technische universität
dortmund

Masterarbeit

Word Spotting mit Online-Handschrift Anfragen

Christian Wieprecht
July 2015

Überarbeitete Fassung

Gutachter:
Prof. Dr.-Ing. Gernot A. Fink
Dipl. Inf. Leonard Rothacker

Technische Universität Dortmund
Fakultät für Informatik
Lehrstuhl 12
http://ls12-www.cs.tu-dortmund.de

INHALTSVERZEICHNIS

MATHEMATISCHE NOTATION

Typ	Beispiel	Notation
Konstante	W, L	Großbuchstaben
Matrix	$\mathbf{A}, \mathbf{B}$	Großbuchstaben, **fett**
Menge	$\mathcal{C}, \mathcal{T}$	Skriptbuchstaben
Parameter, Winkel	α, β, ζ	griechische Buchstaben, klein
Skalar	a, b, C_i, L_i	Kleinbuchstaben, Großbuchstaben mit Index
Vektor	$\mathbf{t}$	Kleinbuchstaben, **fett**

ABKÜRZUNGSVERZEICHNIS

BoF	Bag-of-Features
BoF+SP	Bag-of-Features mit Spatial Pyramid
BoOF	Bag-of-Online-Features
BoOF+SP	Bag-of-Online-Features mit Spatial Pyramid
CSR	Common Subspace Regression
FV	Fisher Vektor
FV+K	Fisher Vektor mit Koordinatenerweiterung
LSA	Latent Semantic Analysis
OCR	Optical Character Recognition
PCA	Principal Component Analysis
QbE	Query-by-Example
QbO	Query-by-Online-Trajectory
QbS	Query-by-String
SVM	Support Vector Machine

1 EINLEITUNG

Word Spotting ist ein Verfahren zum Durchsuchen von digitalen Archiven von Dokumenten. Die Dokumente liegen dabei typischerweise in einer gescannten Fassung vor. Ein Anwender eines Word Spotting Verfahrens hat zwei Möglichkeiten eine Suchanfrage zu stellen. Zum einen kann eine solche Anfrage durch Markieren des gesuchten Wortes in einem beliebigen Dokument aus dem Archiv gestellt werden. Dies wird als *Query-by-Example* bezeichnet, da ein Beispiel des gesuchten Wortes als Anfrage dient. Die zweite Möglichkeit ist die Eingabe einer Zeichenkette über eine Tastatur. Dies wird als *Query-by-String* bezeichnet und ist flexibler, da zum einen dadurch ein beliebiges Wort angefragt werden kann und zum anderen der Anwender nicht erst ein Beispiel dieses Wortes suchen muss.

Das zentrale Ziel dieser Arbeit ist die Evaluierung einer weiteren Eingabemodalität. Durch die fortschreitende Verbreitung von berührungsempfindlichen Bildschirmen (Touchscreens) entwickeln sich auch die Anforderungen an die Benutzerschnittstellen, durch die ein Anwender mit einem Gerät bzw. einer Software interagiert. Daher wird als neue Möglichkeit die Eingabe per Handschrift auf berührungsempfindlichen Oberflächen vorgeschlagen. Der Benutzer schreibt hierbei das gesuchte Wort per Finger oder mit Hilfe eines speziellen Touchscreen-Eingabestifts auf eine geeignete Oberfläche. Dies wird als *Query-by-Online-Trajectory* bezeichnet.

Das restliche Kapitel ist wie folgt aufgebaut. Zunächst werden in Abschnitt 1.1 Word Spotting Verfahren motiviert und von Verfahren zur Transkription von Texten abgegrenzt. Zudem wird die neue Query-by-Online-Trajectory Methode durch beispielhafte Einsatzszenarien erklärt und motiviert. Abschnitt 1.2 beschreibt zuerst die übliche Vorgehensweise zum Aufbau eines Word Spotting Systems und geht anschliessend kurz auf die Umsetzung der in dieser Arbeit evaluierten Systeme ein. Zum Abschluss des Kapitels wird in Abschnitt 1.3 ein Überblick über den weiteren Aufbau der vorliegenden Arbeit gegeben.

1.1 MOTIVATION

Word Spotting beschreibt die Aufgabe, zu einer gegebenen Anfrage relevante Vorkommen von Wörtern aus Dokumentbildern zu ermitteln. Ein Treffer ist dabei ein Bildausschnitt, ein sogenanntes Wortbild, welcher das gesuchte Wort repräsentiert. Word Spotting wurde als Alternative zu Texterkennungs Verfahren vorgeschlagen,

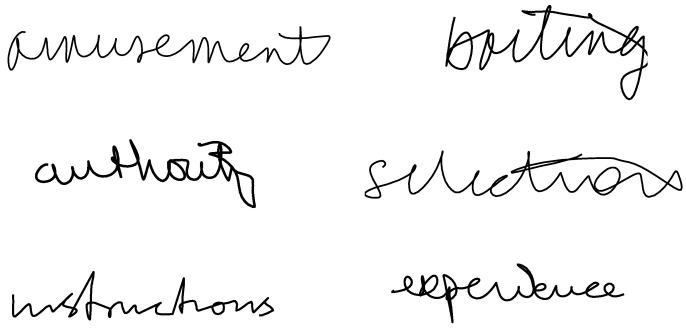
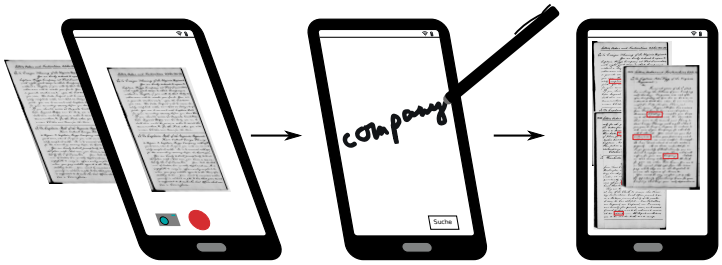


Abbildung 1: Beispiele für Variationen in der Handschrift von mehreren Schreibern aus dem Unipen-Datensatz [Uni] und dem GWO-Datensatz (siehe Abschnitt 7.1.2).

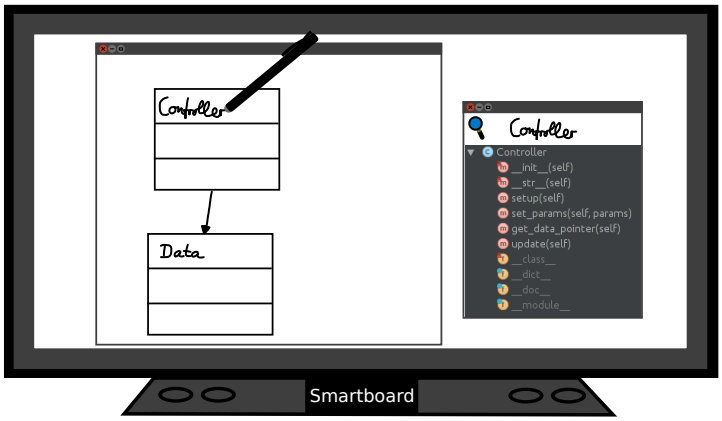
um mit den Texten arbeiten zu können, auf denen diese Verfahren schlechte Ergebnisse liefern oder eine vollständige Transkribierung, d.h. Übersetzen in Maschinenschrift, des vorliegenden Dokuments nicht benötigt wird.

Das Transkribieren von Texten aus Bildern ist ein schwieriges Problem. Besonders die Transkription von handgeschriebenen Texten ist allerdings in Abhängigkeit vom Zustand des Dokuments oftmals nicht oder nur in eingeschränkten Kontexten zufriedenstellend möglich [Vin02, PF09]. Gründe dafür sind zum Teil die Qualität der vorliegenden Dokumentbilder, aber auch ausgebleichte Schrift, Tintenflecken und schlechte Papierqualität des fotografierten oder gescannten Dokuments. Vor allem bereitet allerdings die Variabilität der Handschrift von einem oder mehreren Schreibern große Schwierigkeiten für visuelle Texterkennungs-Verfahren (Optical Character Recognition, OCR). Ein Beispiel für diese Variabilität ist in Abbildung 1 gezeigt. Gleiche Buchstaben unterscheiden sich teilweise deutlich zwischen unterschiedlichen Handschriftstilen. Auch in der Handschrift eines einzelnen Schreibers wird Variabilität dadurch erzeugt, dass dieser Schreiber mit großer Wahrscheinlichkeit Vorkommen des gleichen Buchstaben nicht exakt gleich schreibt.

Word Spotting Verfahren versuchen viele der oben genannten Probleme zu umgehen, indem sie ein vorliegendes Dokument nicht vollständig transkribieren. Stattdessen werden relevante Vorkommen von Wörter in Dokumentbildern anhand visueller Erkennungsmerkmale gesucht. Der Nachteil dieser Vorgehensweise für handschriftliche Dokumente ist, dass sie sich nur bedingt für den Einsatz auf Dokumenten mehrerer Schreiber eignet, da durch unterschiedliche Handschriftstile ein großes Maß an Variabilität in das Schriftbild gelangt. Dies erschwert das Erkennen relevanter Wortbilder nur anhand visueller Merkmale. Word Spotting Verfahren sind allerdings unter Umstän-



(a)



(b)

Abbildung 2: Zwei Beispielanwendungen von Word Spotting mit Online-Handschrift Anfragen. (a) Durch ein Smartphone oder Tablet werden Fotos von Dokumentseiten aufgenommen. Der Benutzer stellt durch handschriftliche Eingabe eine Suchanfrage, welche durch das Word Spotting Verfahren beantwortet wird (Dokumentseiten aus [GW]). (b) UML-Diagramm Modellierung auf einem Smartboard. Für eine bereits bestehende Klasse „Controller“ werden über Word Spotting Details aus dem Programmcode herausgesucht.

den im Vergleich zu OCR-Verfahren effizienter und können auch mit wenig Trainingsdaten gute Erkennungsleistungen erzielen [FKFB12].

In bestehenden Word Spotting Verfahren wird das gesuchte Wort vom Anwender entweder durch Markieren eines Wortbildes oder durch durch Eingabe einer Zeichenkette angegeben. Was ist aber mit Situationen, in denen keine dieser Herangehensweisen geeignet ist? In dieser Arbeit wird dazu eine neue Methode vorgestellt: das Word Spotting mit Online-Handschrift Anfragen. Bei dieser Art der Anfrage wird das gesuchte Wort durch den Benutzer direkt per Handschrift auf einem Touchscreen oder ähnlichem Gerät geschrieben. Dies ist gerade in Situationen sinnvoll, in denen das Einblenden einer virtuellen Tastatur als störend empfunden wird oder nicht gewünscht ist. Auf bestimmten Tablets, die, wie z.B.

die Samsung Galaxy Note Reihe, für Handschrifteingaben ausgelegt sind und mit entsprechendem Touchscreen-Stift ausgeliefert werden, ist es beispielsweise intuitiver, die Eingabe über die primäre Eingabeschnittstelle, den Touchscreen-Stift, zu tätigen, als eine virtuelle Tastatur einzublenden.

Diese neue Form des Word Spotting wird *Query-by-Online-Trajectory* (QbO) genannt. Einige weitere Einsatzszenarien für QbO sind in Abbildung 2 gezeigt. Die obere Abbildung zeigt eine Smartphone Applikation, mit deren Hilfe handschriftliche Aufzeichnungen durchsucht werden können, ohne diese Transkribieren zu müssen. Dazu nimmt der Benutzer Fotos der zu durchsuchenden Dokumente auf und stellt anschliessend eine handschriftliche Suchanfrage. Mit dieser Anfrage aus der Online-Handschrift Domäne werden die Fotos der Dokumente (visuelle Bilddomäne) durchsucht. Die untere Abbildung zeigt die Modellierung von UML-Diagrammen an einem Smartboard. Das kollaborative Design von Software ist ein aktuelles Forschungsthema (vgl. [WG05, BB12]). Dabei wird beispielsweise eine Software von mehreren Personen an einem Smartboard oder berührungsempfindlichem Tisch modelliert. Das entstehende „Tafelbild“ wird üblicherweise auch in einer digitaler Form benötigt, in der die enthaltenen Informationen leicht zugänglich sind. Dies bedeutet oft, dass Handschrift transkribiert wird. Da das Tafelbild in seiner Struktur allerdings unrestriktiv ist – an jeder Stelle des Smartboards kann in beliebiger Ausrichtung geschrieben werden – ist Transkribierung oft schwierig. Durch Anwendung von Word Spotting mit Online-Handschrift Anfragen ist an dieser Stelle keine Transkribierung nötig. Ein geschriebenes Wort kann über ein solches Verfahren beispielsweise direkt als Anfrage für schon bestehende Dokumente (z.B. Informationen zu Klassen aus dem Programmcode) verwendet werden.

1.2 VORGEHENSWEISE

Word Spotting Verfahren folgen der typischen Vorgehensweise eines Information Retrieval Systems (vgl. [BYRN99], Kap. 1). Diese wird im Folgenden anhand Abbildung 3 erläutert. In einer Vorverarbeitung werden die Dokumentbilder dabei für die weiteren Schritte vorbereitet. Dies beinhaltet beispielsweise die Binarisierung aller Dokumentbilder oder die Reduzierung von ungewollter Variabilität in der dargestellten Schrift. Weiterhin wird zwischen segmentierungsfreien und segmentierungsbasierten Verfahren unterschieden. Segmentierungsbasierte Verfahren auf Wortebene extrahieren alle Wortvorkommen aus den Dokumentbildern und arbeiten im Folgenden nur noch auf diesen Wortbildern. Zeilenbasierte Verfahren extrahieren alle Zeilen der Dokumente. Segmentierungsfreie Verfahren arbeiten auf den vollständigen Dokumentseiten. Dabei werden häufig sogenannte Patches gebildet, indem ein Fenster über eine Dokumentseite gescho-

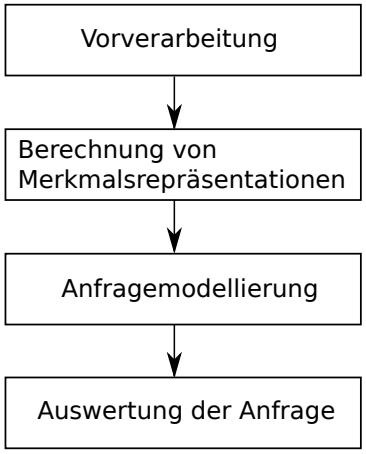


Abbildung 3: Grob verallgemeinerte Darstellung der Vorgehensweise von Word Spotting Verfahren.

ben wird, dessen Inhalt als potenzielles Wortbild gehandhabt wird. Der nächste Schritt ist die Berechnung von Merkmalsrepräsentationen für Wortbilder oder Patches. Merkmalsrepräsentationen beinhalten Eigenschaften der Wortbilder, die deren visuelle Ähnlichkeiten oder Bedeutungsunterschiede zueinander numerisch erfassen. Dies geschieht beispielsweise durch Berechnen von charakteristischen lokalen Bildausschnitten, durch die verschiedene Buchstaben oder Wörter unterschieden werden können. Bei der Anfragemodellierung wird ein Modell zur Erkennung und Unterscheidung der im vorherigen Schritt berechneten Merkmalsrepräsentationen gebildet. Wie dieser Schritt und das Modell in der Praxis umgesetzt werden, ist abhängig von der Art des jeweiligen Word Spotting Verfahrens. Die einfachste Möglichkeit besteht im direkten Vergleich von zwei Merkmalsrepräsentationen. Alternativ wird ein Modell anhand von Beispieldaten gelernt, welches den Vergleich der Merkmalsrepräsentationen durchführt. Hierbei wird unterschieden zwischen überwachtem Lernen, bei dem für die gegebenen Beispieldaten die dargestellten Wörter bekannt sind, und unüberwachtem Lernen, bei dem dies nicht der Fall ist. Die abschliessende Auswertung der Anfrage erstellt die Ausgabe des Word Spotting Systems. Dabei wird eine sortierte Liste von Suchergebnissen zurückgegeben. Dabei sollten Suchergebnisse, die tatsächlich relevant zur Anfrage sind, möglichst weit vorne in dieser Liste platziert sein.

Die in dieser Arbeit vorgestellten Word Spotting Verfahren mit Online-Handschrift Anfragen arbeiten segmentierungsbasiert. Somit werden für segmentierte Wortbilder und die Online-Handschrift Anfrage Merkmalsrepräsentationen berechnet. In der Dokumentenanalyse haben sich in den letzten Jahren Verfahren aus der Computer Vision etabliert, um Merkmalsrepräsentationen zu bestimmen. Im Speziellen werden in dieser Arbeit Bag-of-Features Repräsentationen (vgl.

[OD11]) für Wortbilder verwendet. Diese werden aus Beispieldaten gelernt und bilden eine Statistik über typischerweise auftretende, lokale Bildausschnitte der Wortbilder. Der Bag-of-Features Ansatz wird in dieser Arbeit zum ersten Mal auch für Online-Handschrift verwendet, um eine Merkmalsrepräsentation zu berechnen. Durch die Möglichkeit der Verwendung von Bag-of-Repräsentationen auf verschiedenen Merkmalsdomänen (visuell, textuell, Online-Handschrift), können Methoden des Maschinellen Lernens verwendet werden, um Abbildungen zwischen diesen Domänen zu berechnen. Dadurch wird domänenübergreifendes Word Spotting möglich.

Das Ziel dieser Arbeit ist die Evaluierung einer Auswahl von vielversprechenden Word Spotting Verfahren mit Online-Handschrift Anfragen. Dazu werden die Query-by-String Verfahren aus [ARTL13] und [AGFV14a] für Anfragen mit Online-Handschrift angepasst. Diese Anpassungen umfassen zum einen die Repräsentation der handschriftlichen Eingabe durch geeignete Merkmale aus dem Bereich der Online-Handschrift Erkennung und zum anderen die Berechnung von Abbildungen, mit denen domänenübergreifend mittels Online-Handschrift die Suche nach Wörtern in Bildern von Dokumenten durchgeführt werden kann.

1.3 AUFBAU DER ARBEIT

Die Arbeit ist wie folgt aufgebaut. In Kapitel 2 werden grundlegende Methoden erläutert, welche zum Verständnis der evaluierten Word Spotting Verfahren nötig sind. Kapitel 3 beschreibt Vorverarbeitungs- und Bearbeitungsschritte, welche Merkmale aus Online-Handschrift Trajektorien berechnen, die die unterscheidenden Eigenschaften dieser Trajektorien erfassen. In Kapitel 4 wird der Bag-of-Features-Ansatz erläutert – eine Methode um Merkmalsrepräsentationen für bildbasierte Daten zu berechnen. Durch diese Merkmalsrepräsentation werden die Wortbilder der Dokumente beschrieben, die durch das Word Spotting Verfahren durchsucht werden sollen. Kapitel 5 beschreibt relevante Arbeiten aus der Literatur. Es werden sowohl Query-by-Example Verfahren, als auch Query-by-String Verfahren vorgestellt und ihre Eigenschaften, auch im Hinblick auf das Word Spotting mit Online-Handschrift Verfahren, untersucht. Zwei besonders geeignete Verfahren werden für das Query-by-Online-Trajectory Word Spotting adaptiert. Dies ist in Kapitel 6 erläutert. Zu den durchgeführten Anpassungen gehört auch eine neue Merkmalsrepräsentation für Online-Handschrift Trajektorien mittels des Bag-of-Features Ansatzes. Die vorgestellten Verfahren werden in Kapitel 7 optimiert und evaluiert. Zudem werden die Ergebnisse der Evaluierung für den Einsatz der Word Spotting Verfahren in der Praxis bewertet. Das abschliessende Kapitel 8 fasst die vorgestellten Verfahren und Evaluierungsergebnisse zusammen.

2

GRUNDLAGEN

Dieses Kapitel beschreibt einige methodische Grundlagen, welche zum Verständnis der in dieser Arbeit verwendeten Word Spotting Verfahren notwendig sind. Die Abschnitte dienen dabei der kurzen Einführung des jeweiligen Themas. Für vollständige Erläuterungen sei an den entsprechenden Stellen jeweils an die angegebene Literatur verwiesen.

In der Einleitung wurde eine Übersicht, über die typischen Schritte eines Word Spotting Verfahrens gegeben. Abschnitt 2.1 beschreibt die Vektorquantisierung, die bei der Berechnung von Merkmalsrepräsentationen von Wortbildern zum Einsatz kommt. Die Latent Semantic Analysis (Abschnitt 2.2) ist eine Methode, um Merkmalsrepräsentationen unterschiedlicher Domänen (z.B. textuell und visuell) miteinander vergleichen zu können. Support Vektor Maschinen (Abschnitt 2.3) sind Klassifikatoren, welche in einem weiteren Word Spotting Verfahren ebenfalls zum Vergleich unterschiedlicher Merkmalsrepräsentationen genutzt werden.

2.1 VEKTORQUANTISIERUNG

Bei der in der Einleitung (Kapitel 1) beschriebenen Herangehensweise von Word Spotting Verfahren wird für jedes Wortbild eine Merkmalsrepräsentation berechnet. Wie in Kapitel 4 zu sehen sein wird, werden in dieser Arbeit dafür aus einem Wortbild an bestimmten Positionen Vektoren extrahiert, welche numerische Ausprägungen von visuellen Merkmalen beinhalten. Dabei ist von Interesse, welche dieser Vektoren ähnlich zueinander sind, da die Positionen solcher Vektoren ähnliche Stellen des Wortbildes kennzeichnen. Um diese Ähnlichkeit zu überprüfen, wird aus einer Menge dieser Vektoren ein sogenanntes Kodebuch erstellt, eine festgelegte Anzahl von Vektoren, welche typische Bildausschnitte des Wortbildes beschreiben. Zwei extrahierte Vektoren werden dann als eine ähnliche Bildstelle beschreibend behandelt, wenn sie von allen Vektoren des Kodebuchs zu dem gleichen Vektor die geringste Distanz aufweisen.

Die Folgenden Erläuterungen orientieren sich an [Fino8], Kap. 4. Die Berechnung des Kodebuchs $\mathcal{C}$ erfolgt anhand eines Quantisierers Q. Sei $\mathcal{T} = \mathbf{t}_1, \mathbf{t}_2, \ldots, \mathbf{t}_N$ eine Menge von Vektoren, aus denen ein Kodebuch geschätzt werden soll. Dabei wird eine Quantisierungsregel gefunden, durch die der Quantisierer allen Vektoren $\mathbf{t}_i$ genau den Kodebuch-Vektor $Q(\mathbf{t}_i)$ zuweist, der von allen Kodebuch-Vektoren zu

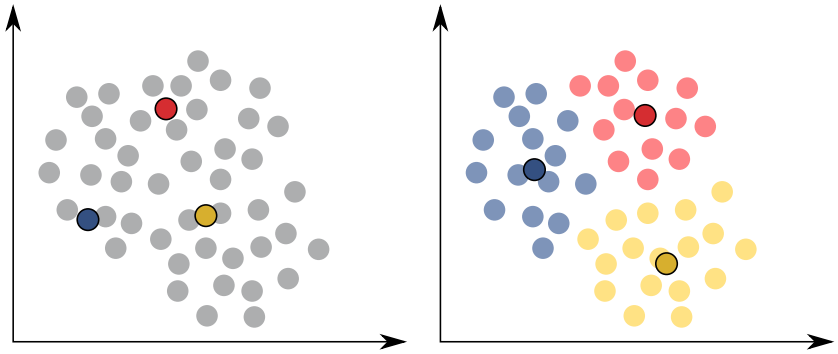


Abbildung 4: Vektorquantisierung mit dem Algorithmus von Lloyd. Dargestellt sind die Initialisierung und ein mögliches Ergebnis des Algorithmus. (a) Eine Menge von Vektoren (grau) soll in K disjunkte Mengen geclustert werden. Dafür werden K zufällige Clusterzentren erstellt (farbig, schwarz umrandet). (b) Durch abwechselndes Neuzuweisen der Vektoren zum nächsten der K Clusterzentren und anschliessendem Neuberechnen dieser Zentren wird das finale Kodebuch berechnet.

