation der Fall ist, scheint der Entropieverlauf ein geeignetes Maß für die Erkennung des Schriftneigungswinkels darzustellen.

Ist der Schriftneigungswinkel nun bekannt, so kann durch eine Scherung um diesen Winkel die Vektorsequenz $\vec{v}(k)$ zu $\vec{v}'(k)$ korrigiert werden. Die Scherung wird entsprechend den in

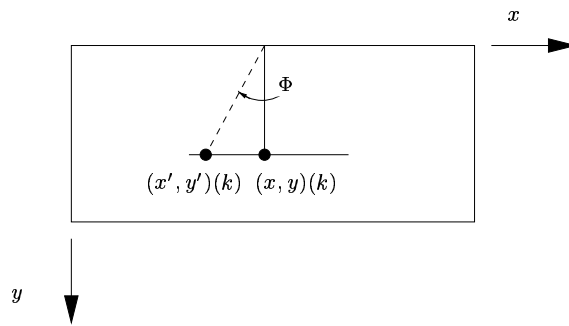


Abbildung 2.6: Vorzeichenkonvention für die Schriftschерung

Abb. 2.6 skizzierten Größen und Vorzeichen durchgeführt.

Von der Scherung bleiben die y -Komponenten der Abtastpunkte unberührt, d. h. es gilt $y'(k) = y(k)$. Die x -Komponenten der Abtastpunkte $x(k)$ werden um $-y(k) \cdot \tan(\Phi)$ korrigiert, sodass sich ein $x'(k) = x(k) - y(k) \tan(\Phi)$ ergibt. Da die Scherung in x -Richtung von der vertikalen Position y abhängt, sollte auch hier statt der x -Achse eine Referenzachse in der nahen Umgebung der Schrift gewählt werden.

In einigen Grenzfällen erweist es sich als schwierig einen exakten Neigungswinkel anzugeben. Dies kann der Fall sein, wenn die Schriftneigung zwischen verschiedenen Wörtern einer Wortfolge, oder sogar innerhalb eines Wortes variiert, oder aber bei sehr kurzen Worten mit entsprechend wenig Abtastpunkten. In solchen Fällen läßt sich beobachten, dass sich weit auseinanderliegende, nahezu gleichwertige Entropieminima einstellen (jeweils untere Kurven in Abb. 2.7). In den in Abb. 2.7 gezeigten Beispielen würde ein solches Nebenminimum sogar zu einer falschen Winkelschätzung führen.

Bei näherer Ursachenforschung fällt dabei auf, dass sich kritische Nebenminima dann bilden, wenn möglichst viele obere Wendepunkte in vertikaler Linie mit unteren Wendepunkten liegen. Dies wiederum liegt darin begründet, dass sich an den Wendepunkten aufgrund der zeitlich äquidistanten Abtastung die Abtastpunkte überdurchschnittlich stark kumulieren (geringere Stiftgeschwindigkeit). Die vertikale Überlagerung oberer und unterer Wendepunkte stellt jedoch nicht zwangsweise den optimalen Fall vollständig aufgerichteter Buchstaben dar. Es ist dabei sogar möglich, dass sich obere Wendepunkte des einen Buchstaben über untere Wendepunkte eines benachbarten Buchstabens schieben. Die Folge sind natürlich unerwünschte starke Schriftneigungen, die u. U. noch drastischer ausfallen als bei den zugrunde liegenden Originaldaten.

Abhilfe für dieses Problem sollte eine vorab durchgeführte Neuabtastung schaffen. Die jeweils oberen Kurven der beiden Beispiele in Abb. 2.7 wurden auf diese Weise erzeugt. Sie scheinen dies zu bestätigen. Kritische Nebenminima treten dort deutlich weniger auf. Darüber hinaus sind die Kurvenverläufe $H(\Phi)$ bei den neuabgetasteten Daten glatter als bei den Rohdaten.

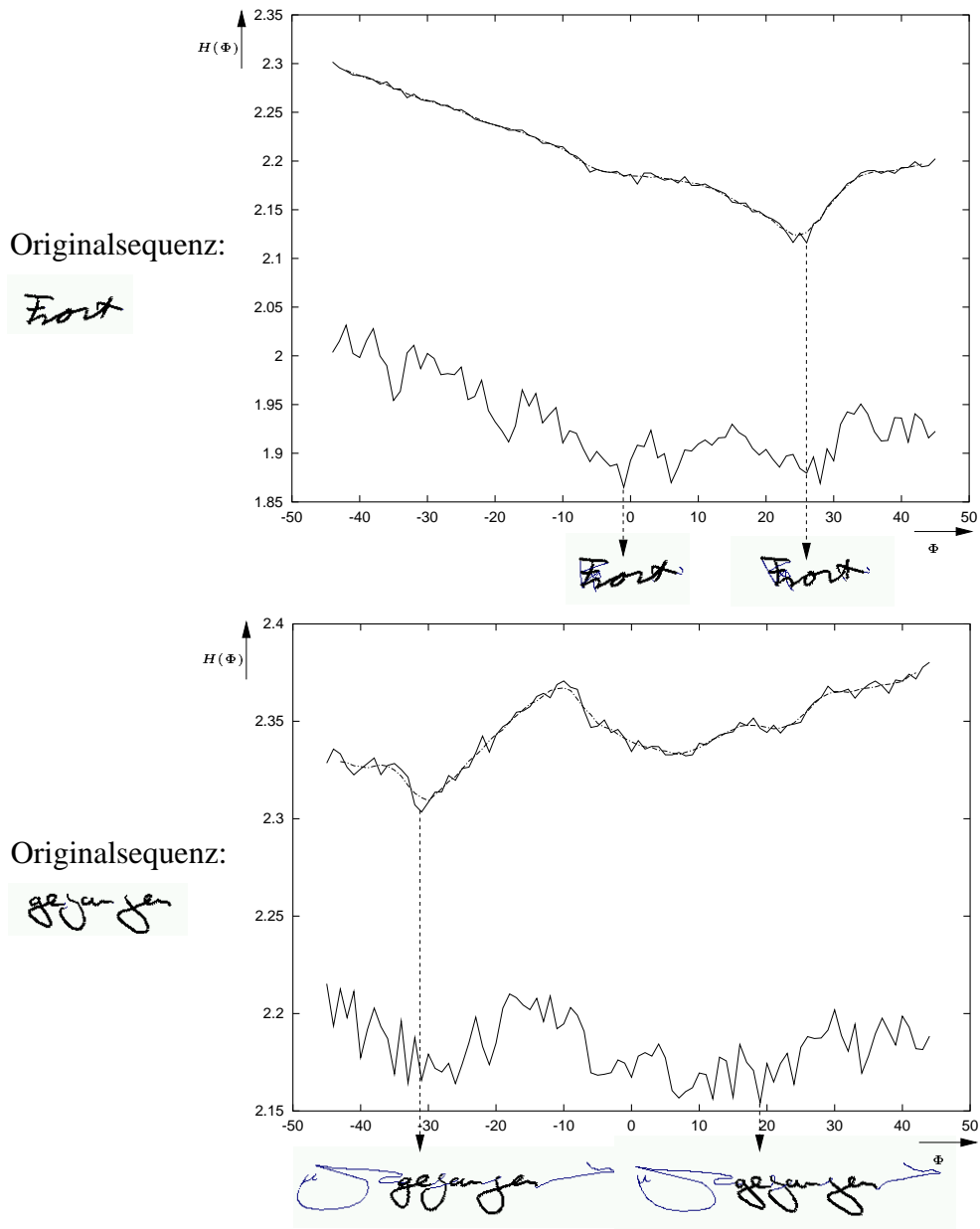


Abbildung 2.7: Beispiele für nicht eindeutig schätzbare Slant-Winkel

Einerseits sollte also die Neuabtastung nach der Normalisierung erfolgen um die Konstanz der Abtastvektoren nicht durch geometrische Transformationen zu stören, andererseits scheinen sich genau dann bessere Ergebnisse einzustellen, wenn die Normalisierung auf neuabgetasteten Daten durchgeführt wird. Als Lösung für dieses Dilemma kann daher vorgeschlagen werden, in einem ersten Schritt die Rohdaten mit sehr kleiner Vektorlänge abzutasten. Damit wird eine homogenisierte Abtastdichte erreicht, ohne signifikante Verluste hinnehmen zu müssen. Anschließend stellt der zweite Schritt die eigentliche Normalisierung dar, während mit dem dritten Schritt eine letzte Neuabtastung zur Datenreduktion und Linearisierung virtueller Segmente durchgeführt wird. So lassen sich gute Ergebnisse bei der Normalisierung mit dem Ziel konstanter Abtastvektorlängen kombinieren.

Zusätzlich zu der vorausgehenden Neuabtastung kann die Entropie $H(\Phi)$ mittels Bildung eines gleitenden Durchschnitts geglättet werden, was die Minimumsuche im Entropieverlauf noch robuster gestaltet (strichpunktierte Kurven in Abb. 2.7). Die Störung durch nahe benachbarte Minima (wie im oberen Beispiel in Abb. 2.7) ist damit minimiert.

2.2.3 Größennormalisierung

Ein weiterer Freiheitsgrad, der mit der Normalisierung eingeschränkt werden sollte, ist die Schriftgröße. Für eine einheitliche Skalierung der Schriftgröße lassen sich die Abmessungen unterschiedlicher Bereiche der Schrift heranziehen. Diese Bereiche werden allgemein durch die vier in Abb. 2.8 eingezeichneten Linien abgegrenzt.

Ein Bereich, den alle Buchstaben, Wörter und Wortfolgen gemeinsam haben, ist der Bereich



Abbildung 2.8: Bereichsunterteilung der Handschrift

zwischen Basislinie und Kernlinie, die sog. Kernhöhe. Eine Approximation dieser beiden Linien und eine anschließende Skalierung der gesamten Vektorfolge auf eine vorgegebene Kernhöhe ist daher anzustreben.

Insbesondere bei der Online-Handschrifterkennung bietet sich die Nutzung der oberen und unteren Wendepunkte zur Approximation der Basis- bzw. Kernlinie an. Nachdem davon ausgegangen werden kann, dass keine Zeilenneigung mehr vorliegt, ließen sich die betreffenden Linien durch horizontale Regressionsgeraden durch die Wendepunkte annähern.

Interessant ist in diesem Zusammenhang ein Blick auf die Häufigkeiten mit denen Wörter mit Ober- oder Unterlängen auftauchen. Dazu wurde ein Text mit 26 Mio. Wörtern ausgewertet. Das Ergebnis ist, dass lediglich 1,8 Mio. Wörter (7 %) weder Ober- noch Unterlängen aufweisen. 7,2 Mio. Wörter (27,6 %) verfügen über Unterlängen. 23,5 Mio. Wörter (90,2 %) enthalten Oberlängen¹. Diese Untersuchung belegt, dass als einzig zuverlässiges Referenzmaß zur Größennormalisierung die Kernhöhe anzusehen ist.

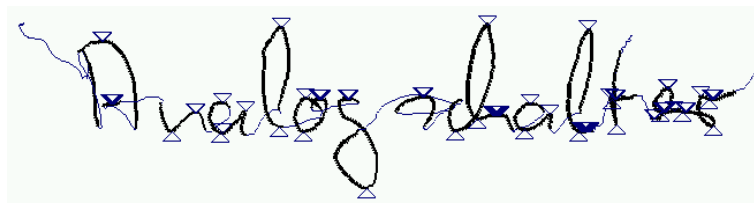
Eine einfache Regression durch alle oberen bzw. unteren Wendepunkte allein kann damit nicht zum Erfolg führen. Als Ergebnis erhielt man bei Einzelworten in 27,6 % (90,2 %) der Fälle Schätzungen für Basislinien (Kernlinien), die zwischen der echten Basislinie (Kernlinie) und der Grundlinie (Oberlinie) verlaufen. Bei Wortfolgen würden sich diese Effekte

¹Die Angaben für Ober- oder Unterlängen sind nicht exklusiv, d. h. die gezählten Wörter mit Oberlängen können durchaus auch Unterlängen enthalten und umgekehrt.

noch verstärken.

Geht man hingegen weiterhin von der Annahme horizontaler Basis- und Kernlinien aus, so läßt sich wiederum ein Projektionsansatz versuchen. Bei einer getrennten y -Projektion von oberen und unteren Wendepunkten, kann auf die so entstandenen (eindimensionalen) Cluster ein modifizierter k-means-Algorithmus angewendet werden (Abb. 2.9).

Für die Initialisierung werden zunächst alle Wendepunkte einer Gruppe (obere oder untere)



initiale obere und untere Wendepunkte (nach Slant-Korrektur)

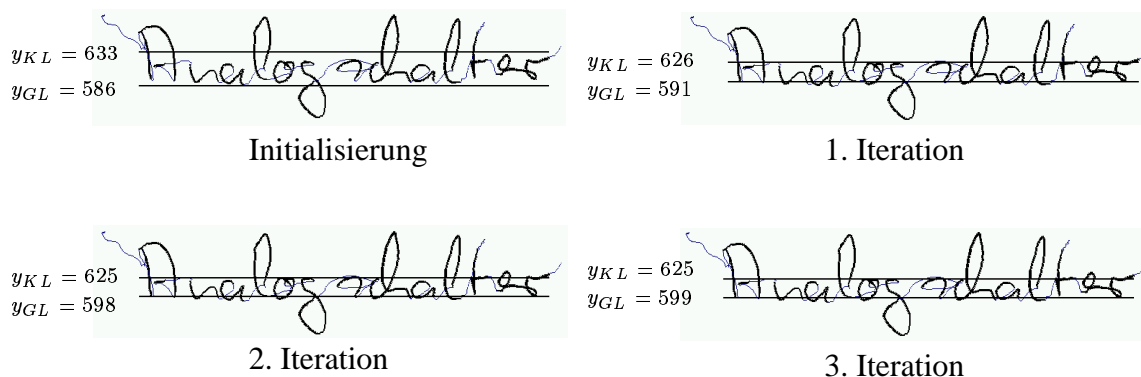


Abbildung 2.9: Iterative Bereichsdetektion

in jeweils einem Cluster zusammengefasst. Damit lassen sich der obere und untere Erwartungswert bilden. Die Positionen dieser Erwartungswerte auf der y -Achse entsprechen somit der geschätzten Kern- bzw. Basislinie. Über ein anzugebendes Abstandsmaß werden aus diesen Clustern alle Wendepunkte entfernt, die dieses Abstandsmaß übersteigen. Mit den reduzierten Clustern werden erneut die Erwartungswerte berechnet und wiederum entfernte Wendepunkte ausgeschlossen. In iterativer Weise werden so die Basis- und Kernlinien schrittweise angenähert. Der Algorithmus wird abgebrochen, wenn sich die Cluster nicht weiter verschieben, d .h. wenn keine weiteren Wendepunkte ausgeschlossen werden. Somit lassen sich kompakte Cluster bilden. Im allgemeinen wird mit diesem Verfahren eine sehr gute Konvergenz erzielt. Mehr als drei Iterationen werden selten benötigt.

2.3 Ergebnisse

Für eine Evaluierung der Verfahren ist es zunächst erforderlich die Testbedingungen zu spezifizieren. Wie bereits erwähnt, kommt der Normalisierung bei schreiberunabhängigen Systemen eine besondere Relevanz zu. Die Testmenge sollte daher Proben mehrerer Schreiber beinhalten. Die Ermittlung der Parameter, wie Neigungswinkel und Kernhöhe ist besonders dann problematisch, wenn eine Probe aus wenig Abtastvektoren besteht. Der Schwerpunkt der Tests liegt somit auf der Normalisierung von Einzelworten. Da diese häufig aus wenigen Buchstaben bestehen und damit eine sinnvolle Normalisierung schwer realisierbar ist, muß darüber hinaus auch ein optimaler Grenzwert für eine Rückweisung gefunden werden.

Ausgehend von diesen Anforderungen, wurde aus dem schreiberunabhängigen Test-Set, bestehend aus insgesamt 4000 Einzelworten per Zufallsauswahl eine Teilmenge von 413 Einzelworten zusammengestellt. Um kostspielige Tests mit dem Gesamtsystem für verschiedene Parameterkombinationen zu vermeiden, wird die Vorverarbeitung zunächst weitgehend entkoppelt betrachtet.

Für die 413 Proben wurden darum die Basis- und die Kernlinien, sowie der Schriftneigungswinkel individuell von Hand ermittelt [Haf99]. Das Set-up für die Vorverarbeitung wird hinsichtlich der mittleren linearen Abweichung zu den manuell bestimmten Referenzwerten optimiert. Im Falle der Schriftneigung wird die Abweichung in Grad angegeben (Tabelle 2.1). Bei der Schätzung der Basis- und Kernlinie erfolgt die Angabe in Pixel bei einer Auflösung von 300 dpi (Tabelle 2.2).

Die z. T. widersprüchliche Forderung nach einer möglichst sicheren Schätzung der Parameter sowie nach einer möglichst umfassenden Normalisierung auch in schwierigen Fällen mit wenigen Abtastpunkten erfordert eine Kompromißlösung. Dementsprechend wird jeweils auch die Anzahl der zurückgewiesenen Wörter angegeben, die zugleich mit dem mittleren Fehler zu optimieren ist.

Wesentlicher Anhaltspunkt für eine Rückweisung ist die Anzahl der verfügbaren Abtastvektoren. Die jeweils akzeptable Mindestanzahl N_{θ}^{-} für die Slantkorrektur ist in Spalte 1 der Tabelle 2.1 angegeben. Wird eine Neuabtastung (NA) zwecks Glättung vorab durchgeführt, so beeinflußt die Abtastvektorlänge natürlich die Gesamtanzahl der Vektoren. Die Spalten 2-6 geben die mittleren Fehler $\frac{1}{413} \sum_{i=1}^{413} |\theta_{0,i} - \theta_i|$ der errechneten Winkel θ_i der i -ten Schriftprobe aus den Testdaten zu den manuell gesetzten Referenzwinkeln $\theta_{0,i}$ wieder. In Klammern findet sich die jeweilige Anzahl zurückgewiesener Wörter. Verglichen mit den nicht normalisierten Testdaten, bei denen sich ein mittlerer Fehlerwinkel von $\frac{1}{413} \sum_{i=1}^{413} |\theta_{0,i}| = 16,47$ Grad ergibt, bedeutet dies im günstigsten Fall (bei $\hat{l} = 1$) eine Slant-Reduktion um 53 % auf 7,72 Grad. Ein positiver Nebeneffekt bei der Abtastung mit kurzen Abtastvektoren ($\hat{l} = 1$) ist, dass die Wahl der Rückweisungsschwelle N_{θ}^{-} weitgehend

N_{θ}^{-}	keine NA	$\hat{l} = 1$	$\hat{l} = 2$	$\hat{l} = 3$	$\hat{l} = 5$
40	8.84 (0)	7.78 (0)	8.14 (0)	9.01 (0)	10.05 (2)
60	8.74 (5)	7.78 (0)	8.14 (0)	9.01 (1)	9.78 (22)
80	8.85 (10)	7.78 (0)	8.14 (0)	8.96 (5)	10.30 (52)
100	8.99 (14)	7.78 (0)	8.13 (1)	8.78 (19)	10.76 (97)
120	9.18 (32)	7.78 (0)	8.09 (5)	9.05 (34)	11.03 (152)
140	9.41 (42)	7.78 (0)	8.04 (15)	9.43 (56)	12.12 (199)
160	9.77 (69)	7.78 (0)	8.02 (24)	9.72 (80)	12.87 (242)
180	10.25 (91)	7.78 (0)	8.27 (32)	9.98 (111)	13.69 (283)
200	10.53 (110)	7.78 (1)	8.42 (45)	10.37 (141)	14.69 (319)
220	11.05 (132)	7.72 (4)	8.78 (59)	10.96 (169)	15.21 (341)

Tabelle 2.1: Ergebnisse der Slantkorrektur

unkritisch ist. Bei einer Glättung des Entropieverlaufes mit einem gleitenden Durchschnitt wird der mittlere Fehler nochmals leicht auf $7,31^{\circ}$, bzw. um insgesamt 56 % verringert.

Analog zu der Berechnung der mittleren Winkelabweichung kann eine mittlere Abweichung der Basis- und Kernlinien zwischen manuellen Referenzwerten und geschätzten Werten bestimmt werden. Mit der Referenz-Basislinie $y_{BL,0,i}$ der i -ten Probe und den geschätzten Basislinien $y_{BL,i}$, ergibt sich die mittlere Abweichung in Pixel zu $\frac{1}{413} \sum_{i=1}^{413} |y_{BL,0,i} - y_{BL,i}|$. Die mittleren Abweichungen sind wiederum in Tab. 2.2 für die Abtastvektorstärke $\hat{l} = 1, 2, 3$ gegeben. Die Parameter 10, 20, 30 geben in dieser Tabelle die vorzugebende maximale Clustergröße in Pixel nach oben (spaltenweise) bzw. nach unten (zeilenweise) an. Bei moderater Wahl der Rückweisungsschwelle $N^{-} \leq 40$ sind bei dieser Wahl der Abtastvektorstärken keine Rückweisungen zu erwarten. Die Anzahl der zurückgewiesenen Wörter bleibt daher außer Betracht. Die in Tab. 2.2 fett markierten Werte stellen für die Basis- bzw. Kernlinie den optimalen Wert für die jeweilige Abtastvektorstärke dar. Anschaulicher werden die Größenverhältnisse, wenn die angegebenen Abweichungen nicht in Pixel, sondern metrisch angegeben werden. Die 1,75 Pixel Abweichung bei der Basislinie entsprechen 0,1483 mm, die 1,79 Pixel gleichen 0,15mm. Die Abweichungen bei der Kernlinie liegen leicht höher. Die jeweiligen Optimalwerte schwanken zwischen 2,98 Pixel = 0,252 mm und 3,22Pixel = 0,27 mm. Diese Differenzen stehen einer durchschnittlichen Kernhöhe von 26.94 Pixel (ca. 25 mm) gegenüber. Die geringfügig größere Abweichung bei der Kernhöhenabschätzung liegt in der deutlich größeren Häufigkeit von Oberlängen begründet, die einen stärkeren Störeinfluss ausüben, als die weniger häufig auftretenden Unterlängen bei den Basislinien.

		Basislinie			Kernlinie		
		10	20	30	10	20	30
$\hat{l} = 1$	10	2.35	1.79	2.17	4.27	3.22	3.61
	20	4.61	3.0	3.07	10.82	4.74	4.15
	30	6.28	3.78	3.7	18.01	7.39	6.01
$\hat{l} = 2$	10	2.25	1.76	2.15	4.35	3.05	3.58
	20	4.3	2.9	2.97	10.82	4.64	4.04
	30	6.03	3.66	3.58	17.88	7.46	5.93
$\hat{l} = 3$	10	2.22	1.75	2.24	4.45	2.98	3.49
	20	4.02	2.76	2.84	10.98	4.62	4.04
	30	5.68	3.49	3.44	18.12	7.41	6.00

Tabelle 2.2: Ergebnisse der Größenskalierung

2.4 Kapitelzusammenfassung

Mit den hier vorgestellten Vorverarbeitungsverfahren soll die Varianz, mit der die Handschrift gewöhnlich erzeugt und erfasst wird, noch vor der eigentlichen Modellierung weitgehend eingeschränkt werden.

Im wesentlichen konnten drei Störgrößen identifiziert werden, die im Rahmen der Vorverarbeitung reduziert werden konnten. Dies sind die variable Schreibgeschwindigkeit und die meist schreiberabhängige Schriftneigung und Schriftgröße. Das erste Problem wurde in dem hier entwickelten Gesamtsystem durch eine (formtreue) Neuabtastung gelöst. Für die Minimierung der Schriftneigung und der Zeilenneigung konnte ein effektiver Ansatz über die Entropieminimierung der Integralprojektionen präsentiert werden. Die Schriftgrößennormierung wurde durch ein iteratives Verfahren zur Grund- und Basislinienapproximation realisiert. Aus den durchgeführten Untersuchungen läßt sich ein allgemeines Ablaufschema für die Vorverarbeitung skizzieren:

1. räumlich äquidistante Abtastung mit $\hat{l} = 1$
2. Normalisierung
 - (a) Korrektur der Zeilenneigung
 - (b) Slantkorrektur
 - (c) Größennormalisierung
3. formtreue Neuabtastung mit $\hat{l} = 5$

Diese Vorverarbeitungsschritte werden im folgenden für sämtliche schreiberunabhängigen Verfahren durchgeführt. Da gewisse Eigenschaften bei schreiberabhängigen Systemen mittrainiert werden können, wird dafür die Vorverarbeitung auf den 3. Schritt reduziert. Eine Sonderanwendung stellt die Korrektur der Zeilenneigung dar, da im allgemeinen Schreiblinien vorgegeben werden. Diese Korrektur kann im Bedarfsfall speziell für den schreiberabhängigen, wie auch für den schreiberunabhängigen Modus aktiviert werden.

Wenngleich das hier gewählte Testszenario mit der Verarbeitung von Einzelworten bereits den ungünstigsten Fall darstellt, konnten bereits erstaunliche Genauigkeiten - insbesondere bei der Grundlinienschätzung erzielt werden. Weiterhin ist zu unterstreichen, dass selbst die manuell bestimmten Referenzwerte für den Neigungswinkel und die Grund- und Basislinien nicht unstrittig sind. Häufig können für diese Variablen selbst durch einen menschlichen Betrachter keine eindeutigen Werte angegeben werden (siehe unteres Beispiel in Abb. 2.7). Eine absolute Genauigkeit kann also bei dieser Problemstellung nicht erwartet werden.