

Vorverarbeitung und Merkmalsextraktion in der Online-Handschrifterkennung

Kai Simon
{ksimon@informatik.uni-freiburg.de}

Zusammenfassung

Für eine erfolgreiche Klassifikation eines Schriftzuges ist die Gewinnung von geeigneten, gegenüber Translation, Rotation und Skalierung invarianter Merkmale notwendig. Damit eine robuste Merkmalsextraktion erfolgen kann, werden die diskret abgetasteten Ortskoordinaten zuvor mithilfe einer Splinekurve geglättet. Bei der Glättung kommen B-Splines zur Anwendung, welche es ermöglichen, Spitzen in den Abtastpunkten durch Knicke zu modellieren. Somit bleiben auch nach einer Glättung wichtige Charakteristiken der Abtastdaten erhalten und können in eine spätere Merkmalsberechnung einfließen. Nach dem Erzeugen der Splinekurve, erfolgt eine Neuabtastung. Hierbei sollen die neu abgetasteten Punkte äquidistant bezüglich der Kurve sein. Um dies zu erreichen, wird die Splinekurve approximativ nach Bogenlänge umparametrisiert. Die Umparametrisierung stellt einen hohen Rechenaufwand dar, welcher bei der Online-Handschrifterkennung nicht akzeptabel ist. Es wird eine schnellere Alternative vorgestellt, welche jedoch eine größere Ungenauigkeit in der Äquidistanz der neu abgetasteten Punkte zur Folge hat. Im Anschluss werden die invarianten Merkmale 'signed ratio of tangents' und 'normalized curvature' für die neu abgetasteten Punkte berechnet. Diese Berechnungen erfolgen direkt auf Grundlage der Splinedarstellung, wodurch, im Gegensatz zu einer Interpolation, eine höhere Rechengenauigkeit gewährleistet ist.

Keywords Online-Handschrifterkennung, Splinekurve, Bogenlängenparametrisierung, Approximation, invariante Merkmale, Kurvenglättung, UNIPEN

Inhaltsverzeichnis

1	Einleitung	4
1.1	Datenformat	7
2	Kurvenglättung durch Splines	8
2.1	Warum Splines	8
2.2	Darstellungsform der Spline	9
2.3	Kurvenparametrisierung	11
2.4	Kubische Splines mittels stückweise zusammengesetzter Polynomfunktionen	14
2.5	B-Splines	19
2.5.1	Definition und Eigenschaft der B-Spline Basis	19
2.6	Kurvenberechnung	20
2.6.1	Vorverarbeitung der Ortskoordinaten	20
2.6.2	B-Spline Berechnung	21
3	Neuabtastung nach Bogenlänge	21
3.1	Approximative Umparametrisierung nach Bogenlänge	22
3.2	Neuabtastung mithilfe der Bogenlängenkurve	23
3.3	Fehlerbetrachtung	25
4	Merkmalsberechnung	26
4.1	Normierte Ortskoordinaten $\mathbf{P}_k^{(\text{norm})}$ beziehungsweise $\tilde{\mathbf{P}}_j^{(\text{norm})}$	26
4.2	Tangentensteigungswinkel θ	27
4.3	Signed ratio of tangents R_α	27
4.4	Normalized curvature χ	28
5	Ergebnisse	29
5.1	Einfluß der Neuabtastung auf die Erkennungsrate	30
5.1.1	Einfluss der Vorverarbeitung auf die Erkennungsrate	31
5.1.2	Vergleich der Neuabtastungsmethoden	31
6	Zusammenfassung und Ausblick	31

7 Danksagung	33
A Berechnung von 'signed ratio of tangents' über Splinedarstellung	34

1 Einleitung

Im Laufe der Entwicklung des Computers wurden verschiedene Eingabegeräte für den Anwender entworfen. Der Informatik Duden [3] teilt grob in folgende Zeitabschnitte ein: “Zu den typischen Eingabegeräten einer Rechenanlage zählten in den 50er- und 60er-Jahren Lochstreifen- und Lochkartenleser, in den 70er und 80er-Jahren die Tastatur und im technischen Bereich Lichtgriffel¹. Zurzeit sind es Tastatur, Maus, Scanner, Videokameras und Spracheingabegeräte.”

Durch die heutige Vielfalt der Eingabegeräte wird dem Anwender ein benutzerfreundlicher Umgang mit dem Computer ermöglicht. Während sich die Maus als Navigationshilfe (Zeigeinstrument) bei der Datenbearbeitung am Bildschirm etabliert hat, dominiert bei der Texteingabe die Tastatur als wichtigstes Erfassungsgerät. Der Wunsch nach “natürlichen” Alternativen zur Texteingabe, hatte zur Folge, dass Sprach- und Schrifteingabegeräte früh entwickelt wurden. Doch erst die Rechenleistung heutiger Rechner haben diesen Alternativen den Einstieg in die Anwenderwelt ermöglicht. Denn während bei der Tastatureingabe eine eindeutige Zuordnung zwischen Tastendruck (digitalem Signal) und ASCII Code erfolgen kann, ist dies bei der Sprach- und Schrifteerkennung nicht möglich. Hier werden die digitalen Eingabesignale erfasst, jedoch besteht keinerlei Bezug zu einem ASCII Code, dieser Bezug muss mittels Erkennungsalgorithmen durch den Computer geschaffen werden. Es sind hierzu komplexe Berechnungen notwendig, damit eine möglichst genaue Zuordnung vorgenommen werden kann. So müssen beispielsweise aus verschiedenen Schreibvariationen eines Buchstabens gleiche Merkmale gefunden werden, welche diesen Buchstaben von anderen unterscheidet. Es ist eine Frage der Erkennungsgenauigkeit, ob der erkannte Text der eigentlichen Eingabe entspricht.

Bei der Handschrifterkennung unterscheidet man zwischen Online- und Offline-Verfahren. Beim Offline-Verfahren wird bereits handgeschriebener Text indirekt durch einen Scanner erfasst und vom Computer ausgewertet. Die Schwierigkeit besteht einerseits in der Extraktion der Handschrift aus der digitalisierten Bildmatrix, andererseits in der anschließenden Wort- beziehungsweise Buchstabenklassifikation. Im Gegensatz zum Online-Verfahren besteht keine Möglichkeit Informationen über Schreibgeschwindigkeit oder Stiftwinkel aus der Bildmatrix zu rekonstruieren. Beim Online-Verfahren wird der Text direkt während des Schreibvorgangs eingelesen. Die Stiftbewegungen werden als eine Sequenz von Orts- und Zeitkoordinaten erfasst. Zusätzlich besteht die Möglichkeit, den Stiftwinkel und Information über Stiftkontakt mit der Schreiboberfläche, in jedem Abtastpunkt digital einzulesen. Das ursprüngliche Verfahren zur Online-Schrifteingabe hat sich aus dem Entwurf technischer Zeichengeräte entwickelt. Die Eingabe erfolgt über ein Grafiktablett, auf welchem mit einem Griffel geschrieben wird. Durch die Entwicklung sogenannter PDA's² hat sich ein wei-

¹Lichtgriffel (engl. light pen): Stift, in dessen Spitze anstelle einer Mine eine lichtempfindliche Zelle (Photozelle) eingebaut ist. Durch Berühren der Bildschirmfläche mit der Griffelspitze können Punkte und Linien gezeichnet werden.

²Abkürzung für "Personal Digital Assistant" auch Organizer genannt. Handlicher Miniatur-Computer, bei dem Eingaben nicht über eine Tastatur, sondern auf dem Display mit Hilfe eines speziellen Stiftes erfolgen. Das PDA wird hauptsächlich für die Verwaltung von Adressen, Terminen und Kurznotizen benutzt.

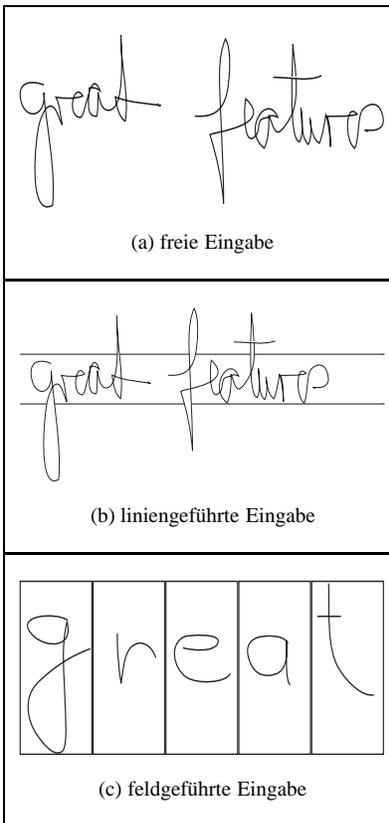


Abbildung 1: Vorgaben bei der Texteingabe

teres neueres Verfahren entwickelt. Hierbei sind die Geräte mit drucksensitiven Bildschirmen (Displays) ausgestattet, auf denen mit einem gewöhnlichen Stift (ohne Mine) geschrieben werden kann. Diese Eingabemöglichkeit ist besonders effizient und platzsparend und gewinnt durch den vermehrten Einsatz der Geräte an zunehmender Bedeutung.

Bei der Texteingabe selbst werden weitere Unterscheidungen vorgenommen. Als Beispiel sind folgende Vorgaben möglich (siehe Abbildung 1) :

- Freie Eingabe: es existieren keine Vorgaben, der Text kann beliebig auf die Schreiboberfläche geschrieben werden.
- Liniengeführte Eingabe: der Text muss anhand einer vorgegebenen Linie ausgerichtet werden.
- Feldgeführte Eingabe: der Text muss in vorgegebene Textfelder Buchstabe für Buchstabe eingegeben werden.

- “Graffiti style³” Eingabe: hierbei muss vom Schreiber eine neues Schriftbild erlernt werden.

Je nach Vorgaben ergeben sich verbesserte Erkennungsvoraussetzungen, jedoch wird der Schreibfluss teilweise stark beeinträchtigt. So kann zum Beispiel im feldgeführten Fall eine Segmentierung der Schrift vollständig entfallen, jedoch kann von einer flüssigen Eingabemöglichkeit nicht die Rede sein. Gerade die Segmentierung, also die Aufteilung eines Wortes in seine einzelnen Buchstaben, stellt ein nicht triviales Problem in der Handschrifterkennung dar.

Bei Erkennungssystemen unterscheidet man generell zwischen schreiberunabhängigen und schreiberabhängigen Systemen. Benötigt man bei schreiberunabhängigen Systemen keine vorherige Schriftprobe, so ist bei der schreiberabhängigen Methode ein vorheriges Training Pflicht. Anhand dieser ersten Schreibprobe wird vom Rechner ein persönliches Schreibprofil erstellt, welches zu einer späteren Klassifikation herangezogen wird.

Beim Schreiben ergeben sich zwangsläufig Variationen von ein und demselben Buchstaben. So kann zum Beispiel das Aussehen eines geschriebenen 'e's am Anfang, gegenüber einem 'e' in der Mitte eines Wortes Unterschiede aufweisen. Der Rechner muss in der Lage sein, beiden Buchstaben bei einer späteren Handschrifterkennung die gleiche Semantik zuzuordnen.

Ziel dieser Arbeit ist es, durch Vorverarbeitung der Abtastdaten, eine robuste Grundlage für neue invariante Merkmale zu bilden, und diese aus den vorverarbeiteten Schrift-daten zu berechnen. Hiermit soll die Diskriminierungsfähigkeit des Klassifikator gesteigert werden. Die Merkmale 'signed ratio of tangents' und 'normalized curvature', welche hier genauer beschrieben werden, besitzen die Eigenschaft gegenüber Rotation, Verschiebung und Skalierung invariant zu sein. Das heißt, dass für zwei Buchstaben, welche sich durch eine beliebige Kombination der aufgeführten Bewegungen ineinander überführen lassen, diese Merkmale konstant bleiben. Es wird ersichtlich, dass eine hohe Invarianz das Merkmal stabil gegenüber Variationen macht. Um jedoch diese Merkmale berechnen zu können, müssen die diskret abgetasteten Ortskoordinaten vorverarbeitet werden. Bei der Gewinnung des Merkmals 'normalized curvature' ist es notwendig, Ableitungen höherer Ordnung zu berechnen. Betrachtet man die abgetastete Handschrift so fällt auf, dass die digitalisierte Schrift, je nach Eingabegerät, ein mehr oder minder starkes Rauschen aufweist. Dieses Rauschen verstärkt sich bei Ableitungen höherer Ordnung, wodurch das Merkmal an Aussagekraft verliert. Die Ursachen für das Rauschen der digitalisierten Daten können unterschiedliche Ursprünge haben:

- ungenügende Rasterauflösung
Als Folge entsteht bei einer mit dem Stift gezeichneten Diagonalen ein typisches Treppmuster beim Verbinden der Abtastwerte (siehe Abbildung 2).

³Bei der Entwicklung des Palm-PDA fand die Idee, dem Benutzer die Schrift des Computers beizubringen, ihre Umsetzung. Es lernten Millionen Käufer bereitwillig "Graffiti", das Palm-Alphabet, in dem beispielsweise das K eher einem griechischen ALPHA ähnelt. Für diese Lernerarbeit wird der Benutzer belohnt — die Software erkennt die Kringel praktisch fehlerfrei. Dabei bezieht die Erkennungssoftware nicht nur die Form des fertigen Buchstaben in ihre Analyse ein, sondern auch seine Entstehung wird berücksichtigt: So "malt" man ein 'v' wie ein 'u', einziger Unterschied man beginnt rechts statt links.

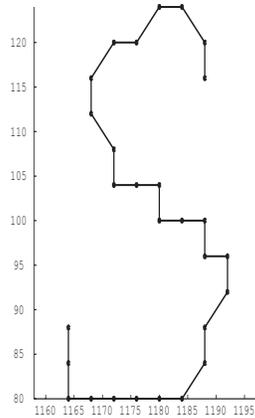


Abbildung 2: Treppmuster bedingt durch zu geringe Rasterauflösung der Eingabegeräte

- geringe Abtastfrequenzen
- mangelnde Fähigkeit des Benutzers mit dem Eingabegerät umzugehen

Um das Rauschen zu mindern ist es notwendig den abgetasteten Schriftzug zu glätten. Die Abtastpunkte werden in diesem Fall nicht interpolierend, sondern approximativ angenähert. Durch diese Art der Glättung befinden sich die originalen Abtastpunkte nicht mehr auf dem Schriftzug, womit eine Neuabtastung des geglätteten Schriftzuges notwendig wird. Nach der Glättung und Neuabtastung ist es möglich die Merkmale auf Grundlage der Splinedarstellung zu berechnen.

In dieser Arbeit wird folgender Ansatz verfolgt. In Abschnitt 2 wird die Glättung des Schriftzuges mittels Splines beschrieben. Um bei der Neuabtastung äquidistante Abtastpunkte zu erhalten, wird die Spline approximativ nach Bogenlänge umparametrisiert; siehe hierzu Abschnitt 3. In Abschnitt 4 werden verschiedene Merkmale und deren Berechnung anhand der neu abgetasteten Punkte aufgezeigt. Im Abschnitt 5 werden die Resultate verschiedener Experimente vorgestellt und in Abschnitt 6 eine Zusammenfassung und Ausblick auf weiterführende Ansätze gegeben.

1.1 Datenformat

Um auf genügend große Trainings- und Testdatensätze von Online-Handschriften zugreifen zu können, wurden die UNIPEN [5, 9] Benchmark Daten “Train-R01/V07” verwendet. Der UNIPEN Datensatz wurde speziell für die Erforschung und Entwicklung von Online-Handschrift Anwendungen entworfen. Die gesammelten Online-

Handschriften in der UNIPEN Datenbank wurden von verschiedenen Personen mit unterschiedlicher Hardware erfasst und in einem vorgegebenen Format abgespeichert. Es bietet außerdem die Möglichkeit verschiedene Online-Handschrifterkennungssysteme untereinander in Vergleich zu setzen, welche auf dem UNIPEN Datensatz getestet wurden. Im Format selbst wird zwischen drei Hierarchien unterschieden:

1. SCHRIFTZEICHEN: jedes einzelne Zeichen des ASCII Alphabets kann hier auftreten.
Beispiel: "A","b","O","2"," ","@"
2. WÖRTER: Sequenz aus einem oder mehreren Schriftzeichen (ohne Leerzeichen) aus dem ASCII Alphabet.
Beispiel: "A","b","O","The","AC/DC","{a|b}"
3. TEXT: kein oder mehrere Schriftzeichen aus dem ASCII Alphabet (ohne führendes/gehendes Leerzeichen), kein Auftreten mehrerer aufeinander folgender Leerzeichen
Beispiel: "","A","that nothing more happened,"

Diese Arbeit befasst sich mit der Schriftzeichenerkennung, jedoch sind Erweiterungen auf Wörter und Text durchaus sinnvoll und denkbar.

Beim Aufzeichnen der Stiftbewegungen werden zwischen zwei verschiedenen Zuständen der Stiftposition im Bezug zur Schreiboberfläche unterschieden:

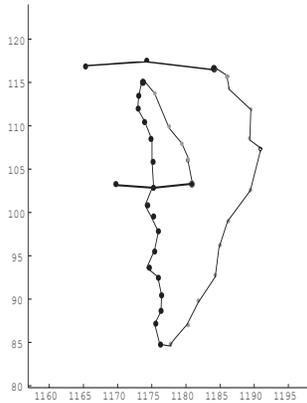
- '.PEN_DOWN': hierbei berührt der Stift die Schreiboberfläche es werden die diskreten Ortskoordinaten als '.PEN_DOWN' Einheiten abgespeichert.
- '.PEN_UP' der Stift verlässt die Schreiboberfläche es werden die diskreten Ortskoordinaten als '.PEN_UP' Einheiten abgespeichert. Nicht bei allen Eingabegeräten ist es auch im '.PEN_UP' Zustand möglich diskrete Werte abzutasten, in diesen Fällen wird eine leere Einheit erzeugt.

Eine solche Einheit von '.PEN_UP' und '.PEN_DOWN' Daten, wird in der UNIPEN Terminologie als *Komponente* bezeichnet. Ein aufgezeichnetes Datensegment für ein Schriftzeichen kann somit zum Beispiel eine Gestalt, wie in Abbildung 3 dargestellt, besitzen. Es ist zu bemerken, dass in der gesamten Arbeit nur aufgezeichnete Ortskoordinaten in den [.PEN_DOWN] Komponenten betrachtet werden. Diese Einschränkung erfolgt aus der Erkenntnis, dass viele [.PEN_UP] Komponenten für die Merkmalsextraktion unbrauchbare Abtastinformationen beinhalten.

2 Kurvenglättung durch Splines

2.1 Warum Splines

Bei der Approximation will man eine möglichst glatte Kurve durch die Abtastpunkte definiert bekommen. Mit anderen Worten soll die Krümmung nicht zu stark lokal



COORD_COMP	[3x2]	[12x2]	[17x2]	[4x2]	[2x2]
PEN_COMP	[.PEN_DOWN]	[.PEN_UP]	[.PEN_DOWN]	[.PEN_UP]	[.PEN_DOWN]

Abbildung 3: Aufgezeichnetes Datensegment mit [.PEN_DOWN] (dicke Abtastpunkte) und [.PEN_UP] (dünne Abtastpunkte) Komponenten. Zur leichteren Erkennung des Buchstabens wurden die Abtastpunkte mittels Geraden verbunden.

abweichen. Sollen nun aber viele Abtastpunkte durch eine Polynomfunktion approximiert werden, so sind dazu Polynome von hohem Grad erforderlich, was wiederum zum Oszillieren der Kurve führt. Hierdurch können extreme Abweichungen des Kurvenverlaufs gegenüber der erwarteten Kurvenform auftreten (siehe Abbildung 4).

Die Unterteilung der Kurve in einzelne Kurvensegmente, welche einen festen Polynomgrad besitzen, macht es möglich, dieses Problem zu lösen. Zu gegebenen $n + 1$ Abtastpunkten $\mathbf{P}_0, \dots, \mathbf{P}_n \in \mathbb{R}^2$ werden somit n Kurvensegmente definiert, die jeweils ein aufeinander folgendes Paar von Punkten $\mathbf{P}_i, \mathbf{P}_{i+1}$ für $0 \leq i \leq n - 1$ approximieren. Jedes der Kurvensegmente wird durch ein separates Polynom definiert, wodurch man eine stückweise polynomiale Kurve, eine so genannte Spline erhält.

Die nachfolgenden Erläuterungen zu Splines sollen einen groben Überblick vermitteln, für umfassendere Hintergrundinformationen wird auf [4] verwiesen.

2.2 Darstellungsform der Spline

Die Darstellung des Schriftzuges durch Splines erfordert eine funktionale Beschreibungsform. Für Kurven in der Ebene gibt es folgende drei Möglichkeiten:

1. explizite Darstellung

- $y = f(x)$

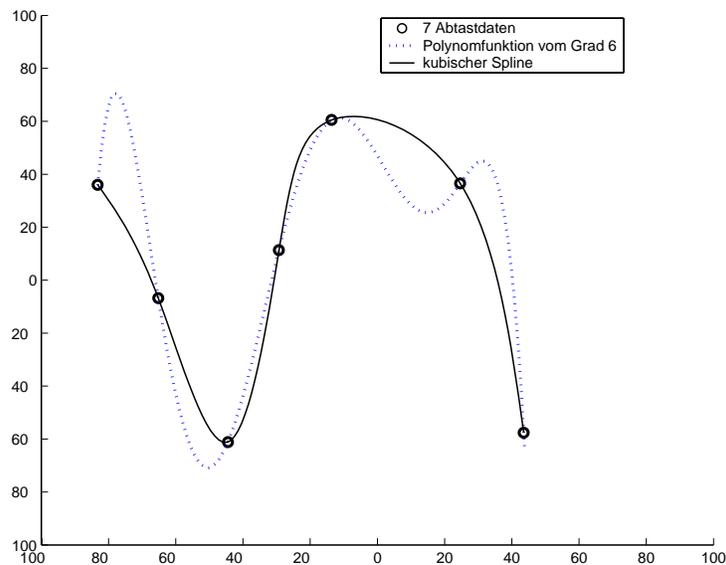


Abbildung 4: Vergleich Interpolation Polynomfunktion vs. Splinefunktion

- Beispiel: $y(x) = \begin{cases} \frac{b}{a}\sqrt{a^2 - x^2} & y \geq 0 \\ -\frac{b}{a}\sqrt{a^2 - x^2} & y < 0 \end{cases}$

Probleme:

- Für ein x-Wert darf es nur einen y-Wert geben (Kreis kann zum Beispiel nicht geschlossen dargestellt werden)
- Keine Kurven mit vertikalen Tangenten möglich

2. implizite Darstellung

- $f(x, y) = 0$
- Beispiel: $x^3 + x^2 - y^2 + 2xy = 0$

Probleme:

- Gleichung kann mehrere Lösungen besitzen, was durch Randbedingungen eingeschränkt werden muss
- Richtung der Tangente ist schwer zu ermitteln

3. parametrische Darstellung

- $\mathbf{S}(t) = \begin{pmatrix} x(t) \\ y(t) \end{pmatrix}$
- Beispiel: $\mathbf{S}(t) = \begin{pmatrix} 3t \cdot \cos(t) \\ \sin(t) + 4t \end{pmatrix}$ für $t \in]-1, 1[$

Vorteile:

- (a) keine Mehrdeutigkeiten
- (b) Ableitungen lassen sich leicht als Tangentenvektoren (Geschwindigkeitsvektor) beschreiben
- (c) die stückweise Unterteilung, wie sie hier benötigt wird, kann durch Einschränkung des Parameterbereichs erreicht werden

Nachteil:

- (a) Kurvenparameter t muss gewählt werden

Durch ihre positiven Eigenschaften hat sich die parametrische Darstellung in der Computergrafik durchgesetzt. Im Folgenden wird diese Darstellung angewandt, womit für jede Koordinatenrichtung eine Splinekurve berechnet werden muss.

Die parametrische Darstellung des Splines hat somit folgende Gestalt. Sei \mathbb{D} der Wertebereich des Kurvenparameters t

$$t : \mathbb{R} \supseteq \mathbb{D} \rightarrow \mathbb{R}^2$$

$$t \rightarrow \mathbf{S}(t) = \begin{pmatrix} S_x(t) \\ S_y(t) \end{pmatrix}$$

2.3 Kurvenparametrisierung

Wie im vorherigen Abschnitt angedeutet, stellt die Wahl des Kurvenparameters t bei der Erzeugung der Splines einen wichtigen Aspekt dar. Der Kurvenparameter kann beliebig gewählt werden, er muss lediglich monoton wachsend sein. Will man beim Abtasten einer Kurve äquidistante Punkte erzeugen, so ist es notwendig die Bogenlänge in jedem Kurvenpunkt zu kennen. Die Parametrisierung nach Bogenlänge erfüllt eben diesen Anspruch. Bei dieser Art der Parametrisierung gibt der Kurvenparameter t für jeden Punkt $\mathbf{S}(t)$ die Bogenlänge $s_{[0,t]}$ des Kurvensegments aus dem Intervall $[0, t]$ wieder.

$$s_{[0,t]} = \int_0^t \|\mathbf{S}'(u)\| du = t \quad \forall t \in \mathbb{D}$$

Dies induziert, dass die Norm der ersten Ableitung in jedem Kurvenpunkt konstant 1 ist.

$$\|\mathbf{S}'(t)\| \equiv 1 \quad \forall t \in \mathbb{D}$$

Das heißt, die Kurve wird mit konstanter Geschwindigkeit durchlaufen. Um nun die Kurve äquidistant abzutasten, kann einfach der Kurvenparameter t äquidistant gewählt werden. Durch die Eigenschaft, dass der Geschwindigkeitsvektor die Länge 1 besitzt ergeben sich weitere Vereinfachungen, deren Vorzüge spätestens bei der Berechnung der Merkmale zur Geltung kommen.

- bei der Berechnung von Winkeln zwischen Tangenten muss keine Normierung durchgeführt werden
- die Krümmung einer 2-dimensionalen Kurve im Kurvenparameter t berechnet sich direkt aus

$$\kappa = \|\mathbf{S}''(t)\|$$

im Gegensatz zur Berechnung bei beliebigen Parametrisierung der Kurve

$$\kappa = \frac{S'_x(t) \cdot S''_y(t) - S'_y(t) \cdot S''_x(t)}{(S'_x(t)^2 + S'_y(t)^2)^{\frac{3}{2}}}$$

Aus den oben genannten Gründen stellt die bogenlängen Parametrisierung die optimale Parametrisierungsmethode dar. Jedoch weist sie praktische Probleme auf. So ist es beim Erzeugen der Spline nicht möglich die Kurve direkt nach Bogenlänge zu parametrisieren, da die Bogenlängen nicht bekannt sind. Es werden somit im folgenden zwei Alternativen vorgestellt, welche sich durch die gegebenen Abtastpunkte ermitteln lassen.

Seien die Abtastpunkte gegeben durch $\mathbf{P}_0, \dots, \mathbf{P}_n \in \mathbb{R}^2$ und für $1 \leq k \leq n$ beginne das k -te Teilpolynom \mathbf{S}_k bei t_{k-1} und ende bei t_k .

1. Der Kurvenparameter t_k wird gleichgesetzt mit dem diskreten Index der Abtastpunkte, im weiteren *Indexstellen*-Parametrisierung genannt (siehe Abbildung 5)

$$t_0 := 0$$

$$t_k := k$$

Vorteile

- einfache Berechnung
- der lokale Kurvenparameter \hat{t}_k des k -ten Teilpolynoms ist aus dem Intervall $\hat{t}_k \in [0, 1]$, wodurch eine einfache Handhabung für die Teilintervalle möglich wird.

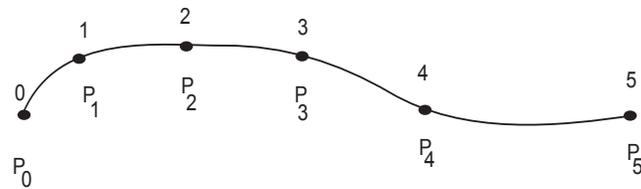
Nachteile

- ungleichmäßige Durchlaufgeschwindigkeit der Splinekurve bei abrupter Wertänderung der Abtastpunkte, evtl. Verlust an numerischer Genauigkeit
2. Als Kurvenparameter t_k wird der geometrische Abstand der Abtastpunkte herangezogen, auch *chordale* Parametrisierung genannt (siehe Abbildung 5)

$$t_0 := 0$$

$$t_k := \sum_{j=1}^k |\mathbf{P}_j - \mathbf{P}_{j-1}|$$

Indexstellen Parametrisierung



Chordale Parametrisierung

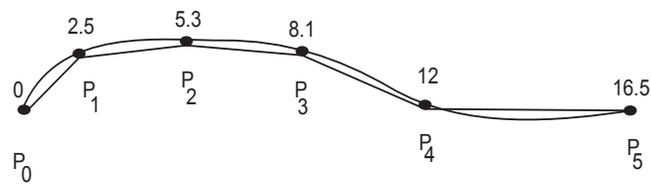


Abbildung 5: Unterschiedliche Wahl des Kurvenparameters t bei der Erzeugung der Spline. Bei der *Indexstellen*-Parametrisierung wird der Parameter durch den Index der Abtastpunkte definiert. Bei der *chordalen* Parametrisierung werden jeweils die Summe der Abstände zwischen den Abtastpunkten als Parameterwert übergeben. Die Spline beginnt jeweils mit dem Kurvenparameter $t = 0$.

Vorteile

- bessere numerische Genauigkeit
- nahezu gleiche Durchlaufgeschwindigkeit der Kurve (Länge des Tangentenvektors liegt nahe bei 1)

Nachteil

- größerer Rechenaufwand wie Methode 1, welcher aber durch Verwendung von Datentypen geringerer Genauigkeit ausgeglichen werden kann

Bei der chordalen Parametrisierung wird der Parameter t über die Approximation der Kurve durch einen Polygonzug beziehungsweise dessen jeweiligen Polygonlängen definiert. Diese Methode bietet eine relativ gute Annäherung an die Bogenlängenvorgabe. Die Grösse der Abweichung zur Bogenlänge definiert sich in diesem Fall über den Verlauf der Kurve zwischen den Abtastpunkten und der Dichte der Abtastpunkte. Liegen die Abtastpunkte nah beieinander, so ist diese Methode eine gute Approximation der Bogenlänge. Sind sie hingegen weit entfernt, so kann die Polygonzuglänge gegenüber der Bogenlänge große Differenzen aufweisen. In Abbildung 6 ist dieser Sachverhalt noch einmal anhand eines allgemeineren Falls grafisch dargestellt. Es wurde die normierte Bogenlänge

$$\hat{s}(\hat{t}) := \frac{s(\hat{t})}{s(1)} \quad \text{für } \hat{t} \in [0, 1]$$

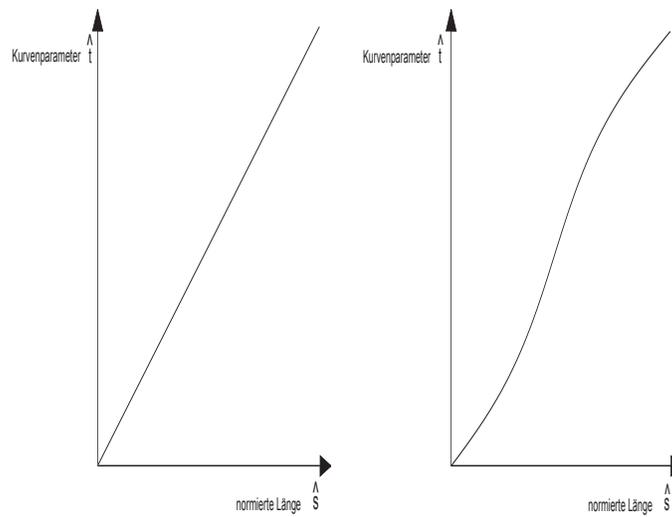
gegenüber dem normierten Kurvenparameter \hat{t} aufgetragen. Als Graph entsteht eine Längenkurve, welche das Verhältnis zwischen Bogenlänge und Kurvenparameter beschreibt. Im Fall der Parametrisierung nach Bogenlänge entspricht die Längenkurve der 1. Hauptdiagonalen. Im Fall einer beliebigen Parametrisierung muss kein Bezug zwischen Bogenlänge und Kurvenparameter existieren. Umso besser die gewählte Parametrisierung die Bogenlänge beschreibt, umso weniger weicht die Längenkurve von der 1. Hauptdiagonalen ab.

Es ergeben sich bei der Neuabtastung mit chordaler Parametrisierung, gerade bei starken Krümmungen im Kurvenverlauf, grössere Abweichungen in der Äquidistanz, da hier der Geschwindigkeitsvektor stark variiert (siehe Abbildung 9).

Bei beiden beschriebenen Alternativen ist es möglich, die Spline nachträglich approximativ nach Bogenlänge abzutasten (siehe Abschnitt 3). Diese nachträgliche Umparometrisierung ist jedoch sehr rechenintensiv. Aus diesem Grund wurden Vergleiche angestellt, ob Abweichungen in der Äquidistanz der neu abgetasteten Punkte Auswirkungen auf die Erkennungsrate haben (siehe hierzu Abschnitt 5).

2.4 Kubische Splines mittels stückweise zusammengesetzter Polynomfunktionen

Ein Spline heißt von der Ordnung m , wenn es aus Polynomen vom Grad $m-1$ zusammengesetzt ist. Es ergeben sich folgende Charakteristiken für die ersten vier Spline



(a) Bogenlängen Parametrisierung. Bogenlänge und Kurvenparameter verhalten sich proportional zueinander.

(b) beliebige Parametrisierung. Die Längenkurve zeigt Abweichungen gegenüber der 1. Hauptdiagonalen auf.

Abbildung 6: Vergleich der Längenkurven im Fall einer Parametrisierung nach Bogenlänge gegenüber einer beliebigen Parametrisierung der Splinekurve.

Ordnungen:

- $m = 1$: Die Spline beschreibt keine echte Kurve, sondern springt von Punkt zu Punkt.
- $m = 2$: Es wird ein Polygonzug durch die Abtastpunkte erzeugt.
- $m = 3$: Die Spline ist stückweise aus Parabeln (Polynome 2. Grades) zusammengesetzt, die an den Übergängen einmal stetig differenzierbar sind.
- $m = 4$: Die Spline ist stückweise aus Polynome 3. Grades zusammengesetzt, die an den Übergängen zweimal stetig differenzierbar sind.

In der Praxis spielen vor allem kubische Splines $m = 4$ eine wichtige Rolle. Laut obiger Beschreibung ist ein kubischer Spline $\mathbf{S}(t)$ stückweise aus Polynomen 3. Grades zur Basis $[t^3, t^2, t, 1]$ zusammengesetzt, welche an den Nahtstellen zweimal stetig differenzierbar sind. Die dritte Ableitung muss an den Nahtstellen nicht mehr stetig sein. Die Nahtstellen (englisch breakpoints) müssen eine monoton steigende Folge beschreiben. Im Falle der Parameterdarstellung geben sie die Intervalle der lokalen Kurvenparameter $[\hat{t}_1, \dots, \hat{t}_n]$ vor.

Kubische Splines erfüllen die Forderung nach möglichst glatten Übergängen zwischen den Polynomstücken. Durch die stückweise zusammengesetzten Polynomfunktionen kann der lokale Kurvenverlauf sehr gut beschrieben werden. Mit Polynomfunktionen höheren Grades wären zwar glattere Übergänge zwischen den Polynomstücken möglich, jedoch nimmt die Neigung zur Oszillation und der Rechenaufwand zur Bestimmung der Koeffizienten zu.

Um die Koeffizienten der Teilpolynome \mathbf{S}_k zu bestimmen, benötigt man an den Nahtstellen folgende Zusatzbedingungen:

Ohne Einschränkung der Allgemeinheit, sei der lokale Kurvenparameter $\hat{t}_k \in [0, 1]$ für $1 \leq k \leq n$ mit $n + 1 = \#$ Abtastpunkte und t nach der Indexstellen-Parametrisierung gewählt. So gilt:

$$\mathbf{S}(t) = \mathbf{S}_k(\hat{t}_k)$$

Wobei der Kurvenparameter t eindeutig das k -te Teilpolynom definierte

$$k := \lfloor t \rfloor + 1 \quad \text{für } t \in [0, n)$$

$$\mathbf{S}_k(\hat{t}_k) = \begin{pmatrix} S_{x,k}(\hat{t}_k) \\ S_{y,k}(\hat{t}_k) \end{pmatrix} = \begin{pmatrix} a_{1,k} \cdot \hat{t}_k^3 + b_{1,k} \cdot \hat{t}_k^2 + c_{1,k} \cdot \hat{t}_k + d_{1,k} \\ a_{2,k} \cdot \hat{t}_k^3 + b_{2,k} \cdot \hat{t}_k^2 + c_{2,k} \cdot \hat{t}_k + d_{2,k} \end{pmatrix}$$

Für $k \geq 2$ gelten bei der Erzeugung folgende Nahtstellenbedingungen:

1. C^1 -Stetigkeit:

$$\mathbf{S}_k(0) = \mathbf{S}_{k-1}(1)$$

$$\mathbf{S}'_k(0) = \mathbf{S}'_{k-1}(1)$$

2. C^2 -Stetigkeit:

$$\mathbf{S}''_k(0) = \mathbf{S}''_{k-1}(1)$$

Die kubischen Splines sind somit C^2 stetig, stimmen also in den Nahtstellen bis zur 2. Ableitung in Richtung und Länge überein, was einen glatten Übergang gewährleistet. Um das Glättungsverhalten der kubischen Splines zu steuern, ist bei ihrer Erzeugung folgende Formel zu minimieren:

$$\min \left(\begin{array}{l} p \sum_{i=0}^n (P_{x,i} - S_x(t_i))^2 + (1-p) \int_{t_0}^{t_n} (S''_x(u))^2 du \\ p \sum_{i=0}^n (P_{y,i} - S_y(t_i))^2 + (1-p) \int_{t_0}^{t_n} (S''_y(u))^2 du \end{array} \right) \quad \text{für } i \in [0, \dots, n], p \in [0, 1]$$

Der Glättungsparameter $p \in [0, 1]$ gibt die Stärke der Glättung an.

Für $p = 0$ beschreibt die Spline eine Gerade, welche den kleinsten quadratischen Abstand bezüglich den Abtastpunkten besitzt.

Für $p = 1$ interpoliert die Spline die Abtastpunkte.

Für $0 < p < 1$ stellt die erzeugte Spline eine Gewichtung der beiden Eigenschaften dar.

Der erste Summand der Formel gibt den quadratischen Abstand der Abtastpunkte zur Spline wieder. Der zweite Summand berechnet auf dem gesamten Intervall $t \in [t_0, \dots, t_n]$, die zweite Ableitung zum Quadrat. Diese Ableitung ist wesentlicher Bestandteil der Krümmung einer Funktion. Die Minimierung über alle Teilpolynome bedeutet also eine Erzeugung einer Spline mit kleinster mittlerer Krümmung.

In Abbildung 7 sind mehrere Parametereinstellungen, bei chordaler Parametrisierung der Splinekurve, am Beispiel des Buchstaben 's' dargestellt. Das Hauptproblem bei der Glättung liegt in der Wahl einer geeigneten Glättungskonstanten p . Besitzt man einen inhomogenen Datensatz von abgetasteten Schriftzügen, wie es im UNIPEN Datensatz der Fall ist, so ist eine feste Wahl der Glättungskonstanten mit Vorsicht zu genießen. Es ist festzuhalten, dass ein fest gewählte Glättungsparameter, wie in [6] beschrieben, bei einigen Segmenten zu subjektiv guten Ergebnissen führt, bei anderen Segmenten jedoch nicht. Dies liegt daran, dass die aufgezeichneten Segmente unterschiedliche Abtastungsdichten aufweisen. Ein Kompromiss verspricht die Methode wie sie in Abschnitt 2.6.2 beschrieben wird.

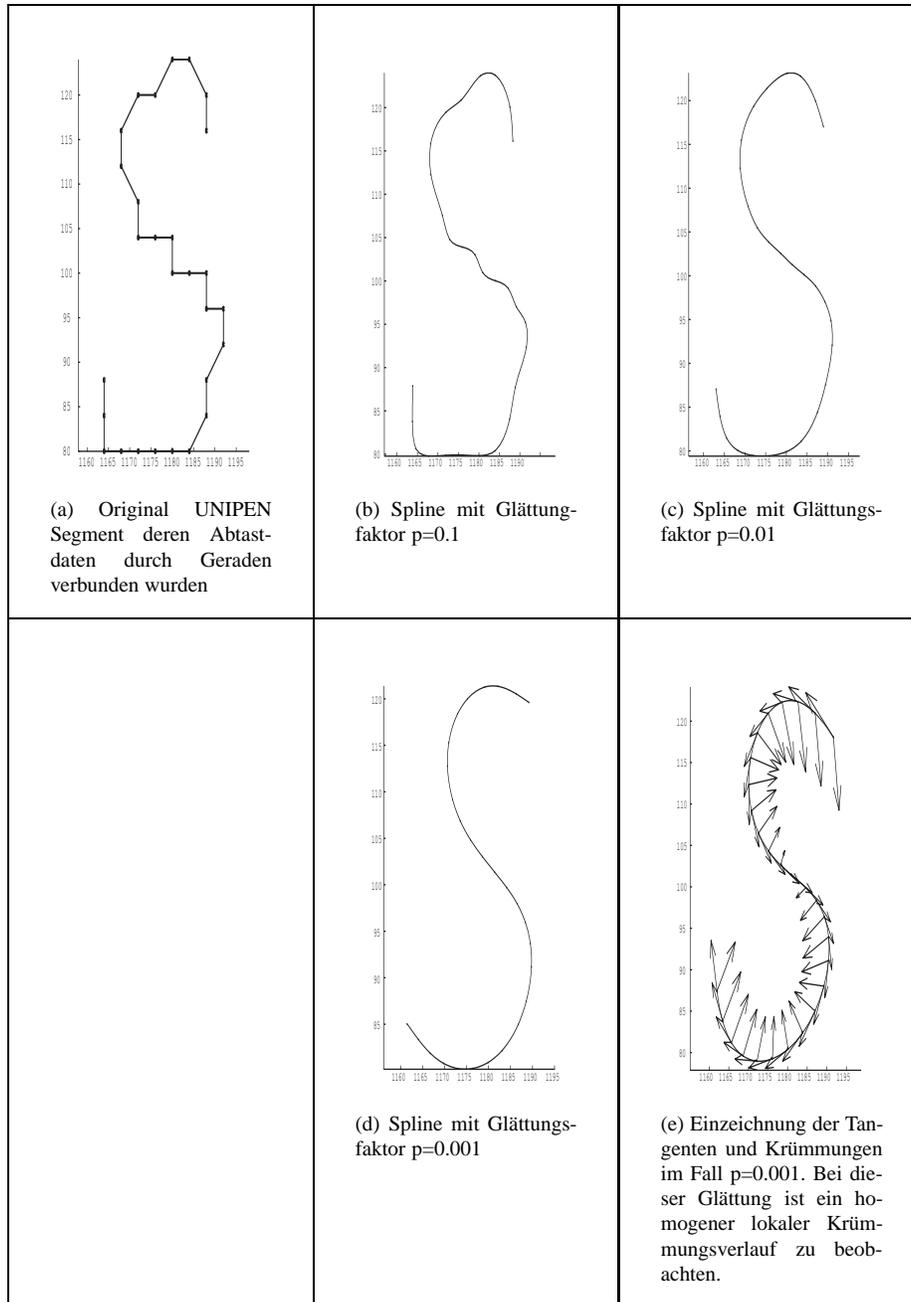


Abbildung 7: Unterschiedliche Glättungen eines Schriftzuges mithilfe einer kubischen Splinekurve. Die Kurve wurde nach der chordalen Methode parametrisiert.

2.5 B-Splines

Bei B-Splines handelt es sich um Splines zur Basis $[N_n, \dots, N_0]$. Diese Basisfunktionen sind lokal definiert, wodurch die Form der B-Spline lokal geändert werden kann. Wie bei Splines ist eine B-Spline aus einzelnen Segmenten zusammengesetzt, die jedoch nun durch eine Sequenz von Knoten t definiert sind. Knoten können im Gegensatz zu Nahtstellen mehrfach hintereinander auftreten, wodurch der Kurvenverlauf der B-Spline gesteuert werden kann.

Die Definition der B-Splines erfolgt durch eine Gewichtung der Kontrollpunkte P_i mit $0 \leq i \leq n$ und den Basisfunktionen $N_{i,k}$, wobei $k =$ Ordnung der Spline und #Kontrollpunkte $= n + 1$ bezeichne

$$B_{n,k}(t) = \sum_{i=0}^n P_i N_{i,k}(t)$$

Die parametrische Darstellung der B-Spline in der Ebene hat somit folgende Gestalt:

$$\mathbf{S}(t) = \begin{pmatrix} B_{x,n,k}(t) \\ B_{y,n,k}(t) \end{pmatrix}$$

2.5.1 Definition und Eigenschaft der B-Spline Basis

Es existieren verschiedene Definitionen der B-Spline Basen, für die Implementierung ist eine rekursive Definition vorteilhaft (siehe [10]).

Sei eine aufsteigende Folge von geordneten Parameterwerten

$$t_0 \leq t_1 \leq \dots \leq t_{n-k}$$

gegeben. So heißt

$$T = \{t_0, \dots, t_{n-k}\}$$

Trägervektor (Knotenvektor). Es gilt für $i = 0, \dots, n + k - 1$

$$N_{i,1}(t) := \begin{cases} 1 & \text{falls } t_i \leq t < t_{i+1} \\ 0 & \text{sonst} \end{cases}$$

$$N_{i,j}(t) := \frac{t - t_i}{t_{i+j-1} - t_i} N_{i,j-1}(t) + \frac{t_{i+j} - t}{t_{i+j} - t_{i+1}} N_{i+1,j-1}(t) \quad \forall j = 2, \dots, k$$

Diese Rekursionsvorschrift bleibt auch für zusammenfallende Knoten unter der Konvention gültig, Terme durch Null zu ersetzen, deren Nenner verschwindet. Die Spline verliert dort eine entsprechende Anzahl an stetigen Ableitungen. Die Anzahl gleich gewählter Knoten wird als deren Multiplizität bezeichnet. Für eine B-Spline der Ordnung k mit einem β -fachen Knoten sind alle Ableitungen bis zur Ordnung $(k - \beta - 1)$ stetig differenzierbar. Ist die Multiplizität $\beta \geq k$, führt dies für die Polynomsegmente links und rechts von diesem Knoten zur Unstetigkeit, womit diese Wahl von β keinen Sinn macht.

Außerdem sind auf einem Knotenintervall höchstens j Basis-Funktionen $N_{i,j}(t)$ der Ordnung j von Null verschieden, somit gilt für das l -te Knotenintervall

$$t \in [t_l, t_{l+1}) \quad \Rightarrow \quad N_{l-j+1,j}, \dots, N_{l,j} \neq 0$$

Will man also einen Kurvenpunkt t^* der B-Spline berechnen, muss man zuerst den Index l des Teilintervalls feststellen und dann alle von Null verschiedenen Gewichtsfunktionen mit der angegebenen Rekursionformel berechnen. Sei $t_l \leq t^* < t_{l+1}$ somit gilt:

$$\mathbf{S}(t^*) = \begin{pmatrix} N_{x_{l-k+1,k}}(t^*)P_{x_{l-k+1}} + \dots + N_{x_{l,k}}(t^*)P_{x_l} \\ N_{y_{l-k+1,k}}(t^*)P_{y_{l-k+1}} + \dots + N_{y_{l,k}}(t^*)P_{y_l} \end{pmatrix}$$

Der Aufwand zur Auswertung der Kurve an der Stelle t^* hängt somit nicht von der Anzahl der Kontrollpunkte, sondern von der Ordnung k der B-Spline ab.

2.6 Kurvenberechnung

2.6.1 Vorverarbeitung der Ortskoordinaten

Je nach Stärke der Glättung können wichtige Unterscheidungsmerkmale eines Buchstabens verloren gehen. Betrachtet man Unterschiede zwischen den Buchstaben 'v' und 'u', so ist die Spitze im 'v' ein wichtiges Merkmal. Es ist somit notwendig, Spitzen gegenüber einer Glättung invariant zu machen. Hierbei kann man sich die angesprochene Multiplizität von Knoten bei der Erzeugung von B-Splines zunutze machen. Denn durch die Multiplizität der Knoten kann die Differentiationsordnung am Übergang zwischen den Segmenten beeinflusst werden, wodurch es möglich wird, Knicke zu modellieren. Als erster Vorverarbeitungsschritt, müssen somit Spitzen in den originalen Abtastwerten gesucht und diese durch Vervielfältigung des Eckpunktes hervorgehoben werden.

Das implementierte Verfahren betrachtet 3 aufeinanderfolgende Punkte $\mathbf{P}_1, \mathbf{P}_2, \mathbf{P}_3$ und berechnet den Winkel zwischen den Vektoren $\overrightarrow{\mathbf{P}_2\mathbf{P}_1}$ und $\overrightarrow{\mathbf{P}_3\mathbf{P}_2}$. Unterschreitet dieser Winkel einen vorgegebenen Schwellwert, so wird \mathbf{P}_2 vervielfacht. Der Schwellwert selbst sollte im offenen Intervall zwischen $]35^\circ, 90^\circ[$ Grad gewählt werden. Die obere Grenze ergibt sich durch die Tatsache, dass Treppmuster einen Winkel von 90° Grad aufweisen. Um diese nicht als Spitzen zu interpretieren, muss der Schwellwert auf jeden Fall kleiner als 90° Grad gewählt werden. Die untere Grenze ergibt sich dadurch, dass ein zu kleiner Schwellwert über zu viele Merkmale hinweg geht, was eine Buchstabenerkennung erschwert. Es hat sich ein Schwellwert von 45° Grad als praktikabel erwiesen.

Die Vervielfachung des Punktes selbst wirkt sich bei der B-Spline Erzeugung als eine Gewichtung des zu approximierenden Knotens aus. Die Folge ist ein Knick der B-Spline an diesem Knoten. Je häufiger der Punkt vervielfältigt wird umso stärker der Knick im Kurvenverlauf. Die Stetigkeit in diesen Kurvenpunkten nimmt ab, so bedeutet

eine Verdoppelung eines Knotens, bei kubischen B-Splines, eine Reduktion von C^2 auf C^1 Stetigkeit.

2.6.2 B-Spline Berechnung

Zu gegebenen $(n+1)$ Abtastpunkten $\mathbf{P}_0, \dots, \mathbf{P}_n \in \mathbb{R}^2$ entstehen somit n Teilsegmente. Es wird versucht eine möglichst glatte Spline durch die Kurvenpunkte zu legen, indem das folgende Integral minimiert wird:

$$F(D^m \mathbf{S})(t) = \int_{t_0}^{t_n} D^m \mathbf{S}(u)^2 du$$

Hierbei bezeichne D^m die m -te Ableitung der Spline \mathbf{S} . Der Wert von m ergibt sich aus der Ordnung der Spline. Desweiteren wird der Abstand der Spline in Bezug zu den Datenpunkten durch

$$E_x(S_x) = \sum_{j=0}^n (x_j - S_x(t_j))^2$$

$$E_y(S_y) = \sum_{j=0}^n (y_j - S_y(t_j))^2$$

für jede Richtung $E_x(S_x)$ und $E_y(S_y)$ gemessen. Der mittlere quadratische Fehlerabstand $E(S) := E_x(S_x) + E_y(S_y)$ soll sich in einer vorgegebenen Toleranz bewegen

$$E(S) < \text{tol}$$

In dieser Arbeit wurde die Toleranz im Bezug zur Größe des Segments gewählt, genauer durch

$$\text{tol} := \text{Höhe}(\text{Segment})$$

Dies hat den Vorteil, dass verschieden grosse Segmente etwa gleiche Glättung erhalten. Somit ergibt sich die gesamte Kurvenberechnung der Spline \mathbf{S} aus der Minimierung der Formel

$$\min \left((E_x(S_x) + E_y(S_y)) + \begin{bmatrix} F(D^m(S_x)) \\ F(D^m(S_y)) \end{bmatrix} \right)$$

Für genauere Details wird auf die Quellen [8] und [7] verwiesen.

3 Neuabtastung nach Bogenlänge

In diesem Abschnitt, wird die nachträgliche Neuabtastung nach Bogenlänge beschrieben. Zuerst soll eine allgemeine Einführung in die Definition der Bogenlänge stattfinden, anschließend wird eine approximative Methode zur Umparametrisierung vorgestellt werden.

Sei $I \subseteq \mathbb{D}$ so existiert zu jeder parametrisierten, regulären und differenzierbaren Kurve $\mathbf{S} : I_{[a,b]} \rightarrow \mathbb{R}^2$ eine orientierungstreue Umparametrisierung $\tilde{\mathbf{S}}(s) : I_s \rightarrow \mathbb{R}^2$ nach der Bogenlänge (siehe hierzu [12]). Es gilt:

Definition

Sei $\mathbf{S} : I_{[a,b]} \rightarrow \mathbb{R}^2$ eine parametrisierte und differenzierbare Kurve, welche zusätzlich regulär ist d.h. es gilt für alle $t \in I$ $\mathbf{S}'(t) \neq 0$. So ist die Bogenlänge der Kurve definiert durch

$$s_a(b) = \int_a^b \|\mathbf{S}'(u)\| du$$

Wie schon in Abschnitt 2.3 bemerkt wurde, besitzt die umparametrisierte Kurve $\tilde{\mathbf{S}}(s)$ die besondere Eigenschaft, dass die Länge des Geschwindigkeitsvektors konstant 1 ist $|\tilde{\mathbf{S}}'(s)| \equiv 1$.

Ohne Einschränkung der Allgemeinheit wird im weiteren davon ausgegangen, dass der lokale Kurvenparameter aus dem Intervall $t_k \in [0, 1]$ ist. Somit ist es notwendig zur Umparametrisierung nach der Bogenlänge bei Splines das Integral

$$s(t_k) = \int_0^1 \sqrt{S'_{x,k}(u)^2 + S'_{y,k}(u)^2} du \quad \text{für } t_k \in [0, 1]$$

zu berechnen. Bedenkt man, dass für eine spätere Berechnung des Merkmals 'normalized curvature' die Spline mindestens eine Ordnung größer als 4 besitzen muss (siehe Abschnitt 4.4), so wird klar, dass eine symbolische Berechnung durch den Rechner in vielen Fällen nicht möglich ist. Es bleibt nur die Alternative einer approximativen Umparametrisierung nach Bogenlänge, wie zum Beispiel in [11] beschrieben wird. Die Grundidee wurde weitestgehend aus [11] übernommen, es wurden jedoch einige Modifikationen durchgeführt, welche in den nachfolgenden Abschnitten genauer beschrieben sind.

3.1 Approximative Umparametrisierung nach Bogenlänge

Jedes Kurvensegment \mathbf{S}_k wird getrennt voneinander nach Bogenlänge umparametrisiert. Es wird versucht, mit möglichst wenigen Berechnungen eine normierte Längenkurve des Kurvensegments \mathbf{S}_k zu erzeugen. Zur Erzeugung der Längenkurve wird der lokale Parameter $\hat{t}_k \in [0, 1]$ ins Verhältnis zu seiner normierten Bogenlänge $\hat{s}(\hat{t}_k) := \frac{s(\hat{t}_k)}{s(1)}$ gesetzt. Um die Berechnung von $\hat{s}(\hat{t}_k)$ so gering wie möglich zu halten wird das Integral mithilfe der adaptiven Simpson Quadratur numerisch berechnet. Bei dieser Methode kann eine Genauigkeitsforderung angegeben werden, nach welcher die Berechnung abgebrochen wird. Das Verfahren selbst basiert auf der Keplerschen Fassregel.

Die Grundidee des Verfahrens lässt sich wie folgt beschreiben: Sei $[a, b]$ das zu integrierende Intervall. Berechne

$$I(f) := \int_a^b f(x) dx$$

über die Fassregel und schätze den Fehler ab. Falls der Fehler klein genug ist, breche die Berechnung ab, andernfalls unterteile $[a, b]$ in die zwei Hälften $[a, \frac{a+b}{2}]$ und $[\frac{a+b}{2}, b]$ und fahre rekursiv fort. Zur Durchführung dieses adaptiven Verfahrens wird eine Fehlerabschätzung benötigt. Die Simpson Summe eignet sich hierzu recht gut:

$$\begin{aligned} \text{Fassregel:} \quad & J(f)_{[a,b]} = \frac{b-a}{6} (f(a) + 4 \cdot f(\frac{a+b}{2}) + f(b)) \\ \text{Simpson Summe:} \quad & \hat{J}(f)_{[a,b]} = J(f)_{[a, \frac{a+b}{2}]} + J(f)_{[\frac{a+b}{2}, b]} \end{aligned}$$

Auf diese Art und Weise kann zum Beispiel für 3 verschiedene Parameter $\{t_{k,1} = 0, t_{k,2} = \frac{1}{3}, t_{k,3} = \frac{2}{3}, t_{k,4} = 1\}$ (beachte $S(t_{k,1}) \equiv 0 \quad \forall k$) die Bogenlänge approximativ berechnet werden. Diese Punkte werden wiederum durch eine kubische Splinekurve S_k interpoliert. Da diese Spline die Längenkurve repräsentiert, wird sie im weiteren als

$$L_k(t_k) := S_k(t_k)$$

bezeichnet. In Abbildung 8 wurden exemplarisch die Längenkurven von 4 Kurvensegmenten (Segment 7-10) eingezeichnet. Es ist zu beobachten, dass bei starker Krümmung im Kurvenverlauf die Längenkurven grosse Abweichungen gegenüber der 1. Hauptdiagonalen, welches der Längenkurve einer nach Bogenlänge parametrisierten Kurve entspricht (siehe Abbildung 6 (a)), aufweisen. Es sei darauf hingewiesen, dass es sich bei den Längenkurven um Splinefunktion handelt und nicht um Splinekurven in Parameterform.

3.2 Neuabtastung mithilfe der Bogenlängenkurve

Will man nun äquidistante Punkte mit einem Abstand r auf dem k -ten Kurvensegment erzeugen, so geschieht dies wie folgt: Man berechnet

$$t_{k,u} = L_k\left(\frac{u \cdot r}{s(1)}\right) \quad \text{für } u \cdot r \in [0, s(1)], \quad u \in N$$

und setzt die erhaltenen Funktionswerte (Parameterwerte) $t_{k,u}$ in das Kurvensegment S_k ein. Die berechneten Punkte sind äquidistant bezüglich S_k .

Um die Neuabtastung in Verhältnis zur Größe des Buchstabens zu setzen, wurde das vertikale Ausmaß

$$a_y = |\max(y) - \min(y)|$$

des Segments in Bezug gesetzt. Der Abstand d ergibt sich somit aus $d = 0.05 \cdot a_y$, also 5% der Gesamthöhe. Dieser Wert liefert genügend Abtastpunkte und die Merkmalsberechnung wird nicht zu aufwendig. Siehe Abbildung 9 für einen direkten Vergleich zwischen den unterschiedlichen Resultaten einer Neuabtastung.

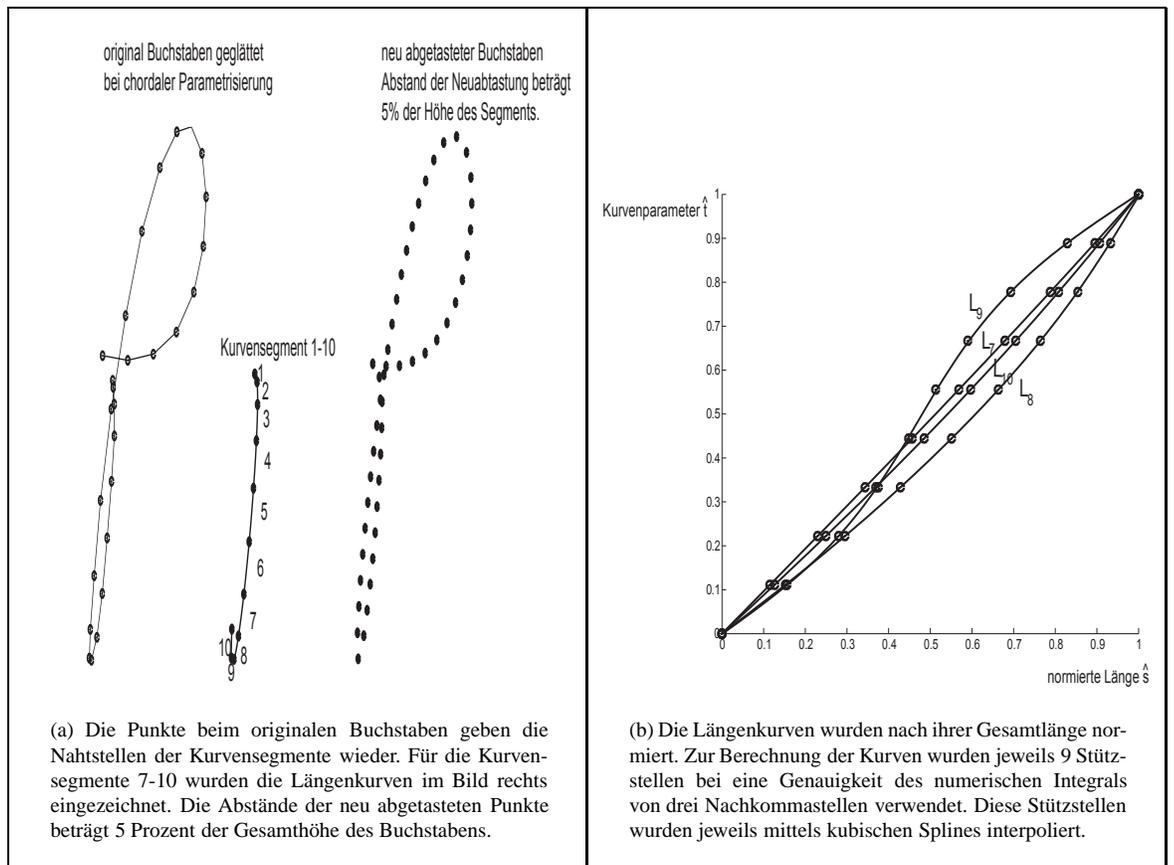


Abbildung 8: Neuabastung mithilfe der Bogenlängenkurve bei chordaler Parametrisierung.

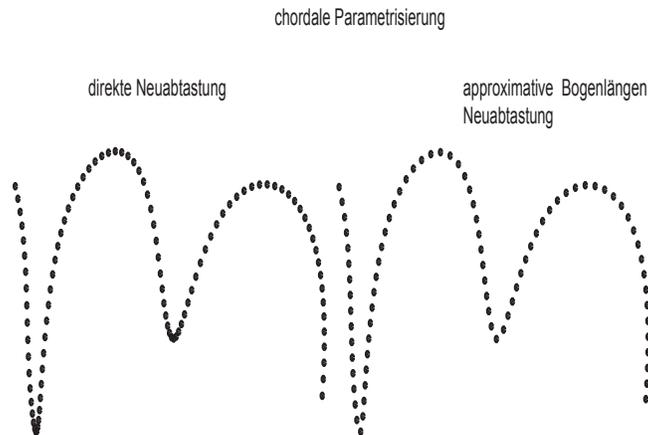


Abbildung 9: Vergleich der direkten Neuabtastung und der approximativen Neuabtastung nach Bogenlänge, bei chordaler Parametrisierung der Splinekurve. Es sind deutlich Unterschiede in der Äquidistanz der Abtastpunkte bei der direkten Neuabtastung sichtbar.

3.3 Fehlerbetrachtung

Betrachtet man die Fehler beim Berechnen der Bogenlänge, so ergeben sich zwei Quellen.

1. Beim Berechnen des numerischen Integrals. Um hier die Genauigkeit zu verbessern ist es notwendig die Rekursionschritte zu erhöhen. Hierdurch wäre es möglich, den Fehler so klein wie irgend gewünscht zu machen. Dies bedeutet aber zugleich eine enorme Erhöhung des Rechenaufwands einzig für die Berechnung einer Bogenlänge.
2. Bei der Erzeugung der Längenkurven. Die Genauigkeit der Längenkurven hängt entscheidend von der Anzahl berechneter Stützstellen ab. Es ist klar, dass es auch hier zu einer Erhöhung des Rechenaufwands kommt, will man den Fehler minimal halten. Da die Längenkurven jedoch maximal zwei Wendepunkte aufweisen, kann eine Stützstellenanzahl von 7 Punkten, welche gleichverteilt auf dem Intervall gewählt sind, als hinreichend angesehen werden (siehe [11]).

Durch die Abhängigkeit beider Fehlerquellen ist es notwendig, einen effizienten Mittelweg zu finden. Der Weg, der in dieser Arbeit verfolgt wurde, beinhaltet die Berechnung des numerischen Integrales mit Hilfe der Simpson Regel bis zu einer Genauigkeit von $1.0e - 2$ und einer Anzahl von 7 Stützstellen, welche über das Intervall $[0, 1]$ gleichverteilt sind.

4 Merkmalsberechnung

In diesem Abschnitt werden unterschiedliche Merkmale vorgestellt und es wird auf deren Berechnung näher eingegangen. War es zuvor notwendig, Merkmale über die Bildung des Differenzenquotienten zu bestimmen (siehe [2] und [1]), so ist es nun möglich, diese Merkmale über Ableitungen der Spline zu errechnen. Desweiteren erlaubt die Splinedarstellung neue Merkmale, wie in [6] beschrieben, für die Klassifikation zu gewinnen. Diese Merkmale haben die Eigenschaft gegenüber Translation, Rotation und Skalierung invariant zu sein, womit sie für die Handschrifterkennung interessant sind. Es wird darauf hingewiesen, dass das Merkmale 'signed ratio of tangents', mithilfe der Splinedarstellung, und nicht wie in [6] beschrieben über lineare Interpolation, berechnet wird. Zumal der direkte Weg über die Splinedarstellung eine genauere Berechnung des Merkmals zur Folge hat. Hätte man die Möglichkeit, die Splinekurve symbolisch, wie in Abschnitt 3 Einleitung beschrieben, nach ihrer Bogenlänge umzuparametrisieren, so würden sich bei der Berechnung der Merkmale die in Abschnitt 2.3 beschriebenen Vereinfachungen ergeben. Da dies nicht der Fall ist, muss bei der Berechnung weiterhin von einer beliebigen Parametrisierung ausgegangen werden. Um zwischen original und den neu abgetasteten Ortskoordinaten unterscheiden zu können, wird folgende Variablenbezeichnung eingeführt:

1. original abgetastete Koordinaten

$$\mathbf{P}_k = \begin{pmatrix} x_k \\ y_k \end{pmatrix} \quad \text{für } 0 \leq k \leq n$$

2. neu abgetastete Koordinaten

$$\tilde{\mathbf{P}}_j = \begin{pmatrix} \tilde{x}_j \\ \tilde{y}_j \end{pmatrix} \quad \text{für } 0 \leq j \leq m$$

Zur Visualisierung der neuen Merkmale sind zwei Abbildungen hilfreich. In der ersten Abbildung 11 wurden unterschiedliche Merkmale am Beispiel einer liegenden Acht eingezeichnet. Diese und weitere künstliche Figuren bildeten den Ausgangspunkt für die Handhabung von Spezialfällen bei der Merkmalsdefinition. Abbildung 12 zeigt schließlich die Berechnung der Merkmale, unter Verwendung der erarbeiteten Sonderregelungen, am Beispiel der Ziffer 3.

4.1 Normierte Ortskoordinaten $\mathbf{P}_k^{(\text{norm})}$ beziehungsweise $\tilde{\mathbf{P}}_j^{(\text{norm})}$

Bei der Berechnung der normierten Ortskoordinaten wird die horizontale/vertikale Koordinate in Bezug zum Mittelwert μ_x bzw. μ_y und der vertikalen Standardabweichung σ_y des Schriftzuges gesetzt:

$$\mathbf{P}_k^{(\text{norm})} := \begin{pmatrix} \frac{x_k - \mu_x}{\sigma_y} \\ \frac{y_k - \mu_y}{\sigma_y} \end{pmatrix}$$

gleiches gilt für die neu abgetasteten Koordinaten.

$$\tilde{\mathbf{P}}_j^{(\text{norm})} := \begin{pmatrix} \frac{\tilde{x}_j - \tilde{\mu}_x}{\tilde{\sigma}_y} \\ \frac{\tilde{y}_j - \tilde{\mu}_y}{\tilde{\sigma}_y} \end{pmatrix}$$

4.2 Tangentensteigungswinkel θ

Das Merkmal berechnet den Winkel θ zwischen der Horizontalen und dem Tangentenvektor in den Abtastpunkten. Dieses Merkmal hat, wie in [6] bemerkt wurde, eine grössere Diskriminierungseigenschaft wie die Merkmale 'signed ratio of tangents' und 'normalized curvature', was durch eine Gewichtung bei der Klassifikation berücksichtigt werden sollte.

Berechnung des Merkmals über Bildung des Differenzenquotienten bezüglich den originalen Ortskoordinaten.

$$\text{tangent}(P_k) = \begin{pmatrix} x_{k-1} - x_{k+1} \\ y_{k-1} - y_{k+1} \end{pmatrix}$$

$$\theta(P_k) = \text{angle}(\text{tangent}_x(P_k) - i \cdot \text{tangent}_y(P_k)) \quad i := \sqrt{-1}$$

Berechnung des Merkmals über die Splinedarstellung bezüglich den neu abgetastete Ortskoordinaten. Sei $i := \sqrt{-1}$ und angle die komplexe Winkelfunktion so gilt:

$$\mathbf{S}'(t_j) = \begin{pmatrix} S'_x(t_j) \\ S'_y(t_j) \end{pmatrix}$$

$$\tilde{\theta}(t_j) = \text{angle}(S'_x(t_j) - i \cdot S'_y(t_j))$$

4.3 Signed ratio of tangents R_α

Dieses Merkmal wird auf der Splinekurve berechnet. Hierzu wird der Tangentenvektor $\tilde{\mathbf{T}}_j$ im Abtastpunkt $\tilde{\mathbf{P}}_j$ errechnet und der korrespondierende Tangentenvektor $\tilde{\mathbf{T}}^*$ welcher die Bedingung $\angle(\tilde{\mathbf{T}}_j, \tilde{\mathbf{T}}^*) = \alpha$ erfüllt, gesucht. Existiert $\tilde{\mathbf{T}}^*$ so wird der zugehörige Punkt $\tilde{\mathbf{P}}^*$ und der Schnittpunkt $\tilde{\mathbf{P}}$ berechnet. Hieraus ergibt sich R_α wie folgt:

$$R_\alpha(t_k) = \text{sign}(\kappa(t_k)) \frac{\|\tilde{\mathbf{P}} - \tilde{\mathbf{P}}^*\|}{\|\tilde{\mathbf{P}}_j - \tilde{\mathbf{P}}\|}$$

Alpha kann beliebig fest gewählt werden, muss bei der Berechnung jedoch gegeben sein.

Falls α zu klein gewählt wird, so wird das Merkmal anfällig gegenüber Rauschen, wird es hingegen zu groß gewählt, so ist das Merkmal zu global und es werden wichtige

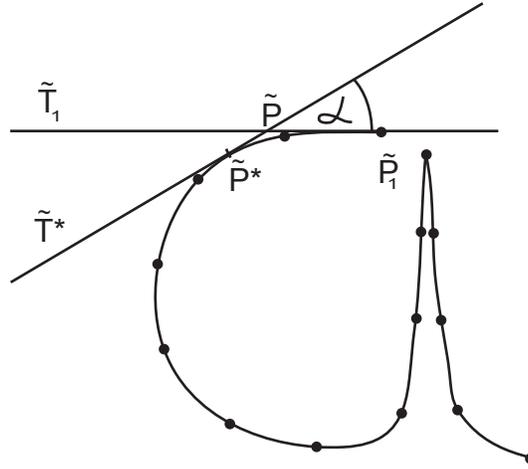


Abbildung 10: Merkmalsberechnung von 'signed ratio of tangents'

$$R_\alpha(t_k) = \text{sign}(\kappa(t_k)) \frac{\|\tilde{\mathbf{P}} - \tilde{\mathbf{P}}^*\|}{\|\tilde{\mathbf{P}}_j - \tilde{\mathbf{P}}\|}$$

lokale Formmerkmale nicht erfasst. In dieser Studienarbeit wurde $\alpha = 10$ gewählt, was der Empfehlung in [6] entspricht.

Um das Merkmal aussagekräftiger zu machen, wird es um das Vorzeichen der Krümmung im Punkt $\tilde{\mathbf{P}}_j$ erweitert. Die Krümmung κ für eine beliebige Parametrisierung an der Stelle t_j berechnet sich aus

$$\kappa(t_j) = \frac{S'_x(t_j) \cdot S''_y(t_j) - S'_y(t_j) \cdot S''_x(t_j)}{(S'_x(t_j)^2 + S'_y(t_j)^2)^{\frac{3}{2}}}$$

Zur Bestimmung des Vorzeichens ist nur der Zähler notwendig.

Für die Berechnung des Merkmals anhand der Splinedarstellung siehe Anhang A.

4.4 Normalized curvature χ

Bei diesem Merkmal ist es notwendig, dass die Spline bis zur dritten Ableitung stetig differenzierbar ist. Diese Voraussetzung ist erst ab quintischen Splines gegeben, was eine Gesamtapproximation der Abtastpunkte durch quintische Splines notwendig macht. Das Merkmal selbst berechnet sich aus der Krümmung κ und deren Ableitung κ' wie folgt:

$$\chi(t_k) = \frac{\kappa'(t_k)}{\kappa(t_k)^2}$$

hierbei berechnet sich κ' aus κ durch Anwendung der Ketten- und Quotientenregel. Zur besseren Lesbarkeit wird definiert:

$$x := S_x(t_k) \quad \text{und} \quad y := S_y(t_k)$$

$$\kappa = \frac{x' \cdot y'' - y' \cdot x''}{(x'^2 + y'^2)^{\frac{3}{2}}}$$

$$\kappa' = \frac{x' \cdot y''' - y' \cdot x'''}{(x'^2 + y'^2)^{\frac{3}{2}}} - \frac{3(x'' \cdot y''(x'^2 - y'^2) + x' \cdot y'(y''^2 - x''^2))}{(x'^2 + y'^2)^{\frac{5}{2}}}$$

Durch die Eigenschaft, dass höhere Ableitungen anfällig gegenüber Rauschen sind, ist es notwendig, die Abtastdaten 'rauschfrei' zu glätten. Dies bedeutet, dass die Spline möglichst gleichmäßige Übergänge in ihrem Krümmungsverhalten ausweisen muss. Zugleich bedeutet aber eine zu starke Glättung eine Reduktion der Schriftinformation, weshalb eine adaptive Glättung besonders wichtig wird.

Konvergiert die Krümmung gegen Null, so ergeben sich bei diesem Merkmal sehr schnell grosse Werte, wodurch die Klassifikation negativ beeinflusst wird. Es wurde deshalb eine Obergrenze von 500 festgelegt, welche sich aus der Studie verschiedener Segmente als sinnvoll erwies.

Enthält die Spline Geradenstücke, deren Krümmung Null ist, so ist das Merkmal nicht definiert. In diesem Fall wurde ebenfalls der Wert 500 festgelegt. Zusammenfassend ergibt sich folgende Definition, welche hauptsächlich im Hinblick auf den Klassifikator gewählt wurde:

$$\chi(t_k) := \begin{cases} 500 & \text{für } \frac{\kappa'(t_k)}{\kappa(t_k)^2} \geq 500 \quad \text{oder} \quad \kappa(t_k) = 0 \\ \frac{\kappa'(t_k)}{\kappa(t_k)^2} & \text{sonst} \end{cases}$$

5 Ergebnisse

Nachfolgend werden einige Ergebnisse vorgestellt, welche mithilfe des in [2] und [1] unter der Bezeichnung "statistical dynamic time warping" (SDTW) beschriebenen Klassifikators erzielt werden konnten. Die Klassifikationen wurden auf unterschiedlich segmentierten Datensätzen des UNIPEN Datensatzes durchgeführt. Die Datensätze sind nach Ziffern, Großbuchstaben und Kleinbuchstaben unterteilt. Jeder dieser Datensätze besteht aus einer Menge an Trainings- und Testdaten, welche unabhängig und je nach Auftretenswahrscheinlichkeit des Buchstabens in der englischen Sprache zusammengestellt ist. Zur Klassifikation wurde jeweils nur ein prozentualer Anteil der gesamten Trainings- und Testdaten verwendet.

Durch die Vielzahl an Parametereinstellungen, die am Klassifikator als auch bei der Merkmalsextraktion vorgenommen werden konnten, war es notwendig, die besten Einstellungen systematisch durch Annäherung an den kleinsten Fehlerwert auszuloten. Es werden im folgenden jeweils nur die besten Erkennungswerte verglichen, welche sich bei individuellen Einstellungen ergaben.

Eine experimentelle Untersuchung der Merkmale 'signed ratio of tangents' und 'normalized curvature' erwies sich wegen folgenden Schwierigkeiten als problematisch und verbleibt somit als eine reizvolle zukünftige Aufgabe:

- Das Ausloten der Klassifikationsparameter gestaltete sich für diese Merkmale als sehr umfangreich.
- Einige Modellannahmen, die bei der Entwicklung des zugrundegelegten Klassifikators verwendet wurden, treffen für diese Merkmale nicht zu.

Zur Untersuchung dieser Merkmale ist somit eine umfangreiche Modifikation des Klassifikators notwendig, welche zunächst die Charakteristiken des Klassifikators und anschließend die Merkmale einander anpasst.

Folgende Untersuchungen mit dem Klassifikator erwiesen sich als aussagekräftig:

1. Untersuchung des Einflusses der Vorverarbeitung auf die Erkennungsrate.
2. Untersuchung der Neuabtastungsmethoden im Hinblick auf eine effiziente Handschrifterkennung.

5.1 Einfluß der Neuabtastung auf die Erkennungsrate

1. Großbuchstaben (A-Z)

Trainingsdaten_gesamt = 28 069

hiervon wurden 66% für das Training benutzt

Train._Testdaten	# Testdaten	Neuabtastung	Merkmale	Fehlerrate in %	#Modelle
0.66_0.33	9051	keine	$\mathbf{P}_k^{(norm)}, \theta$	7,23	269
0.66_0.33	9051	direkt	$\tilde{\mathbf{P}}_j^{(norm)}, \tilde{\theta}$	7,48	225
0.66_0.33	9051	approx. Bogen	$\tilde{\mathbf{P}}_j^{(norm)}, \tilde{\theta}$	7,75	220

2. Kleinbuchstaben (a-z)

Trainingsdaten_gesamt = 61 351

hiervon wurden 20% für das Training benutzt

Train._Testdaten	# Testdaten	Neuabtastung	Merkmale	Fehlerrate in %	#Modelle
0.2_0.1	6071	keine	$\mathbf{P}_k^{(norm)}, \theta$	10,69	237
0.2_0.1	6071	direkt	$\tilde{\mathbf{P}}_j^{(norm)}, \tilde{\theta}$	11,71	192
0.2_0.1	6071	approx. Bogen	$\tilde{\mathbf{P}}_j^{(norm)}, \tilde{\theta}$	13,52	97

3. Zahlen (0-9) [Zeitangaben bezüglich AMD Duron 1.2 MHz mit 1GHz RAM]

Trainingsdaten_gesamt = 15 953

hiervon wurden 66% für das Training benutzt

train._Testdaten	#Testdaten	Neuabtastung	Merkmale	Fehlerrate [%]	#Modelle	Gesamtzeit [Std]	Klassifikationszeit pro Zeichen [Sek]
0.66_0.33	5181	keine	$\mathbf{P}_k^{(norm)}, \theta$	3,28	186	3.5	0.7
0.66_0.33	5181	direkt	$\tilde{\mathbf{P}}_j^{(norm)}, \tilde{\theta}$	4,38	58	5.5	0.9
0.66_0.33	5181	approx. Bogen	$\tilde{\mathbf{P}}_j^{(norm)}, \tilde{\theta}$	4,46	72	21	3.5

5.1.1 Einfluss der Vorverarbeitung auf die Erkennungsrate

Eine Vergleich der Klassifikationsergebnisse zeigt, dass die Erkennung ohne Vorverarbeitung gegenüber der Erkennung mit Vorverarbeitung (Glättung bzw. Neuabtastung) bessere Ergebnisse erzielt. Dies kann daraufhin zurückgeführt werden, dass durch die Glättung der Abtastdaten eine Informationsreduktion stattfindet, wodurch jedoch auf der anderen Seite die Anzahl der Modelle, welche für die Klassifikation eines Buchstabens benötigt werden, teilweise drastisch reduziert wird. Um dies an einem Beispiel zu erläutern, seien die Kleinbuchstaben i und l in Druckbuchstabenschrift gegeben. Die Verwechslungsrate des Klassifikators zwischen diesen Buchstaben ist relativ groß gemessen an anderen Buchstaben. Dies liegt hauptsächlich daran, dass der Kleinbuchstaben i meist keinen zugehörigen i-Punkt besitzt. Durch die Glättung der beiden Buchstaben kommt es nun zur weiteren Angleichung. Im Extremfall bestehen beide Buchstaben nur noch aus einer Geraden. Hierbei können auch die neuen Merkmale 'signed ratio of tangents' und 'normalized curvature' keinen Beitrag, für einer richtigen Zuordnung der Buchstaben, mehr leisten. Da sie, im Fall einer Geraden, nur festgelegte konstante Werte zurückliefern können. Es wäre also notwendig, diese Buchstaben im Schriftkontext zu untersuchen, wodurch eine zusätzliche Diskriminierung über das Größenverhältnis der Buchstaben möglich wäre. Zusammenfassend kann gesagt werden, dass die Anzahl der Modelle durch eine Vorverarbeitung reduziert werden konnten, jedoch Einbußen in der Erkennungsleistung akzeptiert werden mussten.

5.1.2 Vergleich der Neuabtastungsmethoden

Die beiden Arten der Neuabtastung unterscheiden sich nicht nur in Berechnungsmehraufwand und Neuabtastungsgenauigkeit, sondern auch in der Fehlerrate. So konnten mithilfe der direkten Neuabtastung bei chordaler Parametrisierung bessere Ergebnisse erreicht werden, als bei approximativer Neuabtastung nach Bogenlänge. Dies spricht eindeutig gegen die rechenaufwendigere Methode der approximativen Neuabtastung. Deshalb sollte im Hinblick auf eine Online-Handschrifterkennung die direkte Neuabtastung bevorzugt werden. Um einen groben Eindruck von der benötigten Rechenzeit zu bekommen, seien folgende absoluten Daten genannt:

Die Berechnungen der Zeichenklassifikationen fanden auf einem AMD DURON 1.2 MHz mit 1 GHz Hauptspeicher statt. Die Ergebnisse der Zahl-Klassifikation enthalten ungefähre Zeitangaben, welche auf Grundlage des angegebenen Rechnermodells ermittelt wurden. Unter der Gesamtzeit wird die Zeit zur Berechnung der Modelle auf Grundlage der Trainingsdaten plus der Klassifikationszeit der Testdaten verstanden. Bei der Klassifikationszeit pro Buchstaben wurde die Zeit berechnet, welche zur Klassifikation eines Testdatums (Buchstabens) benötigt wurde.

6 Zusammenfassung und Ausblick

In der vorliegenden Arbeit wurde ein Weg aufgezeigt, wie die Merkmale 'signed ratio of tangents' und 'normalized curvature' durch Splinerepräsentation aus Online- Hand-

schriftdaten berechnet werden können. Gerade das Merkmal 'normalized curvature', zu dessen Berechnung die dritte Ableitung benötigt wird, machte es notwendig, die Abtastdaten mithilfe quintischen Splines zu glätten. Bei der Untersuchung von Splines haben sich B-Splines durch ihre positiven Eigenschaften als geeignete Lösung herauskristallisiert. So war es möglich eine adaptive Glättung der Segmente durchzuführen, während weiterhin eine effiziente Berechnung der Spline Darstellung gewährleistet war. Zudem verhalf die Modellierung von Knicken in der Splinekurve, dem Erhalt wichtiger Merkmale bei der Klassifikation. Durch die parametrische Darstellung der Spline war es notwendig, eine geeignete Parametrisierung zu finden, welche im Hinblick auf eine Neuabtastung optimal ist. Es wurden zwei Methoden vorgestellt, welche sich in Effizienz und Genauigkeit unterschieden. Die chordale Parametrisierung mit anschließender direkter Neuabtastung wies zwar grössere Abweichungen in der Äquidistanz auf, zeigt jedoch positivere Ergebnisse bei der Klassifikation. Durch die Eigenschaft, grössere Abtastdichten in Kurvenverläufen mit starker Krümmung zu produzieren, wurden Kurven mit einer stärkeren Merkmalsgewichtung versehen als Geradenstücke, welches bei den erzielten Klassifikationsergebnissen durchaus als positiv bewertet werden konnte. Zusätzlich wurde ein approximatives Verfahren vorgestellt, welches eine Neuabtastung mit äquidistanten Abtastpunkten ermöglicht. Doch selbst bei schneller, und damit ungenauerer Berechnung der approximativen Bogenlänge, ist diese Methode für eine Online-Handschrifterkennung zu langsam und somit nicht praxistauglich. Weiterhin ist festzuhalten, dass die Merkmale selbst stark glättungsabhängig sind. Liegt ein Schriftdatensatz mit grossen Qualitätsunterschieden vor, so ist es schwer eine einheitliche Glättung für alle Datensätze zu finden. Dies konnte gerade bei der Verwendung des UNIPEN Datensatzes festgestellt werden, welcher durch seine grosse Inhomogenität (Aufzeichnung der Daten mithilfe unterschiedlicher Erfassungsgeräte) an Online-Daten, den Einsatz einer festen Glättungskonstanten nicht ermöglichte. Betrachtet man die Instabilität der Merkmale gegenüber unterschiedlichen Glättungsstärken, so ist es zwingend notwendig ein Glättungsalgorithmus zu entwickeln, welcher die unterschiedlichen Datensegmente einheitlich berücksichtigt. Die Glättungstoleranz, welche in dieser Arbeit verwendet wurde, sollte somit durch ein adaptives Verfahren verbessert werden. Deshalb wäre es im Hinblick auf weiterführende Untersuchungen sinnvoll, folgende Wege einzuschlagen:

1. Entwurf eines Glättungsalgorithmus, welcher die Qualität und Dichte der abgetasteten Schriftdaten stärker berücksichtigt.
2. Optimale Anpassung des Klassifikators an die Merkmale.
3. Untersuchung der Merkmale anhand eines homogenen Datensatzes von Online-Handschriften.
4. Gewichtung der Merkmale bei der Klassifikation wie in [6] beschrieben ist. Mithilfe dieser Untersuchungen sollte es möglich sein, weitere Aussagen über die Klassifikationsgüte der Merkmale treffen zu können.

7 Danksagung

An dieser Stelle will ich mich bei der sehr guten Betreuung durch meinen Mentor Herrn Klaus Bahlmann bedanken, welcher mir bei der Gestaltung und Korrektur dieser Studienarbeit zur Seite stand.

A Berechnung von 'signed ratio of tangents' über Splinedarstellung

Um effektiv die zugehörige Tangente $\tilde{\mathbf{T}}^*$ bzw. den Parameter t^* zu berechnen, wird eine Nullstellenberechnung durchgeführt. Bei der Suche nach t^* wird über die einzelnen Teilpolynome der Splinekurve iteriert. Es wird mit dem j -ten Teilpolynom gestartet, indem sich der Kurvenparameter t_j befindet. Im j -ten Teilpolynom wird die Nullstelle im Intervall $[t_j, 1]$ gesucht, beim Übergang zu einem neuen Teilpolynom wird auf dem ganzen Parameterintervall $[0, 1]$ nach einer Nullstelle gesucht. Es ergibt sich folgende Nullstellenberechnung für das j -te Teilpolynom für $j \leq h \leq n$

$$\cos(\alpha) = \frac{\langle \tilde{\mathbf{T}}_j, \tilde{\mathbf{T}}^* \rangle}{\|\tilde{\mathbf{T}}_j\| \cdot \|\tilde{\mathbf{T}}^*\|}$$

$$\Rightarrow \cos(\alpha) = \frac{\langle \tilde{\mathbf{T}}_j, \begin{pmatrix} S'_{x,h}(t^*) \\ S'_{y,h}(t^*) \end{pmatrix} \rangle}{\|\tilde{\mathbf{T}}_j\| \cdot \|\tilde{\mathbf{T}}^*\|}$$

$$\Rightarrow 0 = \frac{S'_{x,j}(t_j) \cdot S'_{x,h}(t^*) + S'_{y,j}(t_j) \cdot S'_{y,h}(t^*) - \cos(\alpha) \cdot \sqrt{S'_{x,j}(t_j)^2 + S'_{y,j}(t_j)^2} \cdot \sqrt{S'_{x,h}(t^*)^2 + S'_{y,h}(t^*)^2}}{\dots}$$

Existiert eine Nullstelle, so ist der gesuchte Kurvenparameter t^* durch die kleinste positive Nullstelle definiert. Hiermit ergeben sich die benötigten Vektoren $\tilde{\mathbf{P}}^*$, $\tilde{\mathbf{P}}$ und $\tilde{\mathbf{T}}^*$ durch:

$$\tilde{\mathbf{P}}^* = \begin{pmatrix} S_x(t^*) \\ S_y(t^*) \end{pmatrix}$$

$$\tilde{\mathbf{T}}^* = \begin{pmatrix} S'_x(t^*) \\ S'_y(t^*) \end{pmatrix}$$

$$\tilde{\mathbf{P}} = \text{Schnittpunkt}(\tilde{\mathbf{T}}, \tilde{\mathbf{T}}^*)$$

Existiert hingegen $\forall h$ keine Nullstelle so wird

$$R_\alpha(t_j) = \begin{cases} 0 & \text{für } j = 0 \\ R_\alpha(t_{j-1}) & \text{sonst} \end{cases}$$

gesetzt.

Diese Überlegung ergibt sich aus der Untersuchungen des Merkmals an künstlichen Figuren (siehe Abbildung 11). Die Idee war, um ein Gefühl für diese Merkmal zu bekommen, den signed ratio of tangents an unterschiedlichen Figuren zu berechnen. Es wurden folgende Figuren verwendet:

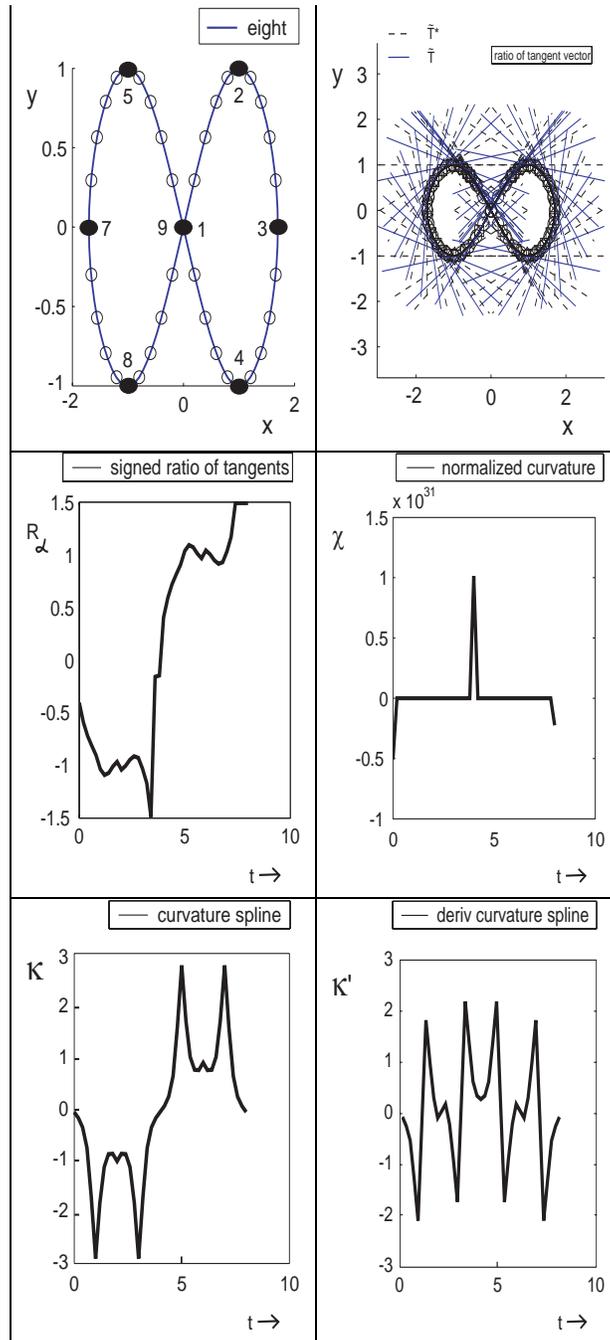


Abbildung 11: Berechnung verschiedener Merkmale am Beispiel der Acht. Anfangs und Endpunkt der Acht liegen im Koordinatenursprung. Zur Erzeugung der Acht, wurden die Punkte $[(0,0),(1,1),(1.7,0),(1,1),(0,0),(1,1),(1.7,0),(1,1),(0,0)]$ mittels einer kubischen Spline interpoliert. Der Kurvenparameter wurde nach der Indexstellen-Parametrisierung gewählt $t \in [0, 1, \dots, 8]$

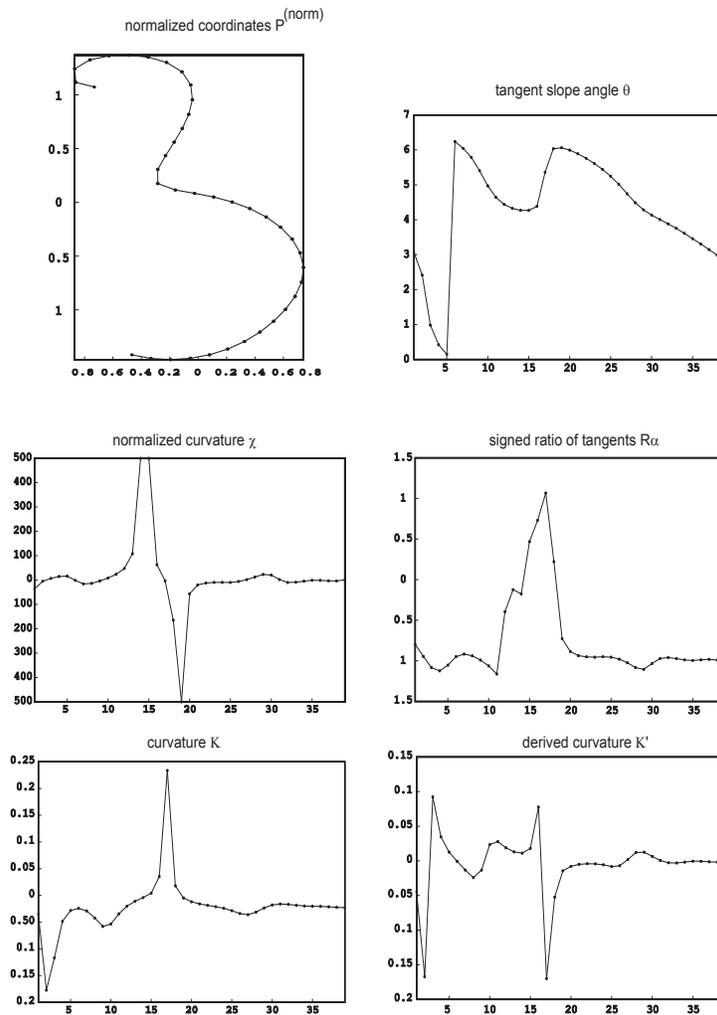


Abbildung 12: Es ist die Berechnung der aufgeführten Merkmale am Beispiel der Ziffer '3' dargestellt. Hierbei wurde eine Indexstellen-Parametrisierung verwendet, wodurch der Kurvenparameter $t \in [0..#Abtastpunkte]$ ist. Für die Neuabtastung wurde eine approximative Neuabtastung nach Bogenlänge gewählt mit einer Genauigkeit des numerischen Integral von $1.e-2$, während 4 Stützstellen für die Erzeugung der Längenkurven berechnet wurden. Die Neuabtastung besitzt einen äquidistanten Abstand, welcher 5% der Gesamthöhe der Ziffer entspricht. Das Merkmal normalized curvature wurde auf den Wertebereich zwischen $[+500, -500]$ eingeschränkt. Beim Merkmal signed ratio of tangents wurden die Vorgängerwerte übernommen, sobald das Merkmal nicht mehr berechnet werden konnte (siehe Endbereich des Graphen).

- Kreis mit Radius $r \in \mathbb{R}$ $t \in [0, 2\pi]$

$$\phi(t) = (r \cdot \cos(t), r \cdot \sin(t))$$

- Ellipse mit $a \neq b \in \mathbb{R}$ $t \in [0, 2\pi]$

$$\phi_{a,b}(t) = (a \cdot \cos(t), b \cdot \sin(t))$$

- Spitze $t \in \mathbb{R}$ mit $t \in [-1, 1]$

$$\phi(t) = (t^3, t^2)$$

- logarithmische Spirale $t \in [0, 12\pi]$ $b \in \mathbb{R}$

$$\phi_b(t) = (\exp(b \cdot t) \cdot \cos(t), \exp(b \cdot t) \cdot \sin(t))$$

- Acht $t \in [0, 2\pi]$ (siehe Abbildung 11)

$$\phi(t) = (\sin(t), \cos(t) \cdot \sin(t))$$

Es hat sich bei der Untersuchung gezeigt, dass das Merkmal am Ende eines Segments nur selten berechnet werden kann. Da die Splinekurve zumeist relativ flach ausläuft. Um nun Merkmale für diese Werte dem Klassifikator übergeben zu können, welche realistisch sind, wurden die Vorgängerwerte übernommen. Liegt hingegen eine Gerade vor, so existiert in keinem Abtastpunkt der 'signed ratio of tangents'. Um diese Situation zu beschreiben, wurde das Merkmal auf den Wert 0 gesetzt. Dieser Wert kann als Funktionswert nicht auftreten, da die Längen beider Strecken echt grösser Null sein müssen. Somit werden Geraden bei der Klassifikation eindeutige Werte zugewiesen.

Literatur

- [1] BAHLMANN, C. und H. BURKHARDT: *Measuring HMM Similarity with the Bayes Probability of Error and its Application to Online Handwriting Recognition*. In: *Proc. of the 6th ICDAR*, S. 406–411, 2001.
- [2] BOCKHORN, D.: *Bestimmung und Untersuchung von Signifikanzgewichtungen für die Erkennung von handgeschriebenen Buchstaben*. Diplomarbeit, Universität Freiburg, Institut für Informatik, 2000.
- [3] CLAUS, V. und A. SCHWILL (Hrsg.): *Duden Informatik: Ein Fachlexikon für Studium und Praxis*. Dudenverlag: Mannheim-Leipzig -Wien-Zürich, 3. Aufl., 2001.
- [4] DE BOOR, C.: *A Practical Guide to Splines*. Springer Verlag, New York, 1978.
- [5] GUYON, I., L. SCHUMAKER, R. PLAMONDON, M. LIBERMAN und S. JANET: *Unipen project of on-line data exchange and recognizer benchmarks*. In: *Proceedings of the 12th ICPR, Jersuaem, Israel*, S. 29–33, Okt. 1994.
- [6] HU, J., M. K. BROWN und W. TURING: *Invariant Features for HMM Based On-Line Handwriting Recognition*. In: *The 8th International Conference on Image Analysis and Processing*, S. 588–593, San Remo, Italy, Sep. 1995.
- [7] REINSCH, C. H.: *Smoothing by spline functions*. In: *Numerische Mathematik*, Bd. 10(3), S. 177–183. Feb. 1967.
- [8] REINSCH, C. H.: *Smoothing by spline functions II*. In: *Numerische Mathematik*, Bd. 16(5), S. 451–454. März 1971.
- [9] SCHUMAKER, L.: *The Unipen Project* <http://unipen.nici.kun.nl>, 1998.
- [10] STROMER, K.: *Effiziente Berechnung und Darstellung von Freiformkurven und Freiformflächen in der Computergraphik*. Diplomarbeit, Technische Universität München, Fakultät für Mathematik (TUM), 1996. Dipolmarbeit.
- [11] WALTER, M. und A. FOURNIER: *Approximate Arc Length Parametrization*. In: *Proceedings of the 9th Brazilian Symposium on Computer Graphics and Image Processing*, S. 143–150, Caxambu, Minas Gerais, Brazil, Okt. 1996.
- [12] WÜNSCH, V.: *Differentialgeometrie: Kurven und Flächen*. Teubner Verlag, Stuttgart-Leipzig, 1. Aufl., 1997.