$\mathbf{t}_i$ den kleinsten Abstand aufweist. Das Quantisieren von $\mathcal{T}$ hat zum Ziel, eine kleinere Menge von $|\mathcal{C}| = K$ Vektoren (das Kodebuch) zu finden, welche die Beispieldaten in $\mathcal{T}$ ausreichend genau beschreiben. Je näher K an der ursprünglichen Anzahl an Vektoren liegt, desto genauer wird diese Menge beschrieben. Durch die Quantisierungsregel wird eine Partitionierung von $\mathcal{T}$ erreicht. Die Bestimmung der Quantisierungsregel erfolgt durch zwei Bedingungen:

1. **Nächster Nachbar Bedingung**. Für ein gegebenes Kodebuch $\mathcal{C}$ wird jeder Vektor $\mathbf{t}_i \in \mathcal{T}$ genau dem Vektor $Q(\mathbf{t}_i) \in \mathcal{C}$ zugewiesen, der von allen Vektoren im Kodebuch den geringsten Abstand zu $\mathbf{t}_i$ hat.

2. **Zentroiden Bedingung**. Die Kodebuch-Vektoren (Zentroiden) werden so gewählt, dass alle Vektoren $\mathbf{t}_i$, welche ihnen durch den Quantisierer zugewiesen werden, zu keinem anderen Zentroiden einen kleineren Abstand haben.

Es gibt keine analytische Lösung, welche einen optimalen Quantisierer zu beiden Regeln findet. Die Lösung erfolgt daher iterativ. Die Güte eines Kodebuchs wird durch den Quantisierungsfehler ausgedrückt. Dies ist die Summe der Abstände der Vektoren in $\mathcal{T}$ zu ihrem nächsten Nachbarn im Kodebuch. Bei der Quantisierung gilt es, diesen Fehler zu minimieren.

Die Vektorquantisierung erfolgt durch unüberwachtes Lernen mittels Clustering aus den Beispieldaten. Im Gegensatz zum überwach-

ten Lernen ist für die Beispielmenge also keine optimale Partitionierung bekannt. Beim Clustern werden die Beispieldaten in K disjunkte Mengen geteilt. Jede Menge wird durch einen Repräsentaten beschrieben, den Zentroiden. Die Menge aller Zentroiden ergibt das Kodebuch. Ein bekanntes Clustering-Verfahren ist der Algorithmus von Lloyd. Als Abstandsmaß für Vektoren wird dabei in dieser Arbeit der euklidische Abstand gewählt. Die Initialisierung und das Ergebnis dieses Verfahrens sind in Abbildung 4 schematisch dargestellt. Der Algorithmus wird mit einem Kodebuch initiiert, welches wahlweise aus K vollständig randomisierten Vektoren besteht oder K Vektoren, die zufällig aus den Beispieldaten gezogen werden. Die Vektoren des Kodebuchs sind in der Abbildung mit schwarzer Umrandung dargestellt. Bei der Berechnung des Clusterings werden zwei Schritte abwechselnd wiederholt:

- Zuweisung aller Vektoren der Beispieldaten zu ihrem nächsten Nachbarn im Kodebuch. Dadurch entstehen K Cluster.
- Neuberechnen der K Zentroiden anhand der ihnen zugewiesenen Vektoren, sodass für jeden Zentroiden der Quantisierungsfehler minimiert wird.

Dadurch werden beide oben genannten Bedingungen abwechselnd iterativ optimiert. Der Algorithmus wird durchgeführt, bis der Quantisierungsfehler geringer als ein gegebener Schwellwert ist oder eine maximale Anzahl an Durchläufen erreicht wurde.

2.2 LATENT SEMANTIC ANALYSIS

Die Latent Semantic Analysis (LSA) ist eine Methode, welche ursprünglich zur Verwendung in Aufgaben der Dokumentenanalyse bzw. des Information Retrieval entworfen wurde [DDF+90]. Dabei geht es beispielsweise um das Finden relevanter Dokumente zu einem angegebenen Stichwort oder der Suche nach weiteren Dokumenten, welche das selbe Thema wie ein angegebenes Dokument behandeln. Die Dokumentenanalyse ist ein weiter Bereich, dessen genauere Erläuterung den Rahmen dieser Arbeit übersteigt. Für genauere Details siehe beispielsweise [BYRN99].

Die LSA soll im Folgenden am Beispiel der Suche nach relevanten Dokumente zu einem gegebenen Dokument erläutert werden. Ein Dokument gilt als relevant, wenn es das gleiche Thema, wie das gegebene Dokument behandelt. Dokumente werden dabei durch Terme beschrieben. Ein Term ist ein Wort aus einem zuvor bestimmten Vokabular. In [DDF+90] werden beispielsweise alle Wörter verwendet, die in mehr als nur einem Dokument vorkommen und zudem keine Stoppwörter sind, d.h. Wörter, wie „the“ und „so“, welche keine Aussage über das Thema eines Dokuments machen. Jedes Dokument wird durch einen Termvektor repräsentiert, der

die Häufigkeit jedes Terms in diesem Dokument beschreibt. Nicht alle Dokumente des gleichen Themas verwenden jedoch die gleiche Auswahl von Termen, zudem zeigen einzelne Terme nicht nur ein Thema an. Daher wird die LSA verwendet, um anhand der Termvektoren abstrakte Merkmale (Topics) zu berechnen, welche die Korrelationen zwischen den Termen erfassen und somit die Themen der Dokumente besser signalisieren. Durch die Berechnung eines Vektorraums, des sogenannten Topic Raums, wird zudem etwaiges Hintergrundrauschen irrelevanter Wörter beseitigt. Zur Berechnung wird eine Term-Dokument-Matrix $\mathbf{X} \in \mathbb{R}^{t \times d}$ erstellt, welche in jeder Spalte den t-dimensionalen Termvektor eines der d Dokumente enthält. Bei der formalen Darstellung wird sich im Folgenden an der Vorlage aus [DDF+90] orientiert. Mit einer Singulärwertzerlegung wird diese Matrix nun in ein Produkt aus drei Matrizen

$$\mathbf{X} = \mathbf{U} \cdot \mathbf{S} \cdot \mathbf{V}^T \tag{1}$$

zerlegt. Dabei enthalten $\mathbf{U} \in \mathbb{R}^{t \times m}$ und $\mathbf{V} \in \mathbb{R}^{d \times m}$ die Eigenvektoren von $\mathbf{X}\mathbf{X}^T$ und $\mathbf{X}^T\mathbf{X}$ respektive. $\mathbf{S} \in \mathbb{R}^{m \times m}$ ist eine Diagonalmatrix, welche die m Singulärwerte enthält. An dieser Stelle wird die große Ähnlichkeit der LSA zur Hauptkomponenanalyse deutlich (PCA, vgl. [Fino8], Kap. 9.1). Die PCA arbeitet auf der Kovarianzmatrix der Daten, welche zuvor jedoch vom Mittelwert befreit werden. Dies entspricht $(\mathbf{X} - \mu)^T(\mathbf{X} - \mu)$, wenn die Daten spaltenweise in $\mathbf{X}$ vorliegen. Sowohl die PCA, als auch LSA berechnen m orthonormale und somit linear unabhängige Faktoren der Matrix $\mathbf{X}$. Wenn nun aus Gleichung 1 die $k \leq m$ größten Singulärwerte und die entsprechenden Eigenvektoren aus $\mathbf{U}$ und $\mathbf{V}$ gewählt werden, lässt sich $\mathbf{X}$ durch

$$\mathbf{X} \approx \mathbf{X}_k = \mathbf{U}_k \cdot \mathbf{S}_k \cdot \mathbf{V}_k^T \tag{2}$$

als Linearkombination der berechneten Faktoren approximieren. In diesem kleineren Vektorraum wurde die Repräsentation eines Dokuments aus t Termen durch eine Repräsentation aus k Topics ersetzt. Da jedes Topic von allen Eigenvektoren beeinflusst wird, werden in $\mathbf{U}_k$ die Korrelationen zwischen verschiedenen Dokumenten und in $\mathbf{V}_k$ die Korrelationen zwischen verschiedenen Termen erfasst. Die Topic-Raum Repräsentation von $\mathbf{X}$ ist dabei durch $\mathbf{V}_k$ gegeben. Um nun ein neues Dokument in diesem Topic Raum mit den anderen Dokumenten vergleichen zu können, muss es in diesen projiziert werden. Die Transformationsmatrix wird durch eine Umstellung von Gleichung 2 nach

$$\mathbf{V}_k = \mathbf{X}_k^T \cdot \mathbf{U}_k \cdot \mathbf{S}_k^{-1} \tag{3}$$

ersichtlich und ergibt sich aus dem Matrixprodukt $\mathbf{U}_k \cdot \mathbf{S}_k^{-1}$. Die Topic Raum Repräsentation des neuen Dokuments $\mathbf{q}$ wird demnach durch

$$\mathbf{V}_q = \mathbf{q}^T \cdot \mathbf{U}_k \cdot \mathbf{S}_k^{-1} \tag{4}$$

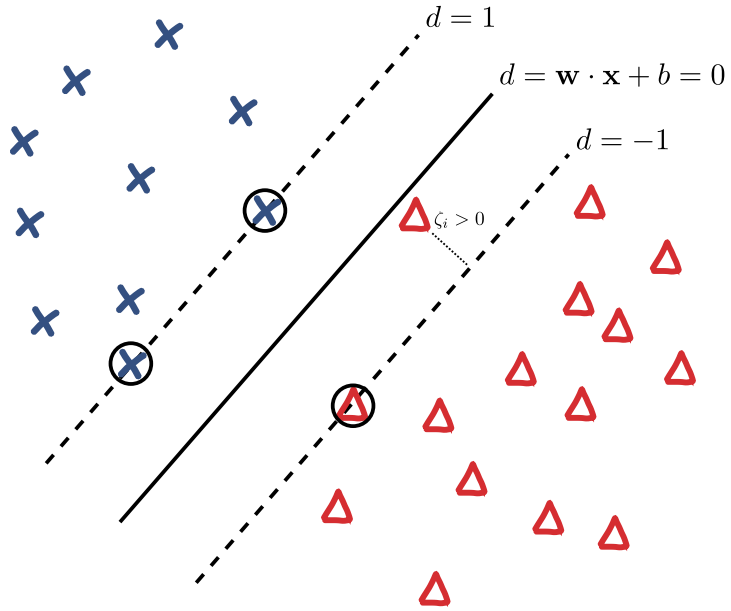


Abbildung 5: Darstellung einer Support Vektor Maschine. Die beiden Klassen (blau und rot) werden anhand der optimalen Hyperebene getrennt, welche durch die Support Vektoren bestimmt wird (eingekreiste Punkte). Für Punkte, die die lineare Trennbarkeit hindern, werden Schlupfvariablen ζ_i eingeführt, welche eine gewisse Abweichung von der exakten Trennung erlauben.

ermittelt. Im Topic Raum gehört jedes Dokument einer Auswahl von Topics an. Jedes Topic entspricht einer Richtung des Topic Raums. Bei der Transformation eines Dokuments in den Topic Raum wird durch Multiplikation mit der Inversen von $\mathbf{S}_k$ (siehe Gleichung 4) eine Normalisierung dieser Richtungen und somit der Topics vorgenommen. Die Abstandsmessung zweier Dokumente im Topic Raum geschieht daher beispielsweise anhand der Kosinusdistanz, d.h. dem Kosinus des Winkels zwischen zwei Topic Raum Repräsentationen, da Repräsentationen, zwischen denen ein kleiner Winkel liegt, einer ähnlichen Auswahl von Topics angehören.

2.3 SUPPORT VEKTOR MASCHINEN

Support Vektor Maschinen (SVM) sind lineare Klassifikatoren, welche ein Zweiklassenproblem lösen (vgl. [Bur98]). Dazu wird eine trennende Hyperebene berechnet, welche zwischen den beiden Klassen liegt und den maximal möglichen Abstand zwischen den nächsten Beispielen jeder Klasse hat. Dies ist in Abbildung 5 verdeutlicht. Bei der formalen Darstellung wird sich im Folgenden an [Bur98] orientiert. Die Hyperebene wird in ihrer Normalenform

$$\mathbf{w} \cdot \mathbf{x} + b = 0 \tag{5}$$

angegeben. Zum Berechnen der Parameter werden Beispieldaten benötigt, für die die Klassenzugehörigkeit bekannt ist, es handelt sich somit um ein überwachtes Lernen (vgl. Abschnitt 2.1). Jedem Trainingsbeispiel sei ein Label $y_i \in \{-1, +1\}$ zugewiesen, das die Zugehörigkeit zu einer der beiden Klassen anzeigt. Eine SVM wird dann bestimmt, indem das Optimierungsproblem

$$\text{argmin}_{\mathbf{w}} \frac{1}{2}||\mathbf{w}||^2 + C \cdot \sum_i \zeta_i \tag{6}$$

unter den Nebenbedingungen

$$\forall i : (\mathbf{w} \cdot \mathbf{x}_i + b) \cdot y_i \geq 1 - \zeta_i \tag{7}$$

gelöst wird, wodurch die Hyperebene mit dem maximalen Abstand von den nächsten Beispielen jeder Klasse ermittelt wird. Die Nebenbedingungen sorgen dafür, dass auf jeder Seite der Hyperebene nur Beispiele der gleichen Klasse liegen. Die Beispiele, für die die Nebenbedingung mit Gleichheit erfüllt ist, und deren Abstand von der Hyperebene somit 1 beträgt, werden Support Vektoren genannt. Durch die sogenannten Schlupfvariablen ζ_i wird ermöglicht, einzelne Aussreisser auf der „falschen“ Seite der Hyperebene zuzulassen, wenn das Klassifizierungsproblem nicht vollständig durch eine lineare Trennebene lösbar ist. Die Zulässigkeit solcher Aussreisser wird durch den Parameter C reguliert. Ein Lösungsverfahren für das in Gleichung 6 und Gleichung 7 dargestellte Optimierungsproblem ist die *Sequential Minimal Optimization* [Pla98], bei der dieses Problem in mehrere kleine quadratische Teilprobleme aufgeteilt wird, die jeweils analytisch gelöst werden.

Bei der Klassifizierung eines Beispiels $\mathbf{x}_i$ mit einer SVM wird durch $d_i = \mathbf{w} \cdot \mathbf{x}_i + b$ der Abstand (positiv oder negativ) von $\mathbf{x}_i$ zur Hyperebene berechnet. Bei der Klassifizierung wird die Klasse anhand des Vorzeichens des Abstandes eines Beispiels zugewiesen. Der Abstand wird auch als SVM-Score bezeichnet und ist insbesondere keine Wahrscheinlichkeit für die Zugehörigkeit zu einer Klasse. Ein mögliches Verfahren, um dies zu beheben ist als Platt's Scaling bekannt und wird in [Pla99] vorgeschlagen. Dabei werden die Scores einer trainierten SVM kalibriert und auf den Wertebereich $[0, 1]$ projiziert. Die neuen Werte werden als Wahrscheinlichkeit für die Klassenzugehörigkeit interpretiert. Die Kalibrierung geschieht durch Anpassung eines Sigmoiden an die SVM-Scores der Trainingsdaten über eine Maximum Likelihood Schätzung (siehe auch [LLW07]). In dieser Arbeit ist eine solche Score-Kalibrierung wichtig, da bei den vorgestellten Verfahren SVM-Scores verschiedener SVMs miteinander verglichen werden. Da der Abstand von der Hyperebene im Allgemeinen nicht beschränkt ist, kann das Ergebnis ohne Kalibrierung der Scores von den unterschiedlichen Wertebereichen beeinflusst werden.

3 REPRÄSENTATION VON ONLINE-HANDSCHRIFT

Das folgende Kapitel beschreibt die typischen Verarbeitungsschritte, mit der eine Online-Handschrift Trajektorie für die Verwendung in einem Erkennungssystem vorbereitet wird. Die einzelnen Schritte stammen aus der Literatur zur Online-Handschrift Erkennung [JMW00, LB06]. Im weiteren Verlauf dieser Arbeit werden die in diesem Kapitel beschriebenen Schritte verwendet, um eine Merkmalsrepräsentation anhand der neuen *Bag-of-Online-Trajectory* Methode (vgl. Kapitel 1) zu berechnen. Dies wird ausführlich in Kapitel 6 erläutert. Jede Online-Handschrift Trajektorie wird zunächst normalisiert, um ungewünschte Variabilität zwischen Trajektorien gleicher Wörter und Buchstaben zu reduzieren. Die Schritte der Normalisierung sind in Abschnitt 3.2 beschrieben. Im Anschluss an die Normalisierung werden für jeden Punkt der Online-Handschrift Trajektorie Merkmale berechnet, welche die Eigenschaften der Trajektorie numerisch ausdrücken. Die Merkmalsberechnung wird in Abschnitt 3.3 beschrieben.

3.1 DEFINITIONEN

Die Online-Handschrift Trajektorien, die in dieser Arbeit benutzt werden, stammen aus zwei Aufnahme-Quellen. Sie wurden auf einem Google Nexus 7 Tablet und einem Wacom 420-L Tablet [Uni] geschrieben und erfasst. Genauere Details zu den verwendeten Datensätzen werden in Abschnitt 7.1.2 beschrieben. Die im Folgenden vorgestellten Grundbegriffe sind in Abbildung 6 verdeutlicht. Eine (Online-Handschrift) Trajektorie ist im Zusammenhang dieser Arbeit der Schriftzug eines Wortes und wird durch eine Sequenz von Punkten $p_0, p_1, \ldots$ repräsentiert. Jeder Punkt p_i wird durch seine Koordinaten (x_i, y_i) im zweidimensionalen Raum dargestellt. Da die Punkte durch die Aufnahme-Hardware in regelmäßigen zeitlichen Abständen erfasst werden, ergibt sich, dass die Nachbarn eines Punktes p_i die zeitlichen, aber nicht notwendigerweise die räumlich nächsten Punkte sind. Insbesondere wenn sich die Schrift kreuzt wird deutlich, dass ein zeitlich späterer Punkt p_j räumlich beliebig viel näher zu p_i liegen kann, als dessen zeitlichen Nachbarn p_{i-1} und p_{i+1}.

Als Bounding Box einer Trajektorie wird das kleinste Rechteck bezeichnet, welches alle Punkte einer Trajektorie einschliesst. Die Bounding Box ist durch die Punkte $(x_{\min}, y_{\min})$ und $(x_{\max}, y_{\max})$ eindeutig bestimmt. Als Grundlinie oder Baseline, wird die waagerechte Linie

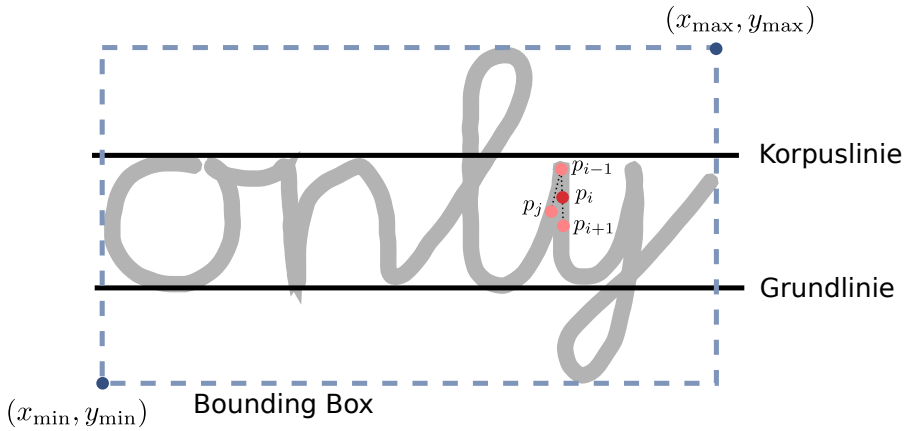


Abbildung 6: Definitionen zur Online-Handschrift. Die gesamte Trajektorie wird von der Bounding Box umrahmt. Grund- und Korpuslinie begrenzen den Korpus der Trajektorie. Die Punkte der Trajektorie werden in zeitlicher Reihenfolge erfasst, somit ist es möglich, dass ein beliebiger Punkt p_j näher an einem Punkt p_i liegt, als dessen zeitliche Nachbarn p_{i-1} und p_{i+1}.

bezeichnet auf der kleine Buchstaben, wie z. B. *o* und *n* aufliegen. Die Korpuslinie begrenzt kleine Buchstaben, wie das *y*, nach oben. Buchstaben, wie das *f* reichen sowohl unter die Grundlinie, als auch über die Korpuslinie.

3.2 NORMALISIERUNG

In diesem Abschnitt werden die einzelnen Normalisierungsschritte beschrieben, mit denen jede Online-Handschrift Trajektorie vorverarbeitet wird. Die Normalisierung ist notwendig, da besonders beim Auftreten von mehreren Schreibern die Steigung, Neigung und Größe der Handschrift stark variiert [LBo6]. Doch auch bei Datensätzen, die von einem einzelnen Schreiber stammen, wird die Variabilität gleicher Wörter durch die Normalisierung verringert. Zunächst wird die Ausrichtung und Höhe jeder Trajektorie vereinheitlicht, dies ist in Abschnitt 3.2.1 und Abschnitt 3.2.2 beschrieben. Anschliessend wird durch die Korrektur der Steigung (Abschnitt 3.2.3) und Neigung (Abschnitt 3.2.4) vom Schreiber verursachte Variablitität aus der Schrift entfernt. Durch eine Neuabtastung und anschliessende Glättung der Trajektorie (Abschnitt 3.2.5 und Abschnitt 3.2.6) wird hingegen versucht, durch die Aufnahme-Hardware verursachte Variabilität zu entfernen. In einem letzten Schritt werden Delayed Strokes aus der Trajektorie entfernt. Dies wird in Abschnitt 3.2.7 erläutert.

3.2.1 *Koordinatensystem*

Als Normalisierung des Koordinatensystems wird ein Schritt bezeichnet, welcher zum einen die Größenordnung der Koordinaten aller Trajektorien angleicht, als auch ihre Höhe und Ausrichtung. Während sich auf den Aufnahmegeräten zur Handschrifterfassung der Ursprung des Koordinatensystems üblicherweise in der oberen linken Ecke des Bildschirms befindet, wird für die meisten der in diesem Kapitel beschriebenen Normalisierungsschritte das in der Mathematik üblichen Koordinatensystem mit dem sich in der unteren linken Ecke befindlichen Ursprung verwendet. Dies wird, wenn nötig, durch eine Spiegelung der Trajektorie an der x-Achse korrigiert. Die Trajektorie wird zudem so verschoben, dass der Punkt $(x_{\min}, y_{\min})$ der Bounding Box im Ursprung liegt (vgl. Abschnitt 3.1).

3.2.2 *Höhenanpassung*

Die Normalisierung der Höhe der Online-Handschrift Trajektorie dient der Korrektur von unterschiedlichen Schriftgrößen von Schreibern [JMW00]. Die Höhe der Trajektorie wird über deren Bounding Box bestimmt (vgl. Abschnitt 3.1). Dabei wird der Abstand $y_{\max} - y_{\min}$ auf eine festgelegte Distanz skaliert. Die x-Koordinaten der Punkte werden zur Beibehaltung des Seitenverhältnisses der Trajektorie um den gleichen Faktor skaliert. Diese Methode hat den Nachteil, dass Buchstaben, die unter die Grundlinie fallen (z. B. g, y und j) und Buchstaben, die über die Korpuslinie reichen (z. B. f, h und alle Großbuchstaben) die Höhe stark beeinflussen. Eine alternative Höhenskalierung kann über die Berechnung der Grund- und Korpuslinie erfolgen. Dabei wird die Trajektorie anhand der Korpushöhe skaliert, dem vertikalen Abstand zwischen Grund- und Korpuslinie [JMW00, LB06]. Informelle Experimente haben gezeigt, dass die zuerst genannte Methode für die im Rahmen dieser Arbeit verwendeten Online-Handschrift Trajektorien gute Ergebnisse zeigt. Eine Normalisierung anhand der Korpushöhe erfolgt daher nicht.

3.2.3 *Steigung*

Die Steigung einer Trajektorie wird beschrieben durch den Winkel α, der zwischen der x-Achse und der Grundlinie eingeschlossen ist [LB06]. Zur Korrektur der Steigung wird dieser Winkel approximiert und die Trajektorie rotiert. Dieses Verfahren wird in Abbildung 7 verdeutlicht. Als Approximation der Grundlinie wird in dieser Arbeit eine lineare Regressionsgerade durch alle Punkte der Trajektorie ermittelt. Dies geschieht nach der Methode der kleinsten Quadrate (vgl. [HTF09], Kap. 3). Jeder Punkt (x_i, y_i) der Trajektorie

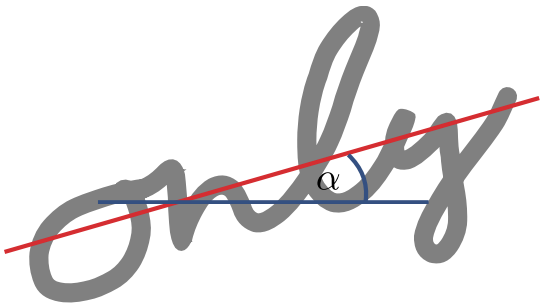


Abbildung 7: Korrektur der Steigung. Die blaue Linie zeigt die Grundlinie der Trajektorie. Die rote Linie beschreibt die lineare Regressionsgerade durch alle Punkte. Der Winkel α zwischen Grundlinie und Regressionsgerade ist die Steigung.

wird anschliessend um den Winkel α, welcher zwischen dieser Regressionsgeraden und der x-Achse liegt, gedreht.

3.2.4 *Neigung*

Die durchschnittliche Neigung eines Wortes beschreibt die Ausrichtung der normalerweise vertikal ausgerichteten Linien, wie z. B. beim Buchstaben *I*. Eine regelmäßige Neigung ist beispielsweise in *kursiver* Schrift gegeben. Da unterschiedliche Schreiber mit unterschiedlicher Neigung der Buchstaben schreiben, ist es von Vorteil, die dadurch entstehende Variabilität in der Handschrift zu korrigieren. Das ganze Verfahren ist in Abbildung 8 verdeutlicht. Dafür wird zunächst für jedes Paar aufeinander folgender Punkte der Winkel bestimmt, der zwischen der y-Achse und der Verbindungsgeraden der zwei Punkte liegt [JMW00, LB06]. Diese Winkel werden in einem Histogramm aufgetragen, welches mit einem Gauß-Filter gewichtet wird. Dabei liegt der Mittelwert der verwendeten Gauß-Verteilung bei 0° und die Anzahl der betragsmäßig größten Winkel (dies entspricht den waagerechten Linien des Schriftzuges) wird am stärksten geglättet. Dies führt dazu, dass waagerechte Geraden, wie beispielsweise der Verbindungsstrich in einem *H*, welche nicht relevant für die durchschnittliche Neigung der senkrechten Linien eines Wortes sind, weniger stark gewichtet werden. Der Neigungswinkel, um den die Trajektorie schliesslich zur Korrektur gescheert wird, entspricht dem Winkel mit dem maximalen Eintrag im Histogramm.

3.2.5 *Neuabtastung*

Eine Neuabtastung (*engl.* resampling) der Trajektorie ersetzt die vorhandene Folge von zeitlich äquidistanten Punkten durch eine neue Folge von räumlich äquidistanten Punkten [LB06]. Dadurch

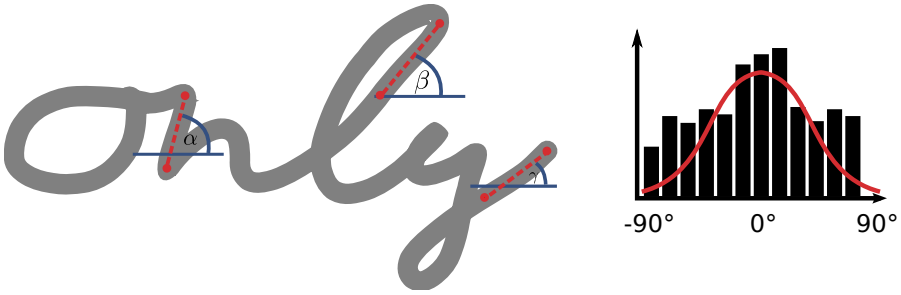


Abbildung 8: Korrektur der Neigung. Bestimmung des Winkels zwischen x-Achse (blau) und Verbindungsgerade (rot gestrichelt) aller paarweise aufeinanderfolgender Punkte. Nach Gauß-Glättung des Histogramms, Scheerung des Wortes um den Winkel mit maximaler Häufigkeit.

wird eine variable Schreibgeschwindigkeit bei der Aufnahme der Trajektorie korrigiert. Zudem kann es in Abhängigkeit der Abtastrate der Aufnahme-Hardware (siehe Abschnitt 3.1) zu fehlenden Punkten kommen, was durch die Neuabtastung ebenfalls korrigiert wird. In dieser Arbeit wurde ein einfaches Verfahren implementiert, welches durch lineare Interpolation neue Punkte an die Stellen der Trajektorie einsetzt, an denen der Punktabstand größer, als ein gewählter Schwellwert ist [PTV05].

3.2.6 *Glättung*

Die Glättung (*engl.* smoothing) der Trajektorie dient der Veringerung der Variabilität der Schrift, die durch ein Zittern des Schreibenden oder zu grobe Punkterfassung durch die Aufnahme-Hardware entsteht. Die Korrektur geschieht durch Anwendung eines Gauß-Filters, der jeden Punkt der Trajektorie mit dem gewichteten Mittelwert seiner Nachbarschaft ersetzt [JMW00].

3.2.7 *Delayed Strokes*

Als Delayed Strokes werden Striche, wie z. B. der Punkt auf einem *i* und der horizontale Strich eines *t*, bezeichnet, deren Zeitpunkt beim Schreiben eines Wortes nicht fest vorgegeben ist [JMW00]. Verschiedene Schreiber setzen diese Striche zu unterschiedlichen Zeitpunkten beim Schreiben des gleichen Wortes. Dadurch wird neben den unterschiedlichen Schriftstilen weitere Variabilität in die Schrift gebracht. Dies ist eine Folge aus der Repräsentation einer Trajektorie als Sequenz von Punkten nach der zeitlichen Reihenfolge der Erfassung, wie in Abschnitt 3.1 beschrieben wurde. Delayed Strokes werden mit einer einfachen Heuristik aus der Trajektorie entfernt. Dafür wird die Trajektorie in Segmente geteilt, die jeweils

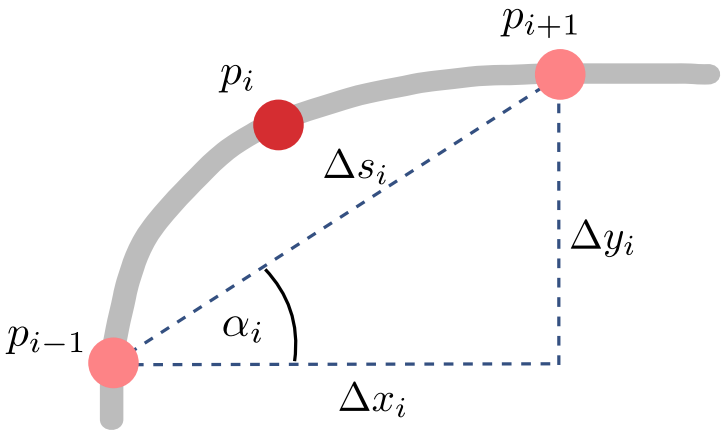


Abbildung 9: Merkmal Schreibrichtung (nach [GAC+91]). Die Schreibrichtung an Punkt p_i ist definiert als die Steigung der Verbindungsgeraden $\overline{p_{i-1}p_{i+1}}$, welche über das Steigungsdreieck bestimmt wird.

mit dem Aufsetzen des Stifts beginnen und mit dem Absetzen enden. Das erste Segment einer Trajektorie ist dabei per Definition kein Delayed Stroke. Die Trajektorie wird nun segmentweise durchlaufen und ein Segment entfernt, wenn es sich räumlich vollständig links vom zeitlich letzten Punkt des vorangegangenen Segments befindet. Eine ähnliche Heuristik kommt in [LB06] zur Anwendung. Die entfernten Segmente werden für die Berechnung des Delayed Strokes Merkmals verwendet (siehe Abschnitt 3.3.3).

3.3 MERKMALE

Im Folgenden werden die Merkmale erläutert, welche zur Informationsextraktion aus der Trajektorie bestimmt werden. Für jeden Punkt der Trajektorie werden dabei alle Merkmale extrahiert und in einem Merkmalsvektor gruppiert. Somit entsteht für jede Trajektorie eine Sequenz von Merkmalsvektoren, die für jeden Punkt numerisch den lokalen Verlauf des Schriftzugs beschreibt. Die Auswahl der implementierten Merkmale stammt aus dem Bereich der Online-Handschrift Erkennung [GAC+91, JMW00, LB06]. Die extrahierten Merkmale umfassen Schreibrichtung und Krümmung der Trajektorie (Abschnitt 3.3.1, Abschnitt 3.3.2), Informationen zur Position des Stiftes (Abschnitt 3.3.3) sowie Nachbarschaftsmerkmale (Abschnitt 3.3.4) und eine Kontext Bitmap (Abschnitt 3.3.5), welche den lokalen Verlauf der Trajektorie in einem Kontext um den untersuchten Punkt analysiert.

3.3.1 *Schreibrichtung*

Dieses Merkmal beschreibt für einen Punkt die aktuelle Bewegungsrichtung des Stiftes [GAC+91, JMW00]. Ermittelt wird die Richtung anhand der beiden Nachbarpunkte $p_{i-1} = (x_{i-1}, y_{i-1})$ und $p_{i+1} = (x_{i+1}, y_{i+1})$ des betrachteten Punktes p_i. Abbildung 9 verdeutlicht die Berechnung des Merkmals und die im Folgenden definierten Variablen. Zunächst werden durch [GAC+91]

$$\Delta x_i = x_{i-1} - x_{i+1}, \tag{8}$$

$$\Delta y_i = y_{i-1} - y_{i+1}, \tag{9}$$

$$\Delta s_i = \sqrt{\Delta x_i^2 + \Delta y_i^2} \tag{10}$$

die Seitenlängen des Steigungsdreiecks der Verbindungsgeraden $\overline{p_{i-1}p_{i+1}}$ bestimmt. Die Schreibrichtung wird dann durch [GAC+91]

$$\cos \alpha_i = \frac{\Delta x}{\Delta s} \tag{11}$$

$$\sin \alpha_i = \frac{\Delta y}{\Delta s} \tag{12}$$

als Sinus und Kosinus des Steigungswinkels α_i zwischen der Geraden und der x-Achse ausgedrückt.

3.3.2 *Krümmung*

Die Krümmung der Trajektorie an einem Punkt wird durch den Sinus und Kosinus des Winkels dargestellt, der durch die Trajektorie in diesem Punkt eingeschlossen wird. Dies ist vergleichbar mit der zweiten Ableitung einer Funktion [GAC+91]. Die Berechnung ist in Abbildung 10 visualisiert. Der Krümmungswinkel β_i ist der Winkel zwischen den zwei Geraden $\overline{p_{i-2}p_i}$ und $\overline{p_ip_{i+2}}$. Die Steigungswinkel dieser Geraden wurden für die Bestimmung der Schreibrichtung der Punkte p_{i-1} und p_{i+1} bereits in Gleichung 11 und Gleichung 12 berechnet und können hier wieder benutzt werden. Der gesuchte Winkel wird durch [GAC+91]

$$\cos \beta_i = \cos \alpha_{i-1} \cdot \cos \alpha_{i+1} + \sin \alpha_{i-1} \cdot \sin \alpha_{i+1} \tag{13}$$

$$\sin \beta_i = \cos \alpha_{i-1} \cdot \sin \alpha_{i+1} - \sin \alpha_{i-1} \cdot \cos \alpha_{i+1}. \tag{14}$$

berechnet.

3.3.3 *Stiftzustand und Delayed Stroke*

Stiftzustand und Delayed Stroke sind zwei binäre Merkmale. Der Stiftzustand beschreibt, ob ein Punkt aktiv durch den Schreiber erzeugt wurde (*pen-down*) oder durch Interpolation, etwa bei der Neuabtastung der Trajektorie, entstanden ist (*pen-up*). Letzteres ist

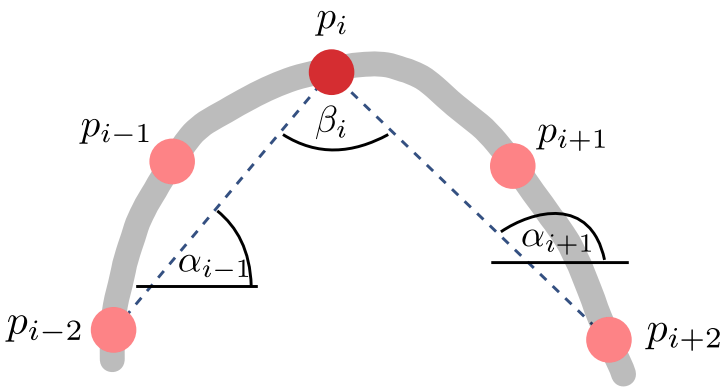


Abbildung 10: Merkmal Krümmung (nach [GAC+91]). Die Krümmung an Punkt p_i ist über den an diesem Punkt eingeschlossenen Winkel β_i bestimmt. Zur Berechnung von β_i werden die Werte der Schreibrichtung für die Punkte p_{i-1} und p_{i+1} wiederverwendet (siehe Abschnitt 3.3.1).

beispielsweise der Fall für den Weg zwischen dem Absetzen des Stiftes und dem Schreiben eines i-Punktes. Das Delayed Stroke-Merkmal zeigt für einen Punkt an, ob dieser sich unterhalb eines solchen Delayed Strokes befindet.

3.3.4 *Nachbarschaftsmerkmale*

Dies ist eine Gruppe von Merkmalen, die für einen Punkt (x_i, y_i) anhand der in seiner Nachbarschaft befindlichen Punkte bestimmt werden. Zur Nachbarschaft eines Punktes p_i gehören, neben p_i selbst, die zwei vorherigen und nachfolgenden Punkte $p_{i-2}, p_{i-1}, p_{i+1}, p_{i+2}$. Als Seitenlängen ΔX_i und ΔY_i der Nachbarschaft von Punkt p_i werden die maximalen Abstände zwischen zwei x- bzw. y-Koordinaten von Punkten in der Nachbarschaft bezeichnet. Dies entspricht den Seitenlängen der Bounding Box der Nachbarschaft und ist nicht zu verwechseln mit der Bounding Box der Trajektorie (siehe Abschnitt 3.1). Die Komponenten der nachfolgend beschriebenen Merkmale sind in Abbildung 11 dargestellt.

Seitenverhältnis

Das Seitenverhältnis A_i der Nachbarschaft wird über die, die Nachbarschaft umgebende, Bounding Box bestimmt und ist definiert als [JMW00]

$$A_i = \frac{2 \cdot \Delta Y_i}{\Delta X_i + \Delta Y_i} - 1. \tag{15}$$

Dies entspricht dem Verhältnis der Seitenlängen der Bounding Box und beschreibt deren Ausdehnung. Je größer das Seitenverhältnis

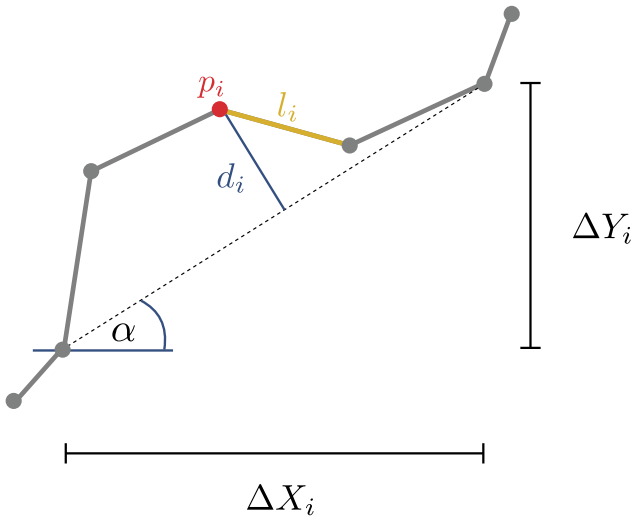


Abbildung 11: Schematische Darstellung der Berechnungskomponenten der Nachbarschaftsmerkmale für die Nachbarschaft von Punkt p_i. l_i bescheibt die Länge der Verbindungsgeraden von p_i und p_{i+1}. d_i ist der Abstand des Punktes p_i von der Verbindungsgeraden zwischen erstem und letztem Punkt der Nachbarschaft. ΔX_i und ΔY_i entsprechen den Seitenlängen der die Nachbarschaft umgebende Bounding Box.

ist, desto schmaler ist die Bounding Box und somit die enthaltene Trajektorie.

Curliness

Das Merkmal „Curliness“ (*engl.* to curl - kringeln) beschreibt die Abweichung C_i der Trajektorie von einer geraden Linie. Es wird durch [JMWoo]

$$C_i = \frac{\sum_{j=0}^{N-2} l_j}{\max(\Delta X_i, \Delta Y_i)} - 2 \tag{16}$$

berechnet, wobei N die Anzahl der Punkte in der Nachbarschaft und l_j der Abstand von Punkt p_j zu p_{j+1} ist. Die Curliness entspricht somit dem Verhältnis der Länge der Trajektorie zu einer geraden Linie mit selbem Start- und Endpunkt wie die Trajektorie.

Lineness

Die „Lineness“ (etwa: Geradlinigkeit) beschreibt den mittleren quadratischen Abstand L_i aller Punkte der Nachbarschaft von der Geraden, die den ersten und letzten Punkt der Nachbarschaft verbindet. Sie wird durch [JMWoo]

$$L_i = \frac{1}{N} \cdot \sum_{j=0}^{N-1} d_j^2, \tag{17}$$

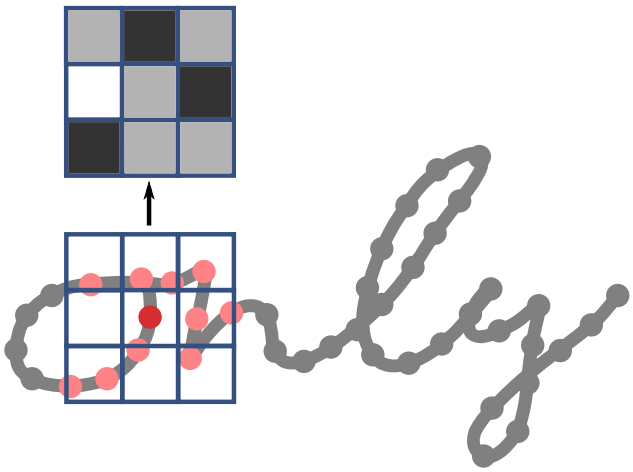


Abbildung 12: Bestimmung der Kontext Bitmap für den rot gekennzeichneten Punkt. Dafür wird erfasst, wieviele Punkte der Trajektorie in jede Region eines Rasters um den aktuellen Punkt fallen. Jeder Region entspricht ein Pixel der Kontext Bitmap, dessen Grauwert von der Anzahl der Punkte in dieser Region abhängt.

berechnet, wobei N die Anzahl der Punkte in der Nachbarschaft ist und d_j den Abstand von Punkt p_j zur Geraden beschreibt. Durch die Lineness wird der „Umweg“ erfasst, welcher durch die Trajektorie auf dem Weg von Start- zu Endpunkt gemacht wird, im Gegensatz zu einer direkten Verbindung.

Steigung

Die Steigung des in der Nachbarschaft befindlichen Trajektoriensegments wird durch den Kosinus des Winkels α (siehe Abbildung 11) beschrieben, der zwischen der Geraden vom ersten zum letzten Punkt und der x-Achse eingeschlossen wird [JMW00].

3.3.5 *Kontext Bitmap*

Eine Kontext Bitmap ist ein Graustufen-Bild, welches die Umgebung des aktuellen Punktes beschreibt [MFW94]. Dabei wird der erfasste Bereich der Trajektorie in gleichgroße Regionen geteilt, die in der Bitmap jeweils durch ein Pixel dargestellt werden. Für jedes Pixel wird nun gezählt, wieviele Punkte der Trajektorie in die entsprechende Region fallen. Dies bestimmt den Grauwert dieses Pixels. Dieses Verhalten ist schematisch in Abbildung 12 dargestellt. Eine Kontext Bitmap beschreibt somit für jeden Punkt die Dichte der Punkte in seiner Umgebung und die Richtungen in der diese Punkte liegen. Sie erfasst dadurch Kontextinformationen, die, abhängig von der Größe der Regionen, mehrere Buchstaben umfasst. Bei der Größe

der Bitmap wurde sich hier an den Vorgaben aus der Literatur zur Online-Handschrift Erkennung orientiert, in der die Bitmap 3x3 Pixel misst [JMW00, LB06]. Zum Abschluss wird jedes Pixel der Bitmap als Merkmal aufgefasst und sein Wert dem Merkmalsvektor hinzugefügt.

3.4 ZUSAMMENFASSUNG

In diesem Kapitel wurden alle benötigten Methoden beschrieben, um eine Online-Handschrift Trajektorie, dargestellt durch eine Sequenz von Punkt-Koordinaten, durch eine Merkmalsvektorsequenz zu repräsentieren. Dazu wird die Trajektorie zunächst normalisiert, um Variabilität in der Schrift zu vermindern, die durch einen oder mehrere Schreiber erzeugt wird. Anschliessend werden für jeden Punkt der Trajektorie insgesamt neun Merkmale (bzw. 19 Werte) extrahiert, welche lokale Eigenschaften des Schriftverlaufs der Trajektorie erfassen. Sowohl die Normalisierungsschritte, als auch die Merkmale wurde aus dem Bereich der Online-Handschrift-Erkennung zusammengestellt. In Kapitel 6 wird beschrieben, wie die Merkmalsvektorsequenzen von Online-Handschrift Trajektorien dazu benutzt werden, um die neue *Bag-of-Online-Trajectory* Merkmalsrepräsentation zu berechnen.

4 REPRÄSENTATION VON WORTBILDERN

In der Einleitung wurde eine kurze Übersicht über die übliche Herangehensweise zum Aufbau eines Word Spotting Systems gegeben (siehe Abschnitt 1.2). Beim segmentierungsbasierten Word Spotting werden Wortbilder in einem Vorverarbeitungsschritt aus Dokumenten extrahiert. Für diese Wortbilder werden in der Regel anschliessend Merkmalsrepräsentationen berechnet, welche deren unterscheidenden Eigenschaften erfassen und hervorheben. In dieser Arbeit wird der bekannte Bag-of-Features Ansatz (BoF) für die Bestimmung der Merkmalsrepräsentation verwendet. Dieser wird in Abschnitt 4.1 zunächst konzeptionell mit anderen Merkmalsrepräsentationen verglichen und anschliessend in Abschnitt 4.2 erläutert. Die Bag-of-Features Repräsentationen enthält keine Informationen über die räumliche Verteilung von Merkmalen in Wortbildern. Da diese jedoch wichtig für die Beschreibungsleistung der Merkmalsrepräsentation sind, werden sie ihr durch eine Spatial Pyramid nachträglich hinzugefügt. Dies ist in Abschnitt 4.3 beschrieben.

4.1 ÜBERSICHT

In diesem Abschnitt wird ein Überblick über einige visuelle Merkmalsrepräsentationen gegeben, welche im segmentierungsbasierten Word Spotting verwendeten werden. Die Merkmalsrepräsentationen werden dabei für Wortbilder berechnet, die Methoden lassen sich jedoch auch für die Berechnung von Repräsentationen von Patches einsetzen, welche häufig beim segmentierungsfreien Word Spotting ermittelt werden (vgl. Kapitel 1). Für ein Wortbild beschreibt eine Merkmalsrepräsentation die Eigenschaften, anhand derer der dargestellte Schriftzug des repräsentierten Wortes von Schriftzügen anderer Wortbilder unterschieden werden kann. Die dabei berechneten Merkmale erfassen visuelle Ausprägungen des dargestellten Schriftzuges. Beispiele für die Verwendung der beschriebenen Merkmalsrepräsentationen sind in Kapitel 5 zu finden.

Eine Möglichkeit zur Repräsentation eines Wortbildes besteht in der Berechnung einer Sequenz von Merkmalsvektoren. In [RM07] geschieht dies anhand von sogenannten Projektionsprofilen. Dabei werden für jede Pixelspalte eines Wortbildes beispielsweise Merkmale aus den Übergängen von dunklen zu hellen Pixeln ermittelt, welche den Übergang des dargestellten Schriftzuges zum Hintergrund des Wortbildes beschreiben. Für jede Pixelspalte entsteht dadurch ein Vektor aus mehreren Merkmalen und für das Wortbild

dementsprechend eine Sequenz von Merkmalen. In [RP08] wird ein ähnliches Verfahren beschrieben. Dabei wird der LGH-Deskriptor (Local Histogram of Gradients) vorgestellt, welcher anhand der Bildintensitäten des Wortbildes Gradienten berechnet. Wie zuvor beschrieben, wird ein Fenster über das Wortbild geschoben. An jeder Position des Fensters wird ein Histogramm über die Orientierungen der sich im Fenster befindlichen Gradienten gebildet. Auch durch dieses Vorgehen entsteht eine Sequenz von Merkmalsvektoren.

Eine Alternative zur Erstellung einer Sequenz von Merkmalsvektoren stellt die holistische Repräsentation eines Wortbildes durch einen einzelnen Merkmalsvektor dar. Eine prominente Merkmalsrepräsentation hierfür ist der Bag-of-Features Ansatz (siehe [OD11, ARTL13]), welcher genauer im nachfolgenden Abschnitt 4.2 erläutert wird. Dabei werden lokale Bilddeskriptoren verwendet, um einzelne Ausschnitte des Wortbildes zu beschreiben. Ein visuelles Vokabular wird durch Clustern von Deskriptoren berechnet, welches typische numerische Ausprägungen der Deskriptoren beschreibt. Die Merkmalsrepräsentation für ein Wortbild besteht aus einem Histogramm, das die Häufigkeit von Deskriptoren, welche anhand des visuellen Vokabulars quantisiert werden, in diesem Wortbild angibt. Eine Erweiterung des Bag-of-Features Ansatz stellen Fisher Vektoren dar (siehe [PSM10, AGFV14a]). Hierbei werden die lokale Deskriptoren nicht, wie oben beschrieben, quantisiert, sondern durch Verteilungsdichten einer globalen Mischverteilung beschrieben.

Beide beschriebenen Vorgehen zur Bestimmung einer Merkmalsrepräsentation bringen Vor- und Nachteile zur Repräsentation von Wortbildern mit sich. Eine Sequenz-basierte Repräsentation enthält durch die Reihenfolge der Merkmalsvektoren Informationen darüber, wo in einem Wortbild welche Merkmalsausprägungen beobachtet wurden. Solche Informationen sind einer holistischen Merkmalsrepräsentation durch einen einzelnen Merkmalsvektor nicht zu entnehmen. Dafür können für den Vergleich von zwei Merkmalsvektoren übliche Abstandsmaße für Vektoren, wie z.B. die euklidische Distanz oder die Kosinusdistanz, verwendet werden. Der Vergleich von zwei Merkmalsvektorsequenzen ist hingegen nicht trivial. Aufgrund der Einfachheit der Darstellung und der Struktur der evaluierten Word Spotting Verfahren, wird in dieser Arbeit der Bag-of-Features Ansatz als Merkmalsrepräsentation für Wortbilder verwendet.

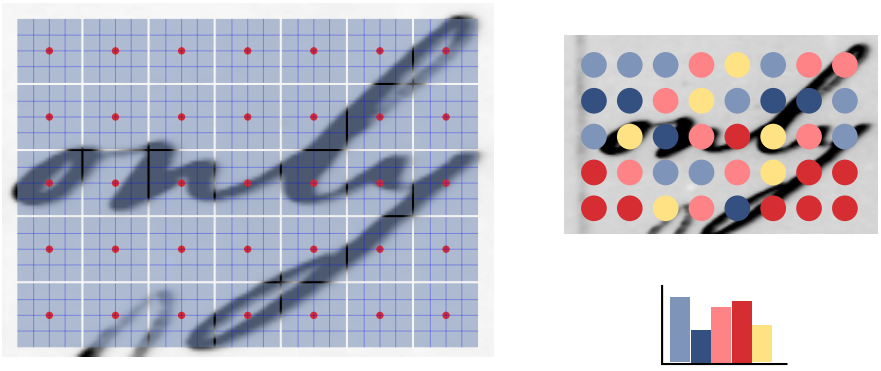
4.2 Bag-of-Features Merkmalsrepräsentation

Die Idee des Bag-of-Features Ansatzes stammt aus einer ähnlichen Vorgehensweise im Bereich der Dokumentenanalyse. Dort beschreibt der Bag-of-Words Ansatz einen Text durch die Häufigkeit der vorkommenden Wörter eines zuvor bestimmten Vokabulars [OD11]. Das Vokabular besteht beispielsweise aus der von Stoppwörtern befreiten

Menge aller Wörter der zur Verfügung stehenden Dokumente (vgl. [BYRN99], S. 62). Stoppwörter, wie z.B. „the“ und „so“, tragen keine relevanten Informationen zum Inhalt eines Dokuments. Anhand der Worthäufigkeiten wird für jedes Dokument ein *Termvektor* erstellt, in dem jede Dimension die Anzahl des Vorkommenes eines Wortes (Terms) des Vokabulars enthält. Häufig wird der Termvektor normalisiert, indem jedes Element durch die Anzahl aller Wortvorkommen des Dokuments geteilt wird. Dadurch ist die Repräsentation unabhängig von der Anzahl der Wörter in einem Dokument. Die Benennung als „Bag“ (*dt.* Beutel, Sack) entstammt der Tatsache, dass die Terme ungeordnet im Termvektor vorliegen und aus ihnen nicht mehr auf die räumliche Struktur eines Dokuments geschlossen werden kann.

Der Bag-of-Words Ansatz wurde für die Arbeit in Bild-Retrieval Aufgaben als Bag-of-Features (BoF) adaptiert [OD11]. Die Berechnung einer Bag-of-Features Merkmalsrepräsentation geschieht anhand eines visuellen Vokabulars, welches durch Clustering (vgl. Abschnitt 2.1) von Bildmerkmalen erstellt wird, die aus einer Menge von Beispielbildern extrahiert werden. Die Clusterzentren entsprechen den Termen des Termvektors beim Bag-of-Words Ansatz und werden Visual Words genannt. Das Clustering ist notwendig, da die Bildmerkmale im Gegensatz zu Termen numerisch sind und somit keine diskreten, zählbaren Einheiten darstellen. Für ein ungesehenes Bild werden zuerst Bildmerkmale extrahiert und anhand der Visual Words quantisiert. Der Termvektor für dieses Bild wird dann erstellt, indem die Anzahl der Vorkommen der einzelnen quantisierten Bildmerkmale gezählt wird. Auch hier kann eine Normalisierung anhand der Anzahl aller Bildmerkmale des Bildes erfolgen. So wie der Termvektor, ist der Visual Words ungeordnet und lässt somit keinen Schluss auf die räumliche Verteilung der Bildmerkmale im Bild zu.

Im Bereich des segmentierungsbasierten Word Spotting wird der Bag-of-Features Ansatz zur Repräsentation von Wortbildern verwendet [ARTL13]. Dabei ist es üblich, lokale Bilddeskriptoren als Bildmerkmale zu nutzen. Diese Bilddeskriptoren werden an bestimmten Positionen des Wortbildes berechnet und erfassen den lokalen Schriftverlauf des Wortes, welches durch das Wortbild dargestellt ist. In dieser Arbeit wird zu diesem Zweck der SIFT-Deskriptor verwendet, welcher ausführlicher in Abschnitt 4.2.1 beschrieben wird. Es sei an dieser Stelle angemerkt, dass der BoF-Ansatz mit beliebigen lokalen Deskriptoren arbeiten kann. SIFT-Deskriptoren werden in einem dichten, regelmäßigen Raster aus einem Wortbild extrahiert. Dies ist in Abbildung 4.13(a) dargestellt. Dabei wird für jeden rot gekennzeichneten Punkt ein SIFT-Deskriptor extrahiert, dessen Größe durch den blauen Bereich angegeben ist. Durch die Größe des Deskriptors wird der Kontext eingestellt, welcher erfasst wird. Bei kleinen Größen werden nur Teile von Buchstaben erfasst, während größere Deskriptoren einen Kontext von mehreren Buchstaben einschliessen. Die tat-



(a) (b)

Abbildung 13: Die Berechnung einer Bag-of-Features Merkmalsrepräsentation anhand von SIFT-Deskriptoren auf einem Wortbild (nach [FR14]). (a) SIFT-Deskriptoren werden in einem dichten, regelmäßigen Raster aus dem Wortbild extrahiert. Für jeden Punkt (rot) wird die lokale Umgebung (blau) in Zellen eingeteilt, und eine Repräsentation aus Gradienteninformationen berechnet. (b) Für das Wortbild aus (a) wurden die berechneten SIFT-Deskriptoren anhand des visuellen Vokabulars quantisiert. Jeder quantisierte Deskriptor wird durch einen farbigen Punkt gekennzeichnet. Die Häufigkeit der quantisierten Deskriptoren wird in einem Histogramm erfasst.

sächlich verwendete Größe ist dabei vom Anwendungsfall abhängig. Die Berechnung des visuellen Vokabulars erfolgt anhand von SIFT-Deskriptoren, welche aus einer Menge von Beispielwortbildern extrahiert wurden. Dies wird in Abschnitt 4.2.2 erläutert. Mit dem vorliegenden visuellen Vokabular kann für ein neues Wortbild nun eine BoF-Merkmalsrepräsentation bestimmt werden. Dafür werden SIFT-Deskriptoren in einem dichten, regelmäßigem Grid auf diesem Wortbild berechnet. Die Deskriptoren werden anhand des visuellen Vokabular quantisiert. Zum Abschluss wird ein Histogramm über die Häufigkeiten der im Wortbild vorkommenden Visual Words erstellt. Die Berechnung der Merkmalsrepräsentation ist in Abbildung 4.13(b) dargestellt. Unterschiedlich farbig gekennzeichnete Punkte stellen hierbei die quantisieren Deskriptoren dar, deren Häufigkeitsverteilung in einem Histogramm erfasst wird.

4.2.1 *SIFT Deskriptoren*

Der SIFT-Deskriptor (scale-invariant feature transform) ist ein lokaler, gradientenbasierter Bilddeskriptor, der der Extraktion von

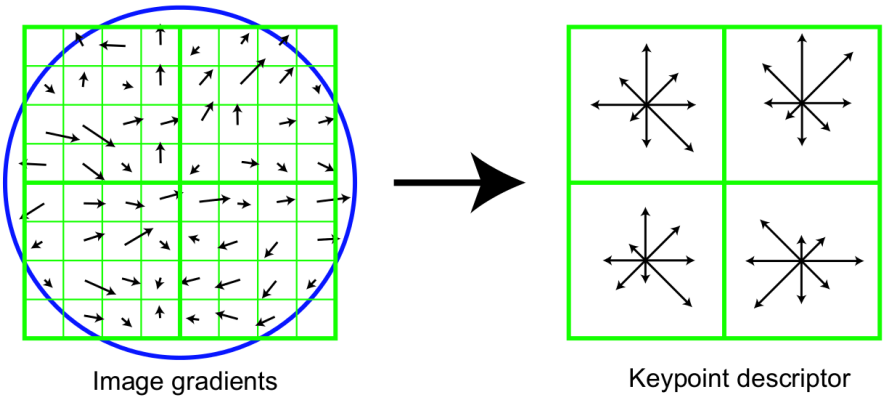


Abbildung 14: Aufbau des SIFT-Deskriptors (aus [Low04]). Für einen Bildpunkt wird die lokale Umgebung in 2x2 Zellen eingeteilt. In jeder Zelle werden anhand der Bildintensitäten Gradienteorientierungen und Gradientenmagnituden bestimmt (links). Alle Gradientenmagnituden werden über eine Gaussverteilung, dargestellt durch einen blauen Kreis, gewichtet. Die Gradientenorientierungen werden in jeder Zelle in einem Histogramm zusammengetragen (rechts). Dabei werden die Orientierungen in 8 Richtungen quantisiert und der jeweilige Beitrag zum Histogrammeintrag anhand der Gradientenmagnituden bestimmt. Die numerische Repräsentation des Deskriptors entsteht durch Konkatenation der Histogramme.

Informationen aus einem Bildbereich um einen Punkt dient [Low99, Low04]. Er bildet zusammen mit dem SIFT-Detektor ein Framework zur Berechnung von Bildmerkmalen. Durch den Detektor werden dabei „interessante“ Punkte in einem Graustufenbild ermittelt und deren lokale Nachbarschaft über den Deskriptor durch Merkmalsvektoren repräsentiert. Interessant sind solche Punkte, in deren lokaler Umgebung Kanten, d.h. beispielsweise Übergange hoher Bildintensität zu niedriger Bildintensität zu finden sind, die die Struktur des Bildes charakterisieren. SIFT ist ein beliebtes Framework im Bereich der Objekterkennung in Bildern [Low99] und des Bild-Retrieval [SZ03].

Im segmentierungsbasierten Word Spotting (vgl. Kapitel 1) werden SIFT-Deskriptoren für die Bestimmung von Merkmalsrepräsentation von Wortbildern eingesetzt. Dabei ist es üblich den SIFT-Deskriptor ohne den SIFT-Detektor zu verwenden und die Punkte, an denen SIFT-Deskriptoren berechnet werden, in einem dichten, regelmäßigen Raster anzuordnen (vgl. [RATL11, LRF+12, ARTL13, AGFV14a]), wie es in Abbildung 4.13(a) dargestellt ist. Der Abstand der Punkt voneinander ist ein manuell zu wählender Parameter. Ein SIFT-Deskriptor

in der Standardkonfiguration (vgl. [Low04]) wird wie folgt für einen Punkt im Wortbild bestimmt. Um den Punkt als Zentrum wird ein Grid aus 4x4 gleichförmigen quadratischen Zellen aufgebaut, deren Größe in Pixel ebenfalls ein zu wählender Parameter für die Anwendung des Deskriptors ist. In jeder dieser Zellen werden anhand der Bildintensitäten, d.h. den Graustufenwerten der Pixel, Gradientenorientierungen und -magnituden bestimmt. Durch die Gradienten werden Kanten in der lokalen Nachbarschaft um den betrachteten Punkte erfasst. In Wortbildern entspricht dies den Kanten des Schriftverlaufs. Dadurch werden durch jeden Deskriptor lokale Ausprägungen des Schriftverlaufes repräsentiert. Die berechneten Gradientenmagnituden werden durch eine Gaussverteilung gewichtet, welche zentral über dem Bildpunkt, für den der Deskriptor berechnet wird, platziert wird. Die Gradientenorientierungen werden für jede Zelle in ein Histogramm aus acht Hauptrichtungen quantisiert. Dabei wird der Beitrag jeder Gradientenorientierung zur quantisierten Hauptrichtung durch die Gradientenmagnitude bestimmt. Durch Konkatenation dieser insgesamt 16 Histogramme entsteht somit ein Deskriptor mit $16 \cdot 8 = 128$ Dimensionen. Die Berechnung eines SIFT-Deskriptors ist in Abbildung 14 dargestellt. Dabei werden hier, im Gegensatz zur oben beschriebenen Konfiguration, nur 2x2 Zellen erstellt (in der rechten Bildhälfte zu sehen), aus denen vier Histogramme gebildet werden.

4.2.2 *Visuelles Vokabular*

Der Bag-of-Features Ansatz arbeitet mit einem sogenannten visuellen Vokabular, welches durch Clustering aus einer Menge von Bildmerkmalen bestimmt wird. Da die Bildmerkmale in der Regel numerisch sind, werden Cluster aus ähnlichen Ausprägungen der Merkmale gebildet. Das Clusterzentrum stellt eine Abstraktion der sich im zugehörigen Cluster befindlichen Merkmalsausprägungen dar. Bei der Anwendung des Bag-of-Features Ansatz für das Word Spotting werden in dieser Arbeit SIFT-Deskriptoren als Bildmerkmale verwendet. Wie in Abschnitt 4.2.1 erläutert, erfasst der SIFT-Deskriptor lokale Gradientenrichtungen eines Bildes in einem Bereich um den Punkt, an dem der Deskriptor berechnet wird. Auf Wortbilder bezogen bedeutet dies, dass die Kanten des dargestellten Schriftzuges erfasst werden. Beim Clustern von SIFT-Deskriptoren aus Wortbildern werden SIFT-Deskriptoren von visuell ähnlichen Kanten- bzw. Schriftverläufen in einem Cluster zusammengetragen. Das Clusterzentrum stellt somit eine typische Ausprägung eines visuellen Schriftverlaufs dar.

Der Begriff des visuellen Vokabulars entstammt, wie in Abschnitt 4.2 bereits erwähnt, aus dem Bereich der Dokumentenanalyse, in dem ein Vokabular, d.h. die Wörter eines Textes, Informationen zum Inhalt eines Textes preisgibt. Wenn zwei Texte sich spezielle, fachbezogene

Wörter teilen, ist dies ein Hinweis darauf, dass die Texte ein gemeinsames Thema behandeln. Analog enthalten Wortbilder, welche eine ähnliche Teilmenge des visuellen Vokabulars gemeinsam verwenden, ähnliche visuelle Schriftverläufe. Die Einheiten des visuellen Vokabulars werden dementsprechend auch als *Visual Words* bezeichnet.

Das visuelle Vokabular wird mithilfe einer Menge von SIFT-Deskriptoren unüberwacht gelernt. Dazu werden aus einer Menge von Beispiel-Wortbildern SIFT-Deskriptoren in einem dichten, regelmäßigen Raster berechnet, wie in Abschnitt 4.2.1 dargelegt. Eine ausreichend große Teilmenge aller Deskriptoren wird anschliessend mittels des Algorithmus von Lloyd geclustert (siehe dazu Abschnitt 2.1). Das entstehende Kodebuch aus Clusterzentren entspricht dem visuellen Vokabular aus Visual Words.

Ein neues Wortbild kann nun anhand des visuellen Vokabulars beschrieben werden, indem das Vorkommen von Visual Words in ihm ermittelt wird. Dazu werden auf diesem Wortbild ebenfalls SIFT-Deskriptoren in einem dichten Raster berechnet. Die Deskriptoren werden anschliessend anhand der Visual Words quantisiert. Dies geschieht beispielsweise durch das Ersetzen jedes Deskriptors mit dem räumlich nächsten Visual Word anhand der euklidische Distanz. Eine Alternative dazu bieten weiche Quantisierungsverfahren, welche einen Deskriptor anteilig durch die k nächsten Visual Words quantisieren. Der Anteil, den ein Visual Word an der Quantisierung eines Deskriptors hat, wird dabei über den Abstand des Deskriptors zu diesem Visual Word ermittelt. Ein solches weiches Quantisierungsverfahren ist beispielsweise Locality-constrained Linear Coding (LLC), welches einen Deskriptor anhand einer Linearkombinationen aus den k nächsten Visual Words quantisiert [WYY+10]. Diese Quantisierungsmethode wird in der Arbeit von Aldavert et al. [ARTL13] verwendet, welche in Abschnitt 5.3.3 genauer beschrieben wird.

4.3 Lokalitätsinformationen über Spatial Pyramid

Eine Bag-of-Features Merkmalsrepräsentation enthält über die in Visual Words geclusterten lokalen Deskriptoren Informationen zum Aufbau des dargestellten Bildes (vgl. Abschnitt 4.2.2). Ein BoF Termvektor besteht dabei aus einem Histogramm über die im Bild vorkommenden Visual Words. Diese Darstellung ist ungeordnet, lässt also keinen Schluss auf die räumliche Anordnung der beobachteten Visual Words zu. Eine Möglichkeit, der BoF-Repräsentation Lokalitätsinformationen hinzuzufügen, besteht in der Anwendung einer Spatial Pyramid, welche zunächst für die Klassifikation von Bildern vorgestellt wurde [LSP06]. Dabei wird das repräsentierte Bild in Regionen aufgeteilt, in denen jeweils separat ein Histogramm der vorkommenden Visual Words erstellt wird, wie im Folgenden erläutert wird.

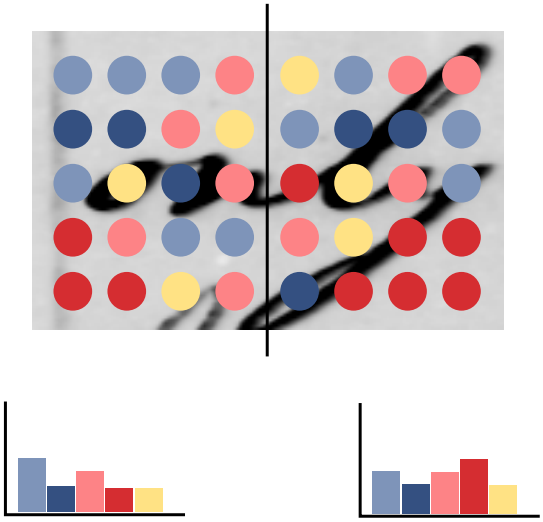


Abbildung 15: Anwendungsbeispiel einer Spatial Pyramid (nach [FR14]). Dargestellt ist ein Wortbild, in dem durch die farbigen Punkte die quantisierten SIFT-Deskriptoren (Visual Words) gekennzeichnet sind. Das Wortbild wird in zwei Regionen geteilt. In dieser „2x1“ Auflösung wird für jede Region ein Histogramm über die Vorkommenden Visual Words gebildet. Die BoF+SP-Merkmalsrepräsentation entsteht durch Konkatenieren und Normalisieren der Histogramme.

Auch im Bereich des Word Spotting wurde gezeigt, dass die Erweiterung des BoF-Ansatzes mit Lokalitätsinformationen einen Zuwachs der Erkennungsraten zur Folge hat (z.B. [ARTL13, AGFV14a]). Auf einem Wortbild berechnete Visual Words geben Informationen zu wiederholt auftretenden lokalen Schriftverläufen des dargestellten Wortes. Durch die Spatial Pyramid kann hier somit ein Hinweis darauf erlangt werden, in welchem Teil des Wortes (bzw. des Wortbildes) welche Schriftverläufe beobachtet wurden. Zur Anwendung der Spatial Pyramid für ein Wortbild werden zunächst, wie zuvor beschrieben, SIFT-Deskriptoren in einem dichten, regelmäßigen Raster extrahiert und anhand der Visual Words quantisiert. Anschliessend wird jedoch nicht ein Histogramm über die quantisierten Deskriptoren des gesamten Wortbildes erstellt, sondern dieses in Teilbereiche aufgeteilt, in denen jeweils ein lokales Histogramm über die quantisierten Deskriptoren gebildet wird. Die Aufteilung und Anzahl der Teilbereiche ist frei wählbar und abhängig von der jeweiligen Anwendung.

Für die Histogramme erfolgen verschiedene Normalisierungsschritte. In dieser Arbeit orientieren sich diese am Vorgehen in [ARTL13]. Die Histogramme der einzelnen Teilbereiche werden dabei separat anhand der Größe des jeweiligen Histogramms L2-normalisiert, da-

mit Teilbereiche mit höherer Anzahl von Deskriptoren nicht diejenigen, mit niedriger Anzahl an Deskriptoren, dominieren. Anschliessend werden alle normalisierten Histogramme zu einem Vektor konkateniert. Dieser Vektor wird zuerst Power-normalisiert (siehe dazu [PSM10]) und anschliessend L2-normalisiert und ergibt so die finale visuelle Repräsentation des Wortbildes. Diese wird im Folgenden auch mit „BoF+SP“ abgekürzt. Die verwendete Notation für die Spatial Pyramid Konfiguration beinhaltet die Anzahl der Regionen pro Level und wird in den folgenden Kapiteln auch als Auflösung bezeichnet (*engl.* resolution [LSP06]). Ein Level beschreibt dabei die Anwendung einer Auflösung – ein Wortbild kann in mehrere Level unterteilt werden, in denen unterschiedliche Auflösungen zum Einsatz kommen. In Abbildung 15 ist eine Spatial Pyramid mit einer „2x1“ Auflösung abgebildet (Ein Level, zwei Regionen in horizontaler Richtung, eine Region in vertikaler Richtung). Die Bezeichnung „2x1/1x1“ erweitert diese Konfiguration beispielsweise um ein zweites Level, in welchem es nur eine Region gibt. In [ARTL13] wird angemerkt, dass für Wortbilder eine horizontale Unterteilung einen größeren Effekt auf die Beschreibungsleistung der Merkmalsrepräsentation hat, als eine vertikale Unterteilung. Die in den folgenden Kapiteln verwendeten Spatial Pyramid Auflösungen folgen dieser Beobachtung.

4.4 ZUSAMMENFASSUNG

In diesem Kapitel wurde der Bag-of-Features Ansatz beschrieben, welcher im Rahmen dieser Arbeit dazu verwendet wird, eine Merkmalsrepräsentation für ein Wortbild zu bestimmen. Zur Bildung der BoF Repräsentation werden SIFT-Deskriptoren verwendet, lokale Bilddeskriptoren, die im Kontext von Wortbildern lokale Schriftverläufe erfassen. Diese Deskriptoren werden anhand eines visuellen Vokabulars quantisiert und ihre Häufigkeit in Histogrammen erfasst. Nach dem Hinzufügen von Lokalitätsinformationen durch eine Spatial Pyramid besteht die finale Merkmalsrepräsentation für ein Wortbild somit aus konkatenierten Histogrammen, welche für jeden Teilbereich des Wortbildes die Häufigkeitsverteilung der quantisierten SIFT-Deskriptoren angeben. Der Bag-of-Features Ansatz wurde zudem mit anderen Merkmalsrepräsentationen aus dem Bereich des Word Spotting verglichen, welche in verschiedenen Verfahren aus der Literatur zum Einsatz kommen (siehe Kapitel 5).

5 WORD SPOTTING VERFAHREN

Das folgende Kapitel stellt eine Auswahl verschiedener Word Spotting Verfahren vor. Zunächst wird in Abschnitt 5.1 die Idee des Word Spotting beschrieben und näher auf das Unterscheidungsmerkmal der zwei typischen Anfragearten Query-by-Example und Query-by-String, welche bereits in der Einleitung beschrieben wurden, eingegangen. Anschließend werden in Abschnitt 5.2 und Abschnitt 5.3 Word Spotting Verfahren aus der Literatur für Query-by-Example bzw. Query-by-String beschrieben. In Abschnitt 5.4 werden zwei dieser Verfahren im Hinblick auf ihre Vorgehensweise verglichen und ihre Relevanz für das Word Spotting mit Online-Handschrift Anfragen beschrieben.

5.1 ÜBERBLICK

Word Spotting wurde 1996 von Manmatha et al. [MHR96] zum ersten Mal vorgestellt, um mit Texten arbeiten zu können, für die eine Transkription nicht oder nur sehr fehlerbehaftet möglich ist. Dabei handelte es sich zunächst um eine Methode, welche jedes Wort aus dem Dokument segmentiert, binarisiert und schließlich anhand des paarweisen XORs dieser Wortbilder ein Clustering vornimmt. Dieses Clustering hat zum Ziel, möglichst viele Wortbilder, die das selbe Wort darstellen, in einem Cluster zusammenzufassen. Durch eine Indizierung der Position der einzelnen Wortbilder im Dokument konnte anschließend durch eine manuelle Annotation „interessanter“ Cluster eine teilweise Transkribierung des Dokuments und eine Erstellung eines Stichwortverzeichnisses erreicht werden. In [KAA+00, RM03, RM07] wurde dieses Verfahren verbessert, indem für jedes segmentierte Wortbild eine Merkmalsvektorsequenz extrahiert wird. Als Merkmale dienen Projektionsprofile, welche die Form des Wortes in einem Wortbild anhand der Übergänge zwischen Schriftzug und Hintergrund erfassen, die für jede Pixelspalte bestimmt werden. Für zwei Wortbilder werden diese Sequenzen mit Dynamic Time Warping verglichen, eine Methode zur Abstandsberechnung zwischen zwei Zeitreihen [RM03].

Anstelle des Aufbaus eines Wortindex wurden in späteren Arbeiten vermehrt Modelle konstruiert, welche anhand einer Anfrage ein Retrieval auf dem zu durchsuchenden Dokument durchführen. Beim Query-by-Example Prinzip besteht die Anfrage aus einem Beispiel eines Wortbildes, welches beispielsweise von einem Benutzer mittels grafischer Oberfläche aus dem Dokument selektiert wird, nach dem

das Dokument durchsucht wird [RP08, AGFV14b]. Aktuelle Query-by-Example Methoden, wie z. B. [RATL11, RATL15], verwenden zudem keinen separaten Vorverarbeitungsschritt zum Segmentieren aller Wörter, sondern bilden durch ein Fenster, welches über die Dokumentseiten geschoben wird, sogenannte Patches. Diese Patches werden jeweils über ihre Merkmalsrepräsentation mit der Merkmalsrepräsentation der Anfrage verglichen. Solche Word Spotting Verfahren werden als segmentierungsfrei bezeichnet. Beim Query-by-String Word Spotting hingegen liegt die Anfrage als Zeichenkette vor [ARTL13, AGFV14a]. Dies hat den Vorteil, dass nicht zunächst manuell eine Instanz des zu suchenden Wortes im Dokument gesucht und markiert werden muss. Die Schwierigkeit beim Query-by-String besteht in der domänenübergreifenden Suche von den textuellen Informationen der Zeichenkette in den visuellen Informationen der Wortbilder.

5.2 QUERY-BY-EXAMPLE

Query-by-Example Word Spotting Systeme durchsuchen eine Menge von Dokumentbildern nach Vorkommen von dargestellten Wörtern, welche einer Anfrage, gegeben durch ein Wortbild, ähnlich sind QbE ist eine schwierige Aufgabe, da Bilder des gleichen Wortes aufgrund von unterschiedlichen Schriftstilen, Schreibfehlern und sonstigem Rauschen sehr unterschiedlich aussehen können [RSP12b]. Hinzu kommen Segmentierungsfehler bei Wortbildern von segmentierungsbasierten Verfahren. Die typische Vorgehensweise von QbE Verfahren ist das Berechnen von Merkmalsrepräsentationen für das Query-Bild und alle Wortbilder der zu durchsuchenden Dokumente. Über ein definiertes Abstandsmaß wird anschließend durch den Abstand der Query zu jedem Kandidaten ein Score berechnet, anhand dessen die Treffer sortiert werden. In Abschnitt 5.2.1 wird ein Verfahren beschrieben, welches die, in Kapitel 4 vorgestellte, Merkmalsrepräsentation für ein segmentierungsfreies Word Spotting Verfahren einsetzt. Abschnitt 5.2.2 stellt ein Verfahren vor, dass Wortbilder anhand von Hidden Markov Modellen vergleicht. Das QbE Verfahren in Abschnitt 5.2.3 setzt schließlich Support Vektor Maschinen ein, um relevante von nicht relevanten Wortbildern für eine Anfrage zu trennen.

5.2.1 *Bag-of-Features mit SIFT-Deskriptoren*

In [RATL11] werden für ein segmentierungsfreies Query-by-String Verfahren Merkmalsrepräsentationen für Patches anhand von SIFT-Deskriptoren ermittelt. Dafür wird anhand einer Dokumentseite ein visuelles Vokabular mit 1500 Wörtern berechnet, wie in Abschnitt 4.2.2 näher erläutert. Für eine gegebene Anfrage wird als visuelle Merkmalsrepräsentation ein Bag-of-Features-Vektor (vgl.

Abschnitt 4.2) bestimmt. Die Dokumentseiten werden in Patches aufgeteilt, indem ein Fenster fester Größe über sie geschoben wird. Für jeden Patch wird ebenfalls ein BoF-Vektor berechnet. Alle BoF-Vektoren werden zudem mit Lokalitätsinformationen über eine Spatial Pyramid (vgl. Abschnitt 4.3) erweitert.

Mit allen visuellen Merkmalsrepräsentationen einer Dokumentseite wird für diese Seite über die Latent Semantic Analysis ein Topic-Raum bestimmt (vgl. Abschnitt 2.2). Die Topics erfassen Korrelationen zwischen den visuellen Merkmalen. Für das Ermitteln von Suchergebnissen für eine Anfrage wird die Merkmalsrepräsentation dieser Anfrage nun in den Topic-Raum jeder Dokumentseite transformiert und dort der Abstand zu den Patches dieser Seite berechnet.

Diese Word Spotting Methode wird in [RATL15] verbessert, indem für jede Dokumentseite Patches in mehreren Größen extrahiert werden, um Queries in unterschiedlichen Größen geeigneter bearbeiten zu können. Desweiteren wird Product Quantization Indexing [JDS11] verwendet, um den Speicheraufwand und die Retrieval-Zeit für die visuellen Merkmalsrepräsentationen der Patches zu reduzieren.

5.2.2 *Sequenz-Modellierung über Hidden Markov Modelle*

In [RSP12b] wird ein Wortbild durch eine Sequenz von Merkmalsvektoren beschrieben. Der Ablauf des im Folgenden beschriebenen Verfahrens ist in Abbildung 16 verdeutlicht. Dabei wird ein Fenster über das Wortbild geschoben und an jeder Position ein Merkmalsvektor gebildet. Die dabei verwendeten Local Gradient Histogram-Merkmale (LGH, siehe dazu Abschnitt 4.1) sind dem SIFT-Deskriptor ähnliche Merkmale, die ebenfalls auf Pixelebene Gradientenrichtungen quantisieren und somit Orientierungen der Schrift erfassen [RP08]. Für jedes Wortbild wird ein semikontinuierliches Hidden Markov Modell (SC-HMM) trainiert. Die Zustände aller Modelle teilen sich dabei eine gemeinsame Menge von Normalverteilungen für die Bestimmung der Emissionsdichten, wobei sie diesen Normalverteilungen Gewichte zuweisen. Diese gemeinsame Mischverteilung (universal GMM [RSP12b]) wird in einem Vorverarbeitungsschritt aus einer Menge von Merkmalsvektoren bestimmt. Ein Wortbild ist durch sein SC-HMM und somit durch die Verteilungsgewichte in dessen Zuständen beschrieben. Die Ähnlichkeit zwischen zwei Wortbildern wird demnach bestimmt, indem die Gewichte ihrer SC-HMMs als Vektorsequenzen dargestellt werden, deren Abstand durch Dynamic Time Warping bestimmt wird. Dynamic Time Warping ist eine Methode zur Berechnung des Abstandes zwischen zwei Sequenzen [RM03].

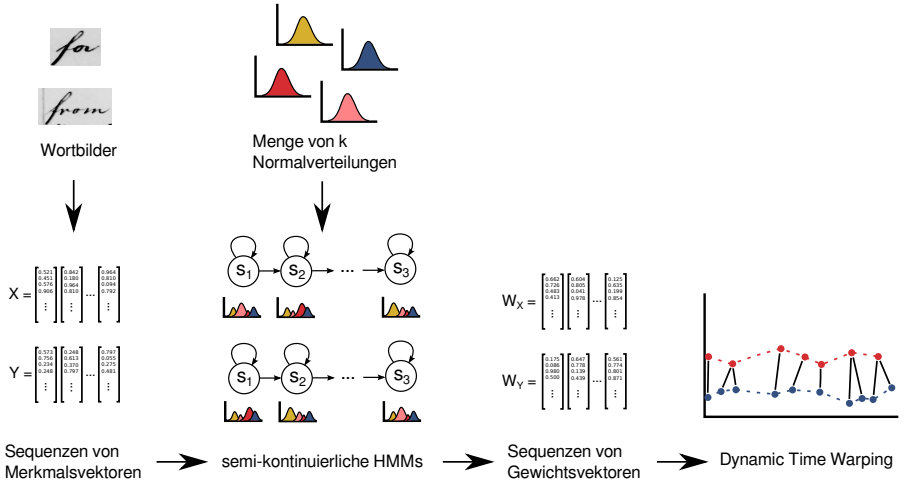


Abbildung 16: Vergleich von Wortbildern durch modellbasierte Sequenzabstände [RSP12b]. Für zwei zu vergleichende Wortbilder (links oben) wird jeweils eine Merkmalsvektorsequenz bestimmt (X bzw. Y) und durch ein semikontinuierliches Hidden Markov Modell beschrieben. Die Modelle teilen sich eine gemeinsame Mischverteilung, bestehend aus k Normalverteilungen, zur Modellierung ihrer Emissionsdichten. Sie unterscheiden sich somit lediglich in den Mischverteilungsgewichten, dargestellt durch die unterschiedliche Größe der farbigen Verteilungsdichten. Anhand dieser Gewichte, repräsentiert in einer weiteren Sequenz von Vektoren (W_X bzw. W_Y), wird mit Dynamic Time Warping der Abstand der beiden Wortbilder ermittelt.

5.2.3 *Exemplar-SVMs*

In [AGFV14b] wird ein Verfahren vorgestellt, das Exemplar-SVMs (siehe Abschnitt 2.3) und HOG-Deskriptoren (Histogram of Oriented Gradients) [DT05] für ein segmentierungsfreies Query-by-Example Word Spotting benutzt. HOG-Deskriptoren gleichen der Anwendung von SIFT-Deskriptoren, wenn diese in einem regelmäßigen Raster aus einem Wortbild extrahiert werden (vgl. Abschnitt 4.2.1). Dazu wird eine Dokumentseite in ein regelmäßiges Raster von Zellen aufgeteilt, die jeweils durch einen HOG-Deskriptor beschrieben werden. Zur Bearbeitung einer Anfrage wird ein Fenster über die Dokumentseite geschoben und somit Patches gebildet, deren visuelle Repräsentation die Gradientenorientierungen der im Fenster befindlichen HOG-Deskriptoren zustande kommt. Die Größe des Fensters (und somit die Anzahl der erfassten Deskriptoren je Patch) hängt nur von der Größe des Anfragebildes ab und ist somit für jede Anfrage unterschiedlich. Anschließend wird eine SVM trainiert, welche das

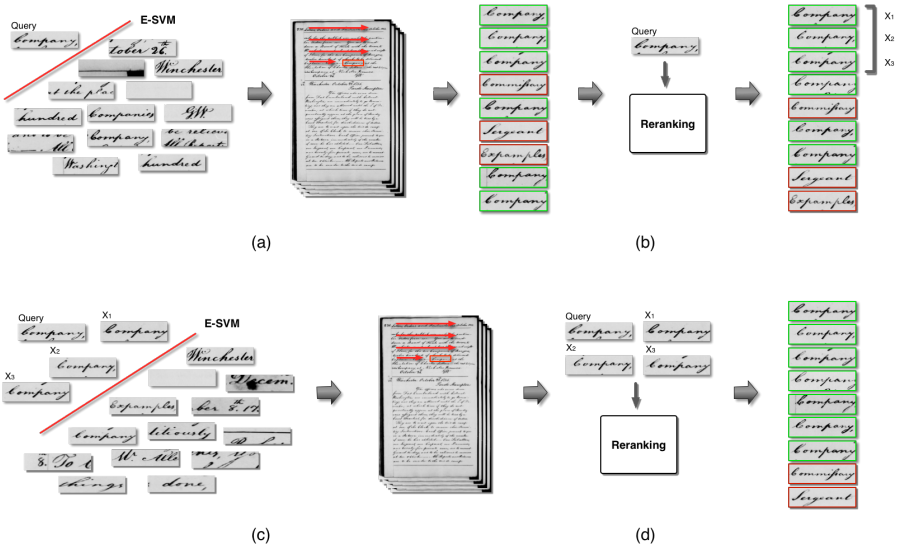


Abbildung 17: Query-by-Example Word Spottings mit Exemplar-SVMs (aus [AGFV14b]). (a) Eine SVM wird trainiert, um das Query-Bild von einer randomisierten Auswahl von Wortbildern zu trennen. (b) Mit dieser SVM wird ein Ranking aller Wortbilder des Dokuments durchgeführt, welches durch Anwendung einer neuen Merkmalsrepräsentation verbessert wird. (c) Die besten Ergebnisse des Rerankings aus Schritt (b) werden zusammen mit der Query für ein erneutes Training der SVM verwendet. (d) Anhand dieser SVM findet ein finales Reranking statt.

Query-Bild, als einziges positives Beispiel im Training, von zufällig gewählten Patches aus dem Dokument-Korpus trennt. Mit dieser Exemplar-SVM wird eine Rangfolge aller Patches aufgestellt, die durch den Abstand zur Anfrage ermittelt wird (siehe Abbildung 17, Teil (a)). Für die Patches mit höchster Relevanz (kleinstem Abstand) wird ein Reranking durchgeführt, indem nur für diese Patches Fisher Vektoren (siehe Abschnitt 4.1) berechnet werden, welche rechenintensiver, als HOG-Deskriptoren sind [AGFV14b]. Anhand der neuen Merkmalsrepräsentationen wird eine neue Rangfolge bestimmt (Abbildung 17, Teil (b)). Es wird nun angenommen, dass die besten Ergebnisse in dieser Rangfolge Treffer für die Anfrage sind. Mithilfe dieser Treffer und dem Query-Bild selber als positive Beispielmenge wird die anfangs trainierte SVM neu trainiert (Abbildung 17, Teil (c)) und anschließend ein erneutes Reranking durchgeführt (Abbildung 17, Teil (d)), das die finale Rangfolge der Patches bestimmt.

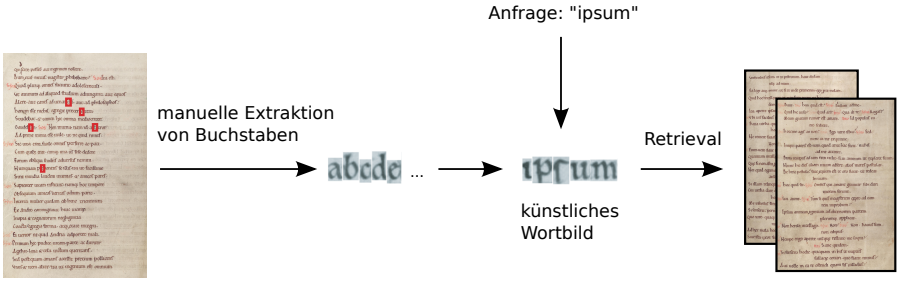


Abbildung 18: Vorgehensweise der synthetischen Query-Generierung in verschiedenen Query-by-String Verfahren [ETF+05, KGN+07, LOLE09]. Anhand eines manuell zusammengestellten Alphabets von Buchstabenbildern wird für die textuelle Anfrage künstlich ein Wortbild erstellt. Mit diesem Wortbild wird anhand des Query-by-Example Prinzips in dem Dokument gesucht (Bilder aus [ETF+05, Oxf]).

5.3 QUERY-BY-STRING

Beim Query-by-String Prinzip (QbS) wird ein Dokumentbild nach Vorkommen eines Wortes durchsucht, welches in Form einer Zeichenkette gegeben ist. Im Gegensatz zu Anfragen nach dem QbE Prinzip sind somit theoretisch auch Anfragen von Wörtern möglich, welche nicht im Dokument vorkommen. Die meisten QbE Verfahren sind ohne eine annotierte Stichprobe von Wortbildern anwendbar, da sie ein Retrieval durchführen, welches lediglich auf visuellen Merkmalen basiert. Beispiele dafür wurden in Abschnitt 5.2 vorgestellt. Im Gegensatz dazu erfolgt bei QbS Verfahren ein Übergang von der textuellen Domäne der Anfrage in die visuelle Domäne der Wortbilder, wofür in den meisten Fällen eine solche Stichprobe benötigt wird [RSP12b]. Dazu wird von vielen Verfahren eine Abbildung zwischen textuellen und visuellen Merkmalsrepräsentationen gelernt, welche diesen Domänenübergang ermöglicht. Die textuelle Merkmalsrepräsentation von Wörtern ist eine zentrale Eigenschaft der im Folgenden vorgestellten QbS Word Spotting Verfahren aus der Literatur. In Abschnitt 5.3.1 wird dazu die Query aus Bildern von Buchstaben künstlich nachgestellt. In Abschnitt 5.3.2 werden anhand der textuellen Informationen Hidden Markov Modelle für einzelne Buchstaben trainiert. Die Verfahren in Abschnitt 5.3.3 und Abschnitt 5.3.4 berechnen Merkmalsrepräsentation für jedes Wort, welche im Wesentlichen die Verteilung der Buchstaben erfassen.

5.3.1 *Synthetisches Querybild*

Eine Klasse von Query-by-String Verfahren versucht auf verschiedene Weise, künstlich ein Bild der angefragten Zeichenkette zu erstellen,

mit welchem dann nach dem Query-by-Example Prinzip das Dokument durchsucht wird [ETF+05, KGN+07, LOLE09, RSP12a]. Das allgemeine Vorgehen ist schematisch in Abbildung 18 dargestellt.

In [ETF+05] wird dafür in einem manuellen Vorverarbeitungsschritt das Bild eines jeden Buchstaben aus dem Dokument extrahiert, welcher in diesem Dokument auftritt. Für die Modellierung der Wortbilder des Dokuments werden n-Gramm Modelle und Hidden Markov Modelle verwendet. Bei der Bearbeitung der Anfrage wird das Dokument zeilenweise durchsucht und Zeilen, welche durch das HMM eine hohe Wahrscheinlichkeit zugewiesen bekommen, das gesuchte Wort zu enthalten, werden als Ergebnis geliefert. In [KGN+07] wird in einem ähnlichen Verfahren jedes Wortbild durch einen Merkmalsvektor beschrieben, indem das Wortbild in Zonen eingeteilt wird, in denen jeweils die Pixeldichte des Schriftzugs im Verhältnis zum Hintergrund ermittelt wird. Die Werte für die einzelnen Zonen werden in einem Vektor zusammengetragen. Für eine Anfrage wird erneut synthetisch ein Wortbild aus Buchstabenbildern zusammengesetzt und der Merkmalsvektor für dieses Wortbild bestimmt. Zur Auswertung der Anfrage wird der Vektorabstand der Merkmalsvektoren zwischen Anfrage und allen Wortbildern aus dem Dokument ausgewertet, welches dem typischen Vorgehen eines QbE Verfahrens gleicht. In [LOLE09] wird die Anzahl der zu vergleichenden Stellen im Dokument reduziert, indem sogenannte Zones of Interest (ZOI) für die künstlich zusammengesetzte Query bestimmt werden. Ähnliche Stellen zu diesen ZOI werden im Dokument gesucht und anstelle aller Patches für die weitere Verarbeitung ausgewertet.

Anstatt der manuellen Zusammenstellung eines Alphabets aus Buchstabenbildern wird in [RSP12a] auf die Generierung ganzer Wörter zurückgegriffen. Die Query wird dazu mittels 25 Computerschriftarten gerendert. Jedes entstehende Wortbild wird anschließend als separate Anfrage nach dem Query-by-Example Prinzip behandelt. Dieses Verfahren ist in großem Maße von der Ähnlichkeit der im Dokument verwendeten Schrift und den zur Verfügung stehenden Computerschriftarten abhängig, spart jedoch im Idealfall den manuellen Vorbereitungsschritt der Extraktion von Buchstabenbildern der anderen genannten Verfahren.

5.3.2 *Word Spotting mit Zeichen-HMMs*

In [FKFB12] werden Hidden-Markov-Modelle (HMM) für jedes Zeichen eingesetzt, um ein Word Spotting Verfahren umzusetzen, welches auf segmentierten Dokumentzeilen arbeitet. Die Parameter der HMMs werden anhand einer Menge von transkribierten Beispieldaten trainiert. Für jede Dokumentzeile aus dieser Trainingsmenge wird an jeder Position eines über die Zeile geschobenen Fensters ein Merkmalsvektor berechnet. Dies ist in Abbildung 5.19(a) dargestellt, wo-

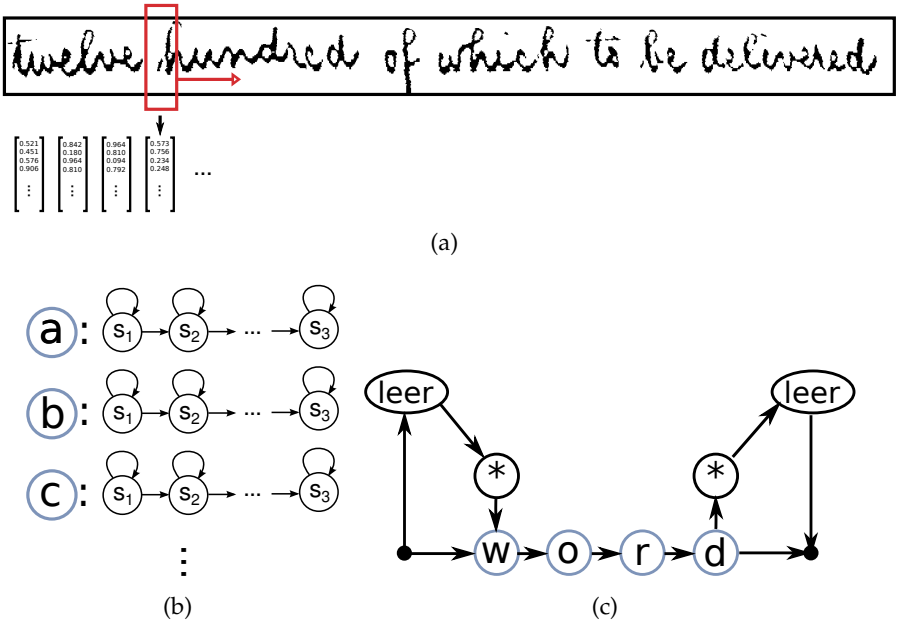


Abbildung 19: Query-by-String Word Spotting mit Zeichen-HMM (nach [FKFB12]). (a) Aus einer Dokumentzeile wird eine Merkmalsvektorsequenz extrahiert. Dazu wird ein Fenster (rot) über die Zeile geschoben und an jeder Position ein Merkmalsvektor berechnet. (b) Aus den in (a) berechneten Merkmalsvektorsequenzen für alle Dokumentzeilen aus einer Menge von Beispieldaten, wird für jeden Buchstaben ein Zeichen-HMM berechnet. Ein HMM ist dabei als blauer Kreis dargestellt, die sich darin befindlichen Zustände und Übergänge rechts daneben. (c) Ein Query-HMM setzt sich aus einzelnen Zeichen-HMMs (blau) zusammen, sowie speziellen HMMs für Leerzeichen („Leer“) und nicht relevante Buchstaben („∗“).

bei die aktuelle Fensterposition in rot markiert ist. Die Merkmale stammen dabei aus dem Bereich der Offline-Handschrift Erkennung [MB01]. Auf diese Weise entsteht für jede Zeile eine Sequenz von Merkmalsvektoren. Anhand der Sequenzen aller Zeilen wird nun für jedes Zeichen ein HMM trainiert (siehe Abbildung 5.19(b)), welches für dieses Zeichen die Verteilungsdichte der Merkmalsausprägungen modelliert. Bei der Auflösung einer Anfrage durch das Verfahren, wird dynamisch ein Query-HMM erstellt, welches für die Beispielanfrage „word“ in Abbildung 5.19(c) abgebildet ist. Dieses Modell besteht aus den korrespondierenden Zeichen-HMMs (blaue Markierung) für jedes Zeichen in der Query sowie besonderen Modellen für Leerzeichen und nicht relevante Zeichen, welche in Abbildung 5.19(c) durch die Markierungen „Leer“ bzw. „∗“ markiert sind. Mit diesem Query-HMM wird jede Dokumentzeile ausgewertet. An-

hand der Scores des Query-HMM werden zum Abschluss die Zeilen mit der höchsten Wahrscheinlichkeit, das Anfragewort zu enthalten, ermittelt. Zudem wird die wahrscheinlichste Position des Wortes innerhalb dieser Zeilen bestimmt.

5.3.3 *Latent Semantic Analysis*

Aldavert et al. beschreiben in [ARTL13] eine Query-by-String Word Spotting Methode, die anhand der Latent Semantic Analysis (LSA, siehe Abschnitt 2.2) einen Unterraum von visuellen und textuellen Merkmalsrepräsentationen von Wortbildern bzw. Zeichenketten bestimmt, in dem die Korrelation zwischen visuellen und textuellen Merkmalen modelliert wird. Für die Modellbildung, d.h. die Berechnung des Unterraums, wird eine transkribierte Beispielmenge von Wortbildern benötigt.

Die textuelle Repräsentation wird anhand von n-Grammen von Buchstaben gebildet. Ein n-Gramm ist eine Folge von n Buchstaben. Bei der Bestimmung von 2-Grammen (auch: Bigramme) werden beispielsweise alle Paare von aufeinanderfolgenden Buchstaben ermittelt. Aus den Annotationen der Trainingsdaten werden nun alle verschiedenen Uni-, Bi- und Trigramme extrahiert. Aus diesen n-Grammen wird ein Kodebuch gebildet, welches jedes n-Gramm in eine Dimension eines Vektors abbildet. Um nun die textuelle Repräsentation für eine Wortannotation zu bilden, werden alle Vorkommen von Uni-, Bi- und Trigrammen dieser Annotation gezählt und ihre Häufigkeit mit Hilfe des Kodebuchs in die entsprechenden Dimensionen des Vektors geschrieben. Der gesamte Vektor wird anschließend mit der euklidischen Norm (L2) normiert. Auf diese Weise werden im Training textuelle Merkmalsrepräsentationen für die Annotationen aller Wortbilder gebildet und zum Zeitpunkt der Anfrage die textuelle Merkmalsrepräsentation der angefragten Zeichenkette erstellt. Im Training nicht vorkommende n-Gramme werden für Anfragen ignoriert.

Die visuelle Repräsentation wird anhand der Bag-of-Features Methode mit Spatial Pyramid (BoF+SP, siehe Kapitel 4) gebildet. Dafür werden aus allen Dokumentseiten SIFT-Deskriptoren der Größen 20, 30 und 40 Pixel im Abstand von 5 Pixel extrahiert, wobei Deskriptoren mit einer zu geringen durchschnittlichen Gradientenmagnitude, d.h. in kontrastarmen Regionen des Dokuments, verworfen werden. Eine zufällige Auswahl der Deskriptoren wird anschließend mit dem Lloyd-Algorithmus in 4096 Visual Words geclustert. Für ein Wortbild wird die visuelle Repräsentation schließlich erstellt, indem erneut SIFT-Deskriptoren mit der oben beschriebenen Konfiguration extrahiert werden und anschließend mit den Visual Words quantisiert werden. Die Zuweisung erfolgt dabei mittels Locality-constrained Linear Coding (siehe Abschnitt 4.2.2) anhand der drei nächsten Nachbarn.

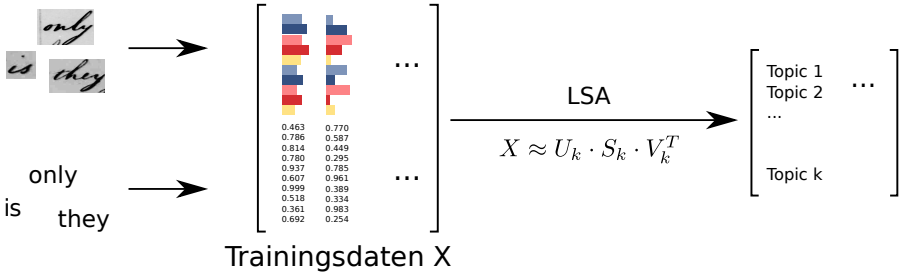


Abbildung 20: Berechnen eines Topic Raums mit der Latent Semantic Analysis (nach [ARTL13]). Für alle annotierten Wortbilder in den Trainingsdaten wird eine gemeinsame Merkmalsrepräsentation aus visuellen- und textuellen Information erstellt und spaltenweise in einer Matrix X zusammengetragen. Aus dieser wird anhand einer Singulärwertzerlegung der gesuchte Topic Raum (rechts) berechnet.

Unter Verwendung einer „9x2/3x2“ Spatial Pyramid Konfiguration (vgl. Abschnitt 4.2.2) wird das Wortbild in Regionen aufgeteilt, in denen jeweils ein Histogramm der Vorkommen der quantisierten SIFT-Deskriptoren gebildet wird. Die Histogramme aller Regionen werden separat L2-normalisiert und anschließend zu einem Vektor konkateniert. Dieser wird zunächst Power-normalisiert (siehe dazu [PSM10]) und abschliessend L2-normalisiert.

Die Berechnung des gemeinsamen Unterraums von visueller- und textueller Merkmalsrepräsentation wird über eine LSA anhand der Trainingsdaten durchgeführt. Dies ist in Abbildung 20 visualisiert. Da für die Berechnung annotierte Wortbilder zur Verfügung stehen, kann hier sowohl die visuelle- als auch textuelle Repräsentation jedes Wortes berechnet werden. Beide Repräsentationen werden für jedes Wortbild konkateniert und schließlich spaltenweise in eine Matrix geschrieben, auf die mit Hilfe der Singulärwertzerlegung eine LSA durchgeführt wird, um einen Topic Raum zu ermitteln (siehe Abschnitt 2.2). In diesem Topic Raum wird die gemeinsame Wortrepräsentation aus visuellen und textuellen Merkmalen auf 2048 Topics projiziert [ARTL13]. Da in den Trainingsdaten für ein Wortbild und die zugehörige Annotation stets beide Merkmalsrepräsentationen gemeinsam berechnet werden können, werden durch den Topic Raum Korrelationen zwischen visuellen und textuellen Merkmalen bestimmt.

Für die Evaluation wird der Praxisfall emuliert, in dem nur ein Teil des Dokuments (manuell) transkribiert wird. Mit diesem Teil wird, wie oben beschrieben, ein Topic Raum berechnet. Für den restlichen Teil der Daten, den Testteil, werden die Annotationen separat als Anfragen verwendet. Die Schritte zur Bearbeitung einer Anfrage sind in Abbildung 21 dargestellt. Für die Wortbilder des Testteils kann,

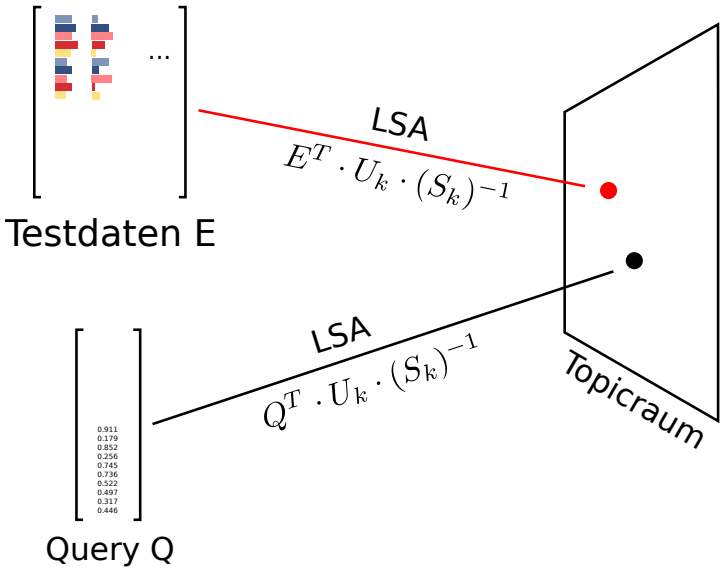


Abbildung 21: Bearbeitung einer Anfrage mittels Topic Raum (nach [ARTL13]). Für Wortbilder E wird nur die visuelle-, für die Anfrage Q nur die textuelle Merkmalsrepräsentation berechnet und zum Vergleich in den Topic Raum transformiert. Die Transformation wurde dabei anhand eines Trainingsschrittes gelernt, in dem visuelle- und textuelle Informationen für jedes Wortbild zur Verfügung stehen. (vgl. Abbildung 20). Die Topic Raum Transformationen der beiden unterschiedlichen Merkmalsrepräsentationen (schwarzer bzw. roter Punkt) können miteinander vergleichen werden, da durch den Topic Raum Korrelationen zwischen visuellen und textueller Merkmalen gelernt wurden.

ohne Annotation, nur die visuelle Merkmalsrepräsentation berechnet werden. In der gemeinsamen Repräsentation wird für diese Daten der textuelle Teil des gemeinsamen Vektors mit Nullen gefüllt. Ähnlich verhält es sich mit den Anfragen, welche aus den Annotationen des Testteils erstellt werden. Hier kann lediglich die textuelle Merkmalsrepräsentation berechnet werden, der visuelle Teil wird mit Nullen aufgefüllt. Die so erstellten Repräsentationen für Testdaten und Anfragen werden in den Topic Raum transformiert. Die Topic Raum-Repräsentationen von Anfrage und Wortbildern können jetzt verglichen werden, da die Topics durch die Korrelationen zwischen visuellen und textuellen Merkmalen berechnet wurden. Im Topic Raum entspricht das Auflösen einer Anfrage dem Nächster-Nachbar-Problem. Dabei wird als Abstandsmaß die Kosinusdistanz verwendet, da im Topic Raum lediglich die Richtung der Vektoren, d.h. die Zugehörigkeit der Vektoren zu verschiedenen Topics, von Bedeutung ist (vgl. Abschnitt 2.2).

5.3.4 *Embedded Attributes*

In [AGFV14a] wird, ähnlich zu [ARTL13], ein Unterraum von visueller und textueller Repräsentation eines Wortes gebildet, in dem diese beiden verschiedenen Repräsentationen verglichen werden können. Dies geschieht anhand eines Attribute Embeddings, bei dem mit Hilfe von Support Vektor Maschinen die gleichen Eigenschaften (Attribute) für Wortbilder und Zeichenketten bestimmt werden.

Als visuelle Merkmalsrepräsentation werden Fisher Vektoren (FV) verwendet (vgl. [PSM10]). Diese sind eine Erweiterung der Bag-of-Features Merkmalsrepräsentation, da sie neben der Häufigkeit von SIFT-Deskriptoren in einem Wortbild auch Informationen zu deren statistischen Verteilung erfassen. Im Gegensatz zu BoF werden für FV die SIFT-Deskriptoren nicht hart quantisiert, sondern durch Verteilungsdichten einer globalen Mischverteilung beschrieben. SIFT-Deskriptoren werden in [AGFV14a] zudem durch die Koordinaten des Punktes erweitert, für den der Deskriptor berechnet wird. Diese Koordinatenerweiterung stattet die entstehende Merkmalsrepräsentation, ähnlich zu der in Abschnitt 4.3 beschriebenen Spatial Pyramid, mit räumlichen Informationen der Deskriptoren im repräsentierten Wortbild aus. Im Folgenden wird die Merkmalsrepräsentation aus Fisher Vektoren und Koordinatenerweiterung mit FV+K abgekürzt.

Die textuelle Merkmalsrepräsentation ist zentraler Bestandteil der Methode, da von ihr die namensgebende Einbettung von Attributen ausgeht. Als Repräsentation für Zeichenketten werden sogenannte PHOC-Vektoren (Pyramidal Histogram of Characters) verwendet. Hierbei handelt es sich um einen Vektor aus $\{0,1\}^n$, der anzeigt, ob und in welchem Teil der Zeichenkette ein Buchstabe vorkommt. Dabei ist n die Anzahl der Attribute, welche im Folgenden hergeleitet wird. Die Bestimmung eines PHOC-Vektors ist in Abbildung 22 anhand des Wortes „beyond“ verdeutlicht. Ein Level beschreibt die Anzahl der Regionen, in die die Zeichenkette aufgeteilt wird (zwei Regionen auf Level 2 usw.). Ein Buchstabe kommt in einer Region vor, wenn sein Bereich mehr als 50% mit dem der Region überlappt. Dies wird am Beispiel des Wortes „beyond“ deutlich: der Buchstabe y wird in Level 2 der ersten von beiden Regionen zugeordnet, da sein Bereich zu mehr als 50% mit der linken Hälfte des Wortes überlappt. In der Praxis werden die Level 2, 3, 4 und 5 verwendet. Zudem werden auf Level 2 analog die Vorkommen der 50 häufigsten englischen Bigramme (vgl. textuelle Merkmalsrepräsentation in Abschnitt 5.3.3) ermittelt. Somit hat der PHOC-Vektor jeder Zeichenkette $n = 604$ Dimensionen (26 Buchstaben und 10 Zahlen in insgesamt 14 Regionen,zusätzlich 50 Bigramme in 2 Regionen) und stellt ein binäres Histogramm über Buchstabenvorkommen in dieser Zeichenkette dar.

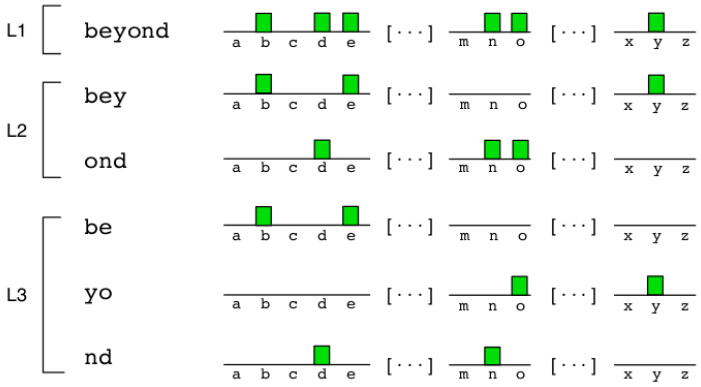


Abbildung 22: Bildung eines PHOC-Vektors (aus [AGFV14a]). Die Zeichenkette „beyond“ wird in eine Anzahl Regionen, hier in Zeilen dargestellt, entsprechend des betrachteten Levels (L1-L3) geteilt. Für jede Region wird nun ein binäres Histogramm über das Auftreten der einzelnen Zeichen gebildet. Der finale PHOC-Vektor besteht aus den konkatenierten Histogrammen der Regionen jedes Levels.

In der Trainingsphase des Word Spotting Verfahrens wird mit den visuellen Repräsentationen und den PHOC-Vektoren ein Attribute Embedding gelernt. Dazu wird für jede Dimension eines PHOC-Vektors eine lineare Zweiklassen-SVM trainiert (vgl. Abschnitt 2.3). Das Lernen des Embeddings ist in Abbildung 23 dargestellt. Jede Dimension des PHOC-Vektors macht eine Aussage über das Vorkommen bestimmter Buchstaben im repräsentierten Wort. So zeigt beispielsweise eine Dimension an, dass in der zweiten Hälfte des Wortes ein *s* auftritt (Level 2, Region 2, Eintrag für den Buchstaben *s*). Diese Information soll als Attribut für die visuellen Repräsentationen gelernt werden. Dadurch, dass aus der Annotation jedes Wortbildes der PHOC-Vektor generiert wird, ist für jedes Wortbild bekannt, ob ein Attribut erfüllt ist oder nicht. Da PHOC-Vektoren nur binäre Werte enthalten, können die visuellen Repräsentationen für jedes dieser Attribute in zwei Klassen aufgeteilt werden, davon abhängig, ob der entsprechende Eintrag im PHOC-Vektor für dieses Attribut 0 oder 1 ist. In der Abbildung ist das Attribut für die Wörter „Letters“ und „is“ erfüllt, nicht aber für „they“. Die visuellen Merkmalsrepräsentationen für die Wortbilder von „Letters“ und „is“ werden somit der positiven Beispielmenge der zu trainierenden SVM zugeordnet, das Wortbild zu „they“ der negativen.

Zum Zeitpunkt einer Anfrage wird für die angefragte Zeichenkette der PHOC-Vektor bestimmt. Anhand der gelernten Attribut-SVMs wird für jedes Wortbild des zu durchsuchenden Dokuments

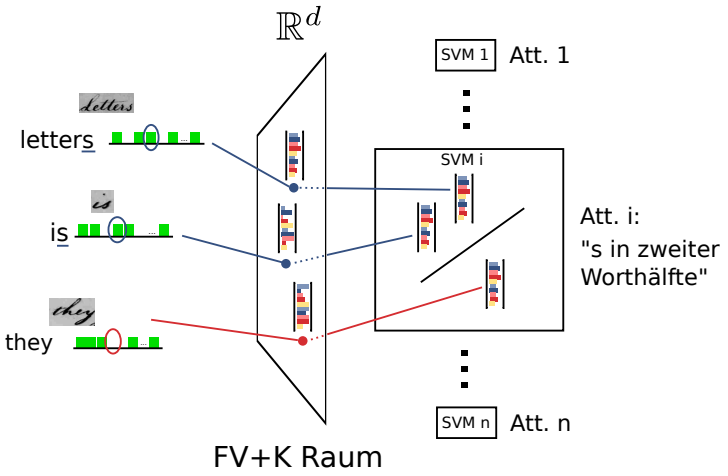


Abbildung 23: Trainieren von Attribut-SVMs mit Hilfe von PHOC-Vektoren (nach [AGFV14a]). Jede PHOC-Dimension ist ein Attribut, welches für Wortbilder durch eine SVM bestimmt werden soll. Für die annotierten Wortbilder der Trainingsdaten (links) ist bekannt, ob ein Attribut erfüllt ist (blaue Markierung) oder nicht (rote Markierung). Nach dieser Information werden die Merkmalsrepräsentationen der Wortbilder (FV Raum) in zwei Klassen eingeteilt und die entsprechende Attribut-SVM trainiert.

das Attribute Embedding durchgeführt, indem die visuelle Repräsentation dieses Wortbildes mit jeder SVM ausgewertet wird und die SVM-Scores in einen Vektor eingetragen werden. Der PHOC-Vektor der Anfrage und die Attribut-Vektoren der Wortbilder haben jeweils $n = 604$ Dimensionen und können demnach direkt miteinander verglichen werden, was nach Experimenten in [AGFV14a] bereits zu guten Ergebnissen führt. Es wird die Kosinusdistanz als Distanzmaß verwendet.

Eine Verbesserung für diesen direkten Vergleich zwischen binären PHOC-Vektoren und SVM-Scores stellt die Kalibrierung der SVM-Scores durch Platt's Scaling (siehe Abschnitt 2.3) dar. Dabei wird für jede SVM unabhängig voneinander der SVM-Score normalisiert. Der SVM-Score ist der Abstand eines zu klassifizierenden Vektors von der trennenden Hyperebene. Nach Normalisierung des Scores auf das Intervall $[0, 1]$ kann er als Wahrscheinlichkeit dafür interpretiert werden, dass der Vektor einer der beiden Klassen angehört.

Eine andere Möglichkeit zur Verbesserung des Retrieval Ergebnisses besteht im Lernen eines Unterraumes, in dem der Abstand zwischen PHOC-Vektor und Attribut-Vektor per linearer Regression oder

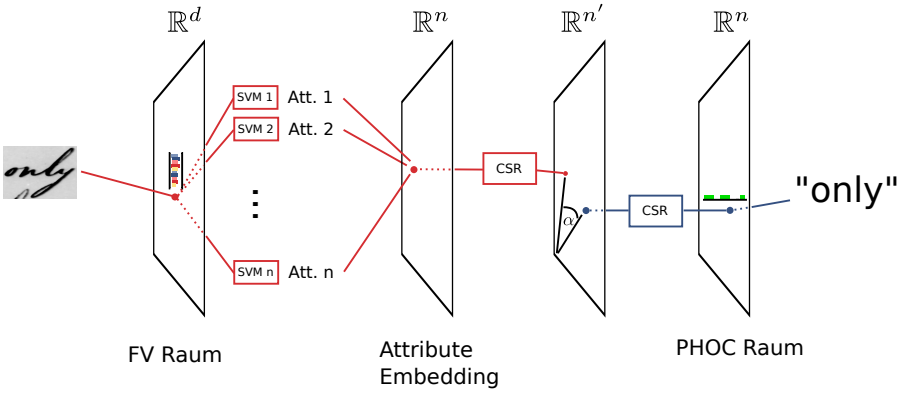


Abbildung 24: Transformation von Wortbild und Zeichenkette in einen gemeinsamen Unterraum (nach [AGFV14a]). Für die FV+K Repräsentation eines Wortbildes (links) wird über die Attribut-SVMs ein Attribute Embedding berechnet. Die Zeichenkette wird als PHOC-Vektor repräsentiert (PHOC Raum, rechts). Der Abstand beider Repräsentationen wird in einem Unterraum ermittelt, welcher durch Common Subspace Regression berechnet wurde. Beide Merkmalsrepräsentationen werden in diesen Unterraum transformiert und dort anhand der Kosinusdistanz (dargestellt durch den Winkel α) verglichen.

Common Subspace Regression (CSR) minimiert wird. Bei der CSR werden durch Lösen des Optimierungsproblems [AGFV14a]

$$\text{argmin}_{\mathbf{U},\mathbf{V}} \frac{1}{2}||\mathbf{U}^T\mathbf{A} - \mathbf{V}^T\mathbf{B}||_F^2 + \frac{1}{2}\alpha||\mathbf{U}||_F^2 + \frac{1}{2}\beta||\mathbf{V}||_F^2 \qquad (18)$$

Transformationsmatrizen $\mathbf{U}, \mathbf{V}$ berechnet, die den Abstand zwischen den visuellen Repräsentationen $\mathbf{A}$ der Wortbilder und den entsprechenden PHOC-Vektoren $\mathbf{B}$ der Annotationen minimieren. Die beiden letzten Summanden sind Regularisierungsterme, die über die Parameter α, β eingestellt werden. Zudem müssen die Matrizen $\mathbf{U}, \mathbf{V}$ jeweils orthogonal sein, wodurch triviale Lösungen, wie beispielsweise Nullmatrizen, ausgeschlossen werden. Die Notation $||\mathbf{U}||_F^2$ beschreibt die Frobenius-Norm der Matrix $\mathbf{U}$. Im Fall der linearen Regression wird nur eine Transformationsmatrix für die visuellen Merkmalsrepräsentationen berechnet. Durch Lösen des Optimierungsproblems werden, im Unterschied zum Platt's Scaling, nicht nur die SVM-Scores kalibriert, sondern gleichzeitig die Korrelation zwischen Attributen ausgenutzt [AGFV14a]. Das Attribute Embedding Verfahren ist anhand der Common Subspace Regression Variante schematisch für eine Anfrage in Abbildung 24 dargestellt. Durch die perspektivisch verschobenen Vierecke sind, wie in vorherigen Abbildungen, die verschiedenen Vektorräume dargestellt. Für das Wortbild wird

zunächst die FV Merkmalsrepräsentation berechnet. Durch Klassifizierung dieser Repräsentation durch alle visuellen Attribut-SVMs entsteht ein SVM-Score-Vektor. Für die Zeichenkette wird die PHOC-Vektor Repräsentation berechnet. PHOC-Vektor und SVM-Score-Vektor werden in einen Unterraum transformiert und dort durch die Kosinusdistanz verglichen. Der Unterraum wird, wie oben beschrieben, durch eine Common Subspace Regression berechnet.

5.4 DISKUSSION

In den vorangegangenen Abschnitten wurden Verfahren aus der Literatur für das Query-by-Example und Query-by-String Word Spotting vorgestellt. In Abschnitt 5.3.3 und Abschnitt 5.3.4 wurden ausführlich zwei Query-by-String Verfahren beschrieben, die auf der Bestimmung eines gemeinsamen Vektorraums für Merkmalsrepräsentationen unterschiedlicher Domänen (visuell und textuell) basieren. Die LSA-Methode [ARTL13] bestimmt diesen Vektorraum über eine Singulärwertzerlegung einer Matrix, in der für jedes Wortbild die gemeinsame Merkmalsrepräsentation aus visuellen und textuellen Merkmalen vorliegt. Durch unüberwachtes Lernen werden dabei Topics berechnet, die Korrelationen zwischen visuellen und textuellen Merkmalen aufdecken. Dieses Vorgehen unterscheidet sich von der Attribute Embedding Methode nach [AGFV14a]. Hier sind die sogenannten Attribute, Eigenschaften, die das Vorkommen von Buchstaben in Wörtern erfassen, explizit vorgegeben und werden durch überwachtes Lernen von SVMs für Wortbilder ermittelt. Während bei der LSA-Methode die Eigenschaften des berechneten Vektorraums durch das unüberwachte Lernen bestimmt werden, ist bei der Attribute Embedding Methode die Möglichkeit größer, einen direkten Einfluss auf die bestimmten Eigenschaften zu nehmen, die berechnet werden sollen. Trotz dieses Unterschieds werden für beide Verfahren annotierte Beispieldaten zum Training des jeweiligen Modells benötigt. Wie im folgenden Kapitel 6 zu sehen sein wird, lassen sich die Ideen dieser beiden Verfahren gut auf das Word Spotting mit Online-Handschrift Anfragen (QbO) übertragen. Dazu wird eine Abbildung zwischen Online-Handschrift Merkmalen und visuellen Merkmalen berechnet. Zudem wird geklärt, warum sich die vorgestellten Query-by-Example Methoden und die Query-by-String Methoden mit künstlicher Erstellung eines Wortbildes aus der Anfrage nicht für das QbO Word Spotting adaptieren lassen.

6

WORD SPOTTING MIT ONLINE-HANDSCHRIFT ANFRAGEN

Das Word Spotting mit Online-Handschrift Anfragen ist eine neue Form des Word Spotting, welche in dieser Arbeit zum ersten Mal beschrieben und evaluiert wird. Wie in der Einleitung (Kapitel 1) bereits erläutert, basiert es auf der Idee, Anfragen an ein Word Spotting System nicht durch Angabe eines Beispiel-Wortbildes oder einer Zeichenkette zu stellen, sondern die Anfragen durch Online-Handschrift zu formulieren. Dies wird *Query-by-Online-Trajectory* (QbO) genannt. Im vorangegangenen Kapitel 5 wurden Query-by-Example und Query-by-String Verfahren aus der Literatur vorgestellt. Um Online-Handschrift Anfragen zu bearbeiten, müssen domänenübergreifend, ausgehend von Online-Handschrift Trajektorien, Wortbilder durchsucht werden, für welche nur visuelle Merkmale berechnet werden können. Die in Abschnitt 5.2 vorgestellten Query-by-Example Verfahren eignen sich daher nicht für diese Aufgabe, da sie zumeist auf dem direkten Vergleich zwischen visuellen Repräsentationen von Wortbildern basieren und somit kein Mittel für einen domänenübergreifenden Vergleich mit sich bringen. Aus einem ähnlichen Grund sind die in Abschnitt 5.3.1 beschriebenen Verfahren nicht geeignet, da hier die Anfrage durch das künstliche Zusammensetzen eines Anfrage-Wortbildes lediglich auf das QbE Problem zurückgeführt wird und somit das gleiche, zuvor beschriebene Problem auftritt.

In Abschnitt 5.3.3 und Abschnitt 5.3.4 wurden zwei Query-by-String Verfahren beschrieben, die auf der Bestimmung eines gemeinsamen Vektorraums für Merkmalsrepräsentationen unterschiedlicher Domänen (visuell und textuell) basieren. Diese Verfahren eignen sich sehr gut zur Umsetzung des QbO Prinzips, da sie eine Abbildung von der textuellen Domäne in die visuelle Domäne berechnen. Analog dazu wird im folgenden Kapitel beschrieben, wie mithilfe dieser zwei Verfahren eine Abbildung von der Online-Handschrift Merkmalsdomäne in die visuelle Merkmalsdomäne berechnet wird. Die Methode zur Auswertung einer Anfrage ist zudem in beiden Verfahren nicht von der speziell gewählten Merkmalsrepräsentation abhängig, diese kann somit modifiziert werden. Beide Verfahren arbeiten segmentierungsbasiert, d.h. sie verwenden bereits im Vorfeld aus dem Dokument segmentierte Wortbilder. Für diese Wortbilder sowie für Online-Handschrift Trajektorien werden Merkmalsrepräsentationen berechnet, auf die in Abschnitt 6.1 eingegangen wird. Das erste QbO Word Spotting Verfahren berechnet mittels Latent Semantic Analysis einen gemeinsamen Vektorraum der beiden Merkmalsrepräsenta-

tionen und wird in Abschnitt 6.2 beschrieben. Das zweite Verfahren (Abschnitt 6.3) ermittelt über ein Attribute Embedding Eigenschaften von Wortbildern und Online-Handschrift, worüber beide Merkmalsrepräsentationen vergleichbar sind.

6.1 MERKMALSREPRÄSENTATIONEN

In diesem Abschnitt wird die Berechnung der Merkmalsrepräsentationen für Wortbilder und Online-Handschrift Trajektorien beschrieben, welche in den evaluierten QbO Word Spotting Verfahren zum Einsatz kommen. Die Bag-of-Features Merkmalsrepräsentation (vgl. Kapitel 4) hat sich in mehreren Word Spotting Verfahren aus der Literatur bereits als geeignet erwiesen und wird in dieser Arbeit zur Repräsentation von Wortbildern verwendet. Ihre Konfiguration wird in Abschnitt 6.1.1 beschrieben. In dieser Arbeit wird der Bag-of-Features Ansatz zum ersten Mal auch auf Online-Handschrift angewendet. Dabei wird eine Trajektorie über Vorkommen typischer lokaler Schriftverläufe charakterisiert. Diese neue Merkmalsrepräsentation von Online-Handschrift wird *Bag-of-Online-Features* (BoOF) genannt und in Abschnitt 6.1.2 beschrieben.

6.1.1 *Visuelle Merkmalsrepräsentation*

Für diese Arbeit wird der bekannte Bag-of-Features Ansatz mit einer Spatial Pyramid als Repräsentation für Wortbilder verwendet. Diese Repräsentation wurde in Kapitel 4 ausführlich erläutert. In [ARTL13] werden SIFT-Deskriptoren mit drei unterschiedlichen Größen in einem dichten Grid aus den Dokumentseiten extrahiert. Deskriptoren aus kontrastarmen Regionen der Dokumentseiten werden dabei ignoriert. Aus den SIFT-Deskriptoren werden durch den Lloyd Algorithmus 4096 Visual Words gebildet. Die Zuweisung von SIFT-Deskriptoren zu Visual Words bei der Quantisierung erfolgt dabei mittels Locality-constrained Linear Coding (siehe dazu Abschnitt 4.2.2). In der vorliegenden Arbeit wird eine von der dieser Vorlage abweichende, vereinfachte Parameterkonfiguration verwendet. Die Experimente in Kapitel 7 werden zeigen, dass mit dieser einfacheren Konfiguration bessere Ergebnisse, als in der Vorlage erzielt werden. Es wird eine einheitliche Größe von 40 Pixeln für jeden SIFT-Deskriptor verwendet, anstatt einer mehrstufige Konfiguration unterschiedlicher Größen. Bei der Quantisierung erfolgt die Zuweisung von SIFT-Deskriptoren zu Visual Words anhand des nächsten Nachbarn (euklidischen Distanz). Bei der Bildung der Spatial Pyramid für ein Wortbild werden, abhängig vom evaluierten Word Spotting Verfahren, ein bis zwei Level unterschiedlicher Auflösung verwendet. Die Auflösung auf jedem Level ist dabei ein zu optimierender Parameter, da durch sie zum

einen die Größe der Merkmalsrepräsentation bestimmt wird, als auch die Genauigkeit, mit der ein Wortbild beschrieben wird. Auch die in [ARTL13] benutzte „9x2/3x2“ Spatial Pyramid wird dabei evaluiert.

6.1.2 *Online-Handschrift Merkmalsrepräsentation*

Analog zu den Ausführungen über die visuelle Repräsentation für Wortbilder in Abschnitt 6.1.1, werden auch Online-Handschrift Trajektorien in dieser Arbeit durch eine holistische Merkmalsrepräsentation repräsentiert. Nach bestem Wissen und Gewissen des Autors dieser Arbeit ist dies der erste Versuch, Online-Handschrift Trajektorien auf diese Weise zu repräsentieren. Die Merkmalsrepräsentation wird *Bag-of-Online-Features* (BoOF) genannt und im Folgenden beschrieben. Sie basiert auf dem Bag-of-Features Ansatz und berechnet typische Vorkommen lokaler Schriftverläufe in Online-Handschrift Trajektorien (Abschnitt 6.1.2). Der BoOF-Repräsentation werden über eine Spatial Pyramid ebenfalls Lokalitätsinformationen hinzugefügt (Abschnitt 6.1.2).

Bag-of-Online-Features

Eine Online-Handschrift Trajektorie wird zunächst mit den in Abschnitt 3.2 beschriebenen Methoden normalisiert, um ungewollte Variabilität in der Handschrift zu reduzieren. Anschliessend wird für jeden Punkt der Trajektorie durch eine Merkmalsberechnung ein Vektor mit den in Abschnitt 3.3 vorgestellten Merkmalen bestimmt, so dass für die gesamte Trajektorie eine Sequenz von Merkmalsvektoren entsteht. Während im Bereich der Online-Handschrift Erkennung typischerweise diese Sequenz z. B. über ein statistisches Modell ausgewertet wird [JMW00, LB06], wird sie in dieser Arbeit durch einen zusätzlichen Schritt zu einer Merkmalsrepräsentation in Form eines einzelnen Vektors fester Größe zusammengefasst. Das Vorgehen basiert auf dem des Bag-of-Features Ansatzes. Hierzu wird zunächst ein Vokabular von typischen lokalen Schriftverläufen einer Trajektorie benötigt. Analog zum visuellen Vokabular des BoF Ansatzes, welches durch Clustern einer großen Menge von lokalen Bilddeskriptoren ermittelt wird, werden dazu Merkmalsvektorsequenzen einer großen Beispielmenge von Trajektorien berechnet. Die einzelnen Merkmalsvektoren werden mittels des Lloyd Algorithmus geclustert (vgl. Abschnitt 2.1). Die finalen Clusterzentren beschreiben Häufungspunkte der Online-Handschrift Merkmalsvektoren im Merkmalsraum und somit prototypische lokale Schriftverläufe in den Trajektorien.

Für eine neue Trajektorie wird nun eine holistische Repräsentation bestimmt, indem zunächst, wie zuvor beschrieben, eine Sequenz der Merkmalsvektoren berechnet wird und die einzelnen Merkmalsvektoren anschliessend anhand des zuvor erstellten Vokabulars quantisiert werden. Die Zuweisung eines Merkmalsvektors zum räumlich

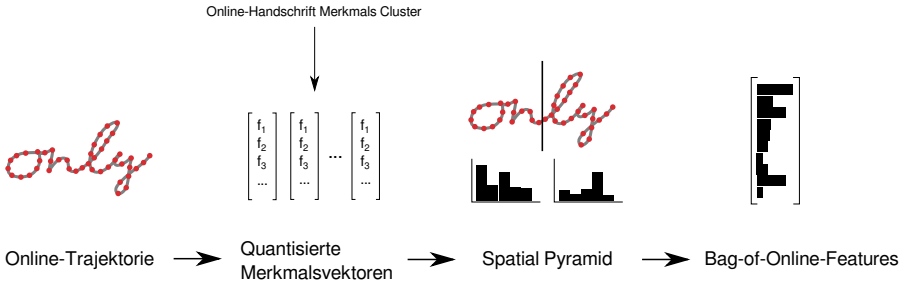


Abbildung 25: Anwendung des Bag-of-Online-Features Ansatz. Für jeden Punkt der Trajektorie wird ein Merkmalsvektor extrahiert. Die entstehenden Vektoren werden anhand des Online-Merkmal-Vokabulars quantisiert. Mit einer Spatial Pyramid wird die Trajektorie in Regionen aufgeteilt, in denen jeweils ein Histogramm der Häufigkeiten der auftretenden quantisierten Merkmalsvektoren gebildet wird. Die finale Merkmalsrepräsentation ist ein Vektor, der alle konkatenierten Histogramme enthält.

nächsten Clusterzentrum geschieht anhand der euklidischen Distanz. Anschliessend wird ein Histogramm über die Häufigkeiten der quantisierten Merkmalsvektoren erstellt. Das Histogramm wird mit der euklidischen Norm auf Einheitslänge normalisiert, um die Länge des Vektors von der Anzahl der quantisierten Merkmalsvektoren unabhängig zu machen. Da die Größe des Histogramms der Anzahl der Cluster im Vokabular entspricht, ist diese Darstellung unabhängig von der Länge der Merkmalsvektorsequenz (bzw. Anzahl der Punkte) einer Trajektorie. Die entstehende Repräsentation wird Bag-of-Online-Features (BoOF) genannt.

Spatial Pyramid für Online-Handschrift

Bei dem in Abschnitt 6.1.2 beschriebenen Verfahren zur Bestimmung einer BoOF-Repräsentation für Online-Handschrift Trajektorien sind die Koordinaten der einzelnen Punkte nicht Teil dieser Repräsentation. Somit lässt sich dieser Repräsentation nicht mehr entnehmen, welche Merkmale (genauer gesagt: welche quantisierten Merkmalsvektoren) an welcher Position der Trajektorie beobachtet wurden. Analog zur Berechnung einer BoF-Repräsentation mit lokalen Bilddeskriptoren (vgl. Abschnitt 4.3) ist diese Information jedoch auch hier wertvoll für die Unterscheidungsfähigkeit der entstehenden Repräsentation. Um der BoOF-Repräsentation Lokalitätsinformationen hinzuzufügen, wird analog zum BoF Ansatz auf die Spatial Pyramid Methode zurückgegriffen. Das gesamte Verfahren zur Berechnung der Merkmalsrepräsentation ist in Abbildung 25 dargestellt. Dazu wird aus einer Online-Handschrift Trajektorie, wie zuvor beschrieben, eine

Merkmalsvektorsequenz extrahiert, deren einzelne Vektoren anhand eines Vokabulars quantisiert werden. Die Größe der Trajektorie wird dann aus der Bounding Box ermittelt, die alle Punkte beinhaltet (vgl. Abschnitt 3.1). In Abhängigkeit zu der Konfiguration der Spatial Pyramid wird diese Bounding Box nun in Regionen aufgeteilt, zu denen die quantisierten Merkmalsvektoren anhand ihrer Koordinaten zugewiesen werden können. Die finale Repräsentation wird gebildet, indem für jede Region ein Histogramm der vorkommenden quantisierten Merkmalsvektoren gebildet wird, jedes dieser Histogramme unabhängig voneinander mit der euklidischen Norm normalisiert wird und die Histogramme anschliessend zu einem gemeinsamen Vektor konkateniert werden. Sie wird im Folgenden auch mit „BoOF+SP“ abgekürzt.

6.2 LÖSUNG MIT DER LSA-METHODE

In [ARTL13] wird eine Query-by-String Word Spotting Methode vorgestellt, welche eine Latent Semantic Analysis zur Berechnung eines gemeinsamen Topic Raums von visueller und textueller Merkmalsrepräsentation verwendet, in dem die Korrelationen zwischen Merkmalen beider Repräsentation durch Topics erfasst werden. Dieses Verfahren wurde in Abschnitt 5.3.3 ausführlich beschrieben. In den einleitenden Ausführungen des aktuellen Kapitels wurde zudem geschlussfolgert, dass diese Word Spotting Methode aufgrund ihrer Struktur und Vorgehensweise geeignet ist um, mit Modifikationen, Anfragen per Online-Handschrift Trajektorien zu ermöglichen. Dazu wird mittels LSA ein Topic Raum von visueller Merkmalsrepräsentation und Online-Handschrift Merkmalsrepräsentation berechnet, in dem beide Merkmalsrepräsentationen auf die selben Topics abgebildet werden. Diese Modifikationen sind in Abbildung 26 und Abbildung 27 dargestellt und werden im Folgenden erläutert.

Um einen gemeinsamen Topic Raum von visuellen- und Online-Handschrift Repräsentationen zu berechnen wird ein Trainingsdatensatz benötigt, welcher zu jedem Wortbild die passende Online-Handschrift Trajektorie gleichen Wortes beinhaltet. Zu jedem dieser Datenpaare werden nun, wie in Abschnitt 6.1 beschrieben, die visuelle- und Online-Handschrift Merkmalsrepräsentation gebildet. Durch Konkatenation der beiden Repräsentationen entsteht eine gemeinsame Merkmalsrepräsentation. Für alle Trainingsdatenpaare wird diese gemeinsame Repräsentation bestimmt und in eine Matrix geschrieben (siehe Abbildung 26, Mitte). Auf dieser Matrix wird, analog zur Beschreibung in Abschnitt 5.3.3, durch eine Singulärwertzerlegung eine Latent Semantic Analysis durchgeführt und somit ein Topic Raum bestimmt.

Mit dem berechneten Topic Raum kann nun ein Retrieval von Wortbildern anhand einer Online-Handschrift Anfrage durchgeführt wer-

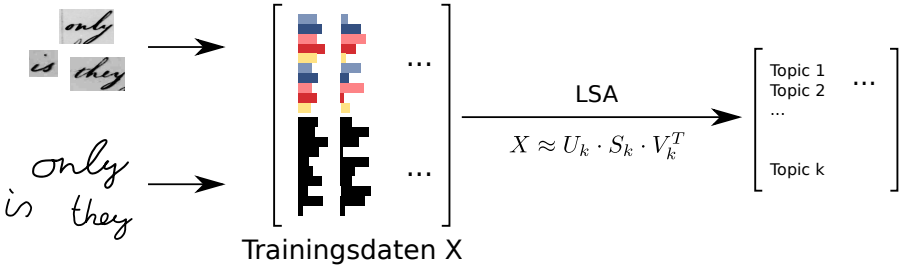


Abbildung 26: Berechnen eines Topic Raumes für Anfragen mit Online-Handschrift Trajektorien mit der LSA (nach [ARTL13]). Für alle Wortbilder und Online-Handschrift Trajektorien in den Trainingsdaten wird eine gemeinsame Repräsentation aus Bag-of-Features und Spatial Pyramid sowie Bag-of-Online-Features und Spatial Pyramid erstellt und in einer Matrix zusammengetragen. Aus dieser wird anhand einer Singulärwertzerlegung der gesuchte Topic Raum berechnet.

den. Die Bearbeitung einer Anfrage ist in Abbildung 27 visualisiert (vgl. Abbildung 21). Dazu wird, wie ähnlich der zuvor beschriebenen Berechnung, ein Vektor für eine gemeinsame Merkmalsrepräsentation erstellt. Für die Anfrage, welche als Online-Handschrift Trajektorie, nicht aber als Wortbild vorliegt, kann keine visuelle Merkmalsrepräsentation bestimmt werden. Daher wird die gemeinsame Repräsentation lediglich mit den Informationen aus der BoOF+SP-Merkmalsrepräsentation gefüllt. Die zu durchsuchenden Wortbilder bieten nur visuelle Informationen, daher werden hierfür die BoF+SP Merkmalsrepräsentationen berechnet und in den entstehenden gemeinsamen Repräsentationen der Online-Handschrift Teil mit Nullen gefüllt. Diese jeweils „halb gefüllten“ Merkmalsrepräsentationen von Trajektorie und Wortbildern werden in den berechneten Topic Raum transformiert. Da durch die Topics Korrelationen zwischen den Merkmalen der visuellen Domäne und der Online-Handschrift Domäne erfasst wurden, liegen Wortbilder und Trajektorien, welche den selben Topics angehören, nah zusammen. Das Retrieval wird daher durch Berechnung der nächsten Nachbarn der Topic Raum-Transformation der Anfrage durchgeführt.

6.3 Lösung mit Embedded Attributes

In Abschnitt 5.3.4 wurde ein Query-by-String Word Spotting Verfahren beschrieben, welches über ein Attribute Embedding von visuellen und textuellen Merkmalsrepräsentationen einen gemeinsamen Vektorraum berechnet, in dem beide Repräsentationen anhand ihrer Attribute verglichen werden können [AGFV14a]. Wie zuvor für die

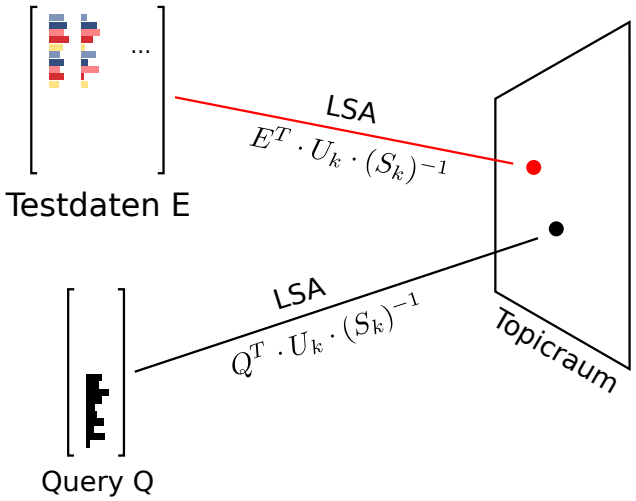


Abbildung 27: Vorgehen beim QbO Word Spotting mit der LSA-Methode nach [ARTL13]. Transformation von visuellen Merkmalsrepräsentation (in E) und Online-Handschrift Merkmalsrepräsentation (in Q) in den Topic Raum (vgl. Abbildung 26). Im Normalfall ist für die Anfrage kein Wortbild- und für die Wortbilder keine Online-Handschrift Trajektorien verfügbar. Daher wird die entstehende gemeinsame Repräsentation jeweils nur zur Hälfte berechnet.

LSA-Methode beschrieben wurde (siehe Abschnitt 6.2), eignet sich auch dieses Verfahren für die Umsetzung von Anfragen mit Online-Handschrift Trajektorien, da die wesentlichen Komponenten des Verfahrens für diese neue Art von Word Spotting adaptiert werden können.

Bei der Embedded Attributes Methode für QbS werden binäre PHOC-Vektoren verwendet (vgl. Abbildung 22), welche die Attribute für eine Zeichenkette definieren. Ein Attribut macht dabei eine Aussage über das Vorkommen eines Buchstabens in einer Region des repräsentierten Wortes. Um diese Attribute für ein Wortbild bestimmen zu können, wird für jedes PHOC-Attribut eine SVM trainiert. Die dafür verwendete Beispielmenge von Wortbildern, kann anhand ihrer Annotationen in eine positive und eine negative Teilmenge aufgeteilt werden. Somit lernt die SVM, anhand der visuellen Merkmalsausprägungen zu entscheiden, ob für ein neues Wortbild das entsprechende Attribut vorhanden ist. Dazu wird die Merkmalsrepräsentation des Wortbildes mit der SVM ausgewertet. Der SVM-Score, der dem Abstand der Merkmalsrepräsentation von der Hyperebene entspricht, gibt Aufschluss darüber, ob das Attribut für dieses Wortbild vorhanden ist.

Um Anfragen per Online-Handschrift Trajektorien auswerten zu können, wird eine zweite Menge von Attribut-SVMs benötigt, welche

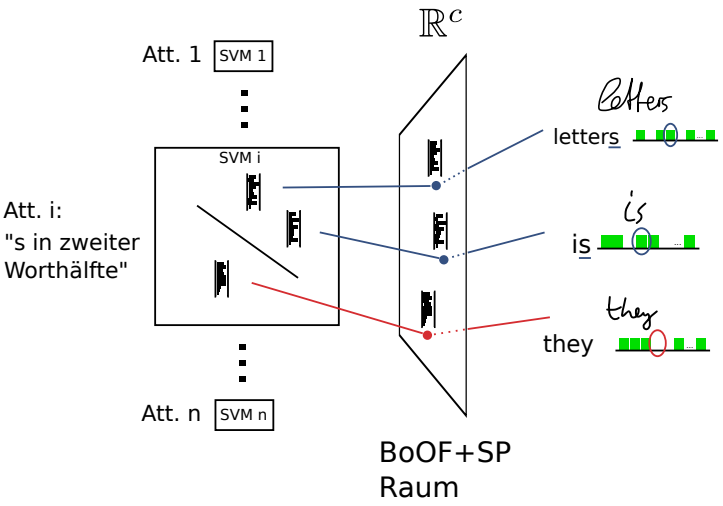


Abbildung 28: Lernen von Attribut-SVMs für Trajektorien (nach [AGFV14a]). Jede PHOC-Dimension ist ein Attribut, welches für Online-Handschrift Trajektorien durch eine SVM bestimmt werden soll. Die Darstellung der PHOC-Vektoren orientiert sich an Abbildung 22. Im Training ist durch den PHOC-Vektor der Annotation jeder Trajektorie bekannt, ob für diese ein Attribut vorhanden ist (verdeutlicht durch blaue Markierung) oder nicht (rote Markierung). Nach dieser Information werden die Trajektorien in zwei Klassen eingeteilt und die entsprechende Attribut-SVM trainiert.

die gleichen Attribute auch für Trajektorien bestimmen kann. Das Training dieser SVMs geschieht analog zum Training für Attribut-SVMs für Wortbilder und ist in Abbildung 28 dargestellt (vgl. Abbildung 23). Dabei werden für alle Trajektorien die BoOF+SP-Merkmalsrepräsentationen berechnet. Durch die im Training vorhandenen Annotationen (bzw. die daraus berechneten PHOC-Vektoren) ist für jede Trajektorie bekannt, welche Attribute vorhanden sind. Nach dieser Information werden die Merkmalsrepräsentationen jeder Trajektorie pro Attribut in eine positive Klasse (Attribut vorhanden) und eine negative Klasse eingeteilt, mit denen die entsprechende Attribut-SVM trainiert wird. Für eine neue Online-Handschrift Trajektorie werden die Attribute bestimmt, indem die BoOF+SP-Merkmalsrepräsentation mit allen Attribut-SVMs ausgewertet wird und die SVM-Scores in einen gemeinsamen Vektor eingetragen werden.

In [AGFV14a] führte bereits der direkte Vergleich der SVM-Scores und PHOC-Vektoren zu guten Ergebnissen für das Query-by-String Word Spotting. Diese Variante wird auch in dieser Arbeit evaluiert, wobei hier die Scores von Wortbild-SVMs und Trajektorien-SVMs direkt verglichen werden. Die zweite Variante für den Vergleich von PHOC-Repräsentation und SVM-Scores wurde in [AGFV14a] über

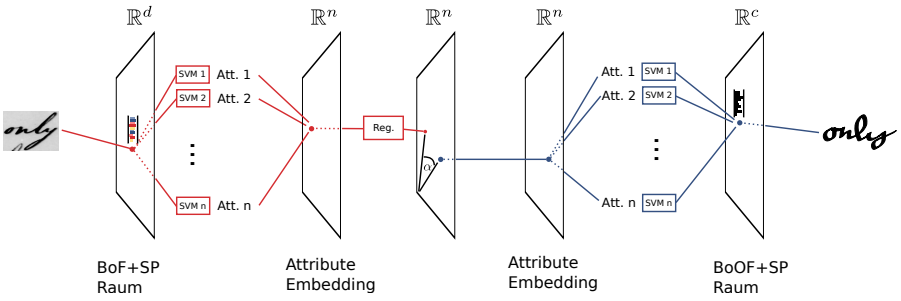


Abbildung 29: Transformation der Repräsentationen von Wortbild und Trajektorie in einen gemeinsamen Unterraum (nach [AGFV14a]). Für das Wortbild und die Trajektorie wird über die BoF (bzw. BoOF), Spatial Pyramid und die Attribut-SVMs eine Attribut Embedding berechnet. Der Abstand beider Repräsentationen wird durch Regression über eine gelernte Transformation minimiert.

das Kalibrieren der Scores mit Platt's Scaling (vgl. Abschnitt 2.3) durchgeführt. Dies führte zu besseren Ergebnissen, da PHOC-Vektoren aus dem Wertebereich $\{0,1\}^{604}$ stammen, während die SVM-Score-Vektoren aus dem Wertebereich $\mathbb{R}^{604}$ stammen. Nach der Kalibrierung liegen die SVM-Scores im Wertebereich $[0,1]^{604}$. Beim QbO Word Spotting werden durch dieses Verfahren die Wertebereiche der Scores, welche sowohl für Wortbild-SVMs, als auch Trajektorien-SVMs unbeschränkt sind, angeglichen.

Eine andere Möglichkeit zur Verbesserung der Retrieval Ergebnisse wurde in [AGFV14a] durch lineare Regression und Common Subspace Regression erreicht. Beim QbS Verfahren wird dafür per Formulierung eines Optimierungsproblems der Vektorabstand zwischen beiden Merkmalsrepräsentationen des gleichen Wortes reduziert, indem für eine oder beide Repräsentationen eine Transformation gelernt wird (siehe dazu Abschnitt 5.3.4). Hierbei wird der Wertebereich der SVM-Scores unter Beachtung von Korrelationen zwischen einzelnen Attributen angepasst [AGFV14a]. Das Embedded Attributes Verfahren mit linearer Regression ist für das QbO Verfahren in Abbildung 29 dargestellt. Bei der linearen Regression wird hierbei eine Transformation für visuelle Attribut-SVM-Scores berechnet.

6.4 DISKUSSION

In diesem Kapitel wurden zwei Verfahren für die neue Idee des Word Spotting mit Online-Handschrift Anfragen vorgestellt. Beide Verfahren berechnen auf unterschiedliche Weise einen Vektorraum, in dem visuelle und Online-Handschrift Merkmalsrepräsentationen

verglichen werden. Insgesamt wurden fünf Varianten der Verfahren erläutert:

- Transformation in Topic Raum (LSA),
- Lernen von Attribut SVMs, direkter Vergleich der SVM-Scores,
- Lernen von Attribut SVMs, Vergleich kalibrierter SVM-Scores (Platt's Scaling),
- Lernen von Attribut SVMs, Lernen einer einseitigen Transformation durch lineare Regression,
- Lernen von Attribut SVMs, Lernen einer Transformation je Domäne durch Common Subspace Regression.

Beim Betrachten der Vorgehensweisen dieser Varianten fällt auf, dass für das Berechnen von LSA und den beiden Regressions-Varianten jeweils Beispieldaten in bestimmter Form vorliegen müssen. Da visuelle und Online-Handschrift Merkmalsrepräsentation für ein repräsentiertes Wort jeweils gemeinsam verwendet werden, müssen die Beispieldaten Wörtern beinhalten, für die jeweils sowohl ein Wortbild, als auch die passende Trajektorie vorliegen. Dies ist in der Praxis besonders dann hinderlich, wenn durch ein Verfahren ein neues Dokument durchsucht werden soll, dabei aber auf schon vorhandene Online-Handschrift Trajektorien zurückgegriffen werden soll.

Eine Möglichkeit, dies für nicht zueinander passende Datensätze zu lösen, ist das Finden von Paaren von Wortbildern und Trajektorien gleicher Wörter. Das Problem dabei besteht in den Wortbildern oder Trajektorien, für die kein Gegenstück gefunden werden kann. In Abhängigkeit der Anzahl der Trainingsdaten und der repräsentierten Wörter in jeder Domäne, kann dies den Verlust eines großen Teils dieser Trainingsdaten bedeuten.

Die beiden QbO-Varianten mit direktem Vergleich von unkalibrierten oder durch Platt's Scaling kalibrierten SVM-Scores sind von diesem Problem nicht betroffen. Für jede Domäne werden hier zwar auch unabhängig SVMs trainiert, welche die Attribute für genau diese Domäne beschreiben. Der nachfolgende Optimierungsschritt, welcher den Abstand beider Repräsentationen anhand der Trainingsdaten minimiert, fällt allerdings weg. Dadurch können sich die repräsentierten Wörter in den Trainingsdaten für das Lernen der beiden SVM-Mengen beliebig unterscheiden. In Kapitel 7 werden alle Varianten für das Word Spotting mit Anfragen eines einzelnen Schreibers evaluiert. Die beiden zuletzt genannten Varianten werden zudem verwendet, um Word Spotting Modelle mit Online-Handschrift Anfragen zu evaluieren, welche schreiberunabhängige Anfragen verarbeiten können.

7 EVALUATION

In diesem Kapitel werden die Experimente vorgestellt, welche der Auswertung der zuvor beschriebenen Word Spotting Methoden dienen (vgl. Kapitel 5, Kapitel 6). Zunächst werden in Abschnitt 7.1 die drei verwendeten Datensätze beschrieben. Nach der Erläuterung des allgemeinen Vorgehens bei der Evaluierung in Abschnitt 7.2, werden die Auswertungen der Experimente der evaluierten Baseline-Verfahren für die Query-by-Example (Abschnitt 7.3) und Query-by-String Verfahren (Abschnitt 7.4) präsentiert. Diese dienen der Prüfung der Korrektheit der erstellten Implementierungen und, im Fall der Query-by-String Verfahren mit Attribute Embedding, der Auswertung einer alternativen visuellen Merkmalsrepräsentation im Vergleich zu dem in [AGFV14a] beschriebenen Verfahren. Anschliessend werden die vorgeschlagenen Verfahren für Query-by-Online-Trajectory in Abschnitt 7.5 evaluiert und die wichtigsten Erkenntnisse in Abschnitt 7.6 zusammengefasst.

7.1 DATENSÄTZE

In diesem Abschnitt werden die Datensätze vorgestellt, welche zum Training und zur Auswertung der vorgestellten Word Spotting Verfahren genutzt werden. Zwei Arten von Daten werden dabei verwendet: Wortbilder und Online-Handschrift Trajektorien. Für die Evaluation ist es zudem wichtig, dass sowohl für Wortbilder, als auch für Online-Handschrift Trajektorien Transkriptionen vorliegen. Dies ist durch die in diesem Abschnitte beschriebenen Datensätze erfüllt. Die weiteren Anforderungen an die verwendeten Daten unterscheiden sich in Abhängigkeit des evaluierten Word Spotting Verfahren. Besondere Voraussetzungen werden im Folgenden in den jeweiligen Abschnitten gesondert erläutert.

Die verwendeten Wortbilder stammen aus dem George Washington Datensatz [GW] (Abschnitt 7.1.1). Sie werden in jedem Experiment sowohl im Modelltraining (sofern vorhanden) als auch für die Evaluierung, d.h. die Auswertung von Anfragen, eingesetzt. Online-Handschrift Trajektorien werden in den Experimenten zu den in dieser Arbeit vorgestellten QbO Verfahren verwendet. Dabei wird zwischen schreiberabhängigen- und schreiberunabhängigen Experimenten unterschieden. In schreiberabhängigen Experimenten werden im Modelltraining nur Online-Handschrift Trajektorien eines einzigen Schreibers verwendet. Das berechnete Word Spotting Modell kann somit bei Anfragen nicht zwischen unterschiedlichen Handschriftstilen

unterscheiden. Diese Experimente dienen der Parameteroptimierung der Verfahren und dem Vergleich zu verwandten Arbeiten. Zu diesem Zweck wurde eine Online-Handschrift Version des George Washington Datensatzes erstellt, deren Eigenschaften in (Abschnitt 7.1.2) erläutert werden. Bei schreiberunabhängigen Experimenten (auch: Multischreiber Experimente) muss das trainierte Word Spotting Modell in der Lage sein, Anfragen eines nicht in den Trainingsbeispielen enthaltenen Handschriftstils verarbeiten zu können. Um dies zu erreichen, wird das Verfahren im Training mit Online-Handschrift Trajektorien von mehreren Schreibern trainiert. Dazu werden Online-Handschrift Trajektorien aus der UNIPEN-Datenbank [Uni] (Abschnitt 7.1.3) verwendet. Während Beispieldaten der UNIPEN-Datenbank ausschließlich im Modelltraining verwendet werden, kommen bei der Evaluierung ausschließlich die oben erwähnten Trajektorien des George Washington Online Datensatzes zum Einsatz.

7.1.1 *George Washington Datensatz*

Der George Washington Datensatz [GW] besteht aus Graustufen-Bildern von handschriftlichen Briefen, welche von George Washington und wenigen Mitarbeitern verfasst wurden [RM07, FKFB12]. Da sich das Schriftbild aller Schreiber sehr ähnelt und somit eine niedrige Variabilität der dargestellten Handschrift vorliegt, wird dieser Datensatz oftmals in Einzel-Schreiber-Szenarios verwendet [AGFV14a]. Die Ground Truth des Datensatzes beinhaltet 20 Seiten mit 4860 segmentierten Wortbildern. Jedes Wortbild ist mit der Transkription des abgebildeten Wortes annotiert [RM07]. Abbildung 7.30(a) zeigt eine Auswahl einiger enthaltener Wortbilder. Der Datensatz wird im Folgenden auch als GW-Datensatz bezeichnet.

7.1.2 *George Washington Online Datensatz*

Wie in Abschnitt 6.4 erläutert wurde, eignen sich nur zwei der ingesamte fünf evaluierten QbO-Varianten für Datensätze, in denen Wortbilder und Trajektorien einen unterschiedlichen Wortschatz an dargestellten Wörtern beinhalten. Zur Auswertung der Word Spotting Verfahren mit Online-Handschrift Anfragen bietet es sich daher an, zunächst einen Datensatz von Wortbildern und Trajektorien zu verwenden, welche auf dem gleichen Wortschatz basieren. Zu diesem Zweck wurde eine handschriftliche Version des George Washington Datensatzes angelegt, welche im Folgenden als GWO-Datensatz bezeichnet wird. Die Trajektorien wurden dabei auf einem Google Nexus 7 Tablet (Version 2012) mit einer speziell dafür angefertigten Applikation erfasst. Für alle Trajektorien wurde der gleiche Handschriftstil verwendet. Zu jedem der 4860 Wortbilder ist eine entsprechende Online-Handschrift Trajektorie in diesem

Datensatz vorhanden. Dieser Umstand wird nachfolgenden auch als *Paarung* eines Wortbildes und einer Trajektorie beschrieben. Der GWO-Datensatz wird zunächst zur Parameteroptimierung und Evaluierung in schreiberabhängigen Experimenten der fünf QbO-Varianten verwendet. Bei den Multischreiber-Experimenten dient er anschließend als Testset für die Evaluierung schreiberunabhängiger Word Spotting Modelle (Abschnitt 7.5.2). Einige gerenderte Beispiel-Trajektorien sind in Abbildung 7.30(b) dargestellt.

7.1.3 *UNIPEN*

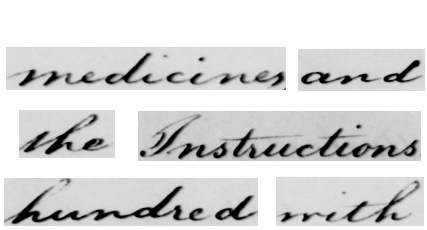
Der Unipen-Datensatz [Uni] ist eine von der Unipen Foundation veröffentlichte Sammlung von Online-Handschrift Daten, die weit über 100 000 Wortvorkommen einer Vielzahl von Schreibern enthält. In dieser Arbeit wird nur ein kleiner Ausschnitt des Datensatzes verwendet, welcher 27112 in Wörter segmentierte Trajektorien enthält (Kennzeichnung „`sta0`“). Diese Wörter wurden von 62 Schreibern auf einem Wacom 420-L Grafiktablet geschrieben. Eine Auswahl von verschiedenen gerenderten Trajektorien dieser kleineren Teilmenge des Unipen-Datensatzes ist in Abbildung 7.30(c) dargestellt. Pro Schreiber ergibt sich eine Menge von etwa 430 Beispieltrajektorien. Im Folgenden werden die Unipen-Trajektorien aussschliesslich für die Evaluierung schreiberunabhängiger Word Spotting Modelle verwendet, da der Wortschatz der enthaltenen Wörter nur eine kleine Menge an Überschneidungen mit dem Wortschatz des George Washington Datensatzes bildet. Während der GWO-Datensatz (Abschnitt 7.1.2) zu jedem Wortbild eine passende Online-Handschrift Trajektorie enthält, ist dies im UNIPEN-Datensatz nicht der Fall.

7.2 EVALUATIONS PROTOKOLL

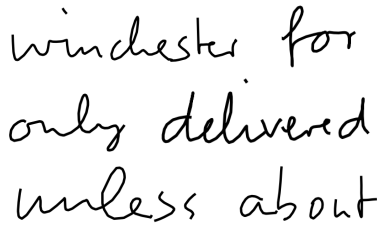
Word Spotting Verfahren lösen ein Retrieval-Problem, welches aus der Suche nach relevanten Wortbildern in einem Archiv aus Dokumenten zu einer gegebenen Anfrage besteht. Bei den nachfolgenden Erläuterungen wird davon ausgegangen, dass durch ein Word Spotting Verfahren eine begrenzte Anzahl an (nach Relevanz sortierten) Wortbildern für eine Anfrage zurückgegeben wird. Dies wird als Retrieval-Liste bezeichnet. Alle Wortbilder der Retrieval-Liste, die tatsächlich relevant sind, d.h. das angefragte Wort darstellen, werden Treffer genannt. Zwei typische Gütemaße, um Retrieval Methoden zu vergleichen, sind Precision und Recall ([BYRN99], Kap. 4)

$$\text{Precision} = \frac{\text{Anzahl Treffer}}{\text{Länge der Retrieval-Liste}} \tag{19}$$

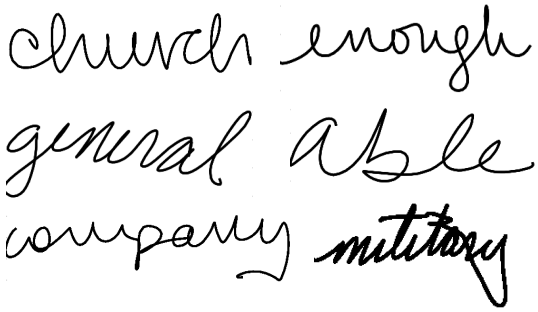
$$\text{Recall} = \frac{\text{Anzahl Treffer}}{\text{Anzahl tatsächlich relevanter Wortbilder}}. \tag{20}$$



(a) George Washington Datensatz



(b) George Washington Online-Handschrift Datensatz



(c) Unipen Datensatz

Abbildung 30: Einige Beispiele für Wortbilder (a) und Online-Handschrift Trajektorien (b, c) aus den verwendeten Datensätzen (Bilder aus dem GWO-Datensatz und [GW, Uni]).

Die Precision beschreibt das Verhältnis von Treffern zu der Anzahl der Suchergebnisse in der Retrieval-Liste. Wenn beispielsweise die Hälfte der Ergebnisse das angefragte Wort darstellen, während die andere Hälfte der Ergebnisse fälschlicherweise durch das Word Spotting Verfahren als relevant eingestuft wurde, ist die Precision 50%. Der Recall ist das Verhältnis der Treffer zu allen tatsächlich relevanten Wortbildern in der durchsuchten Menge von Wortbildern. Dies entspricht allen Wortbildern, die tatsächlich das angefragte Wort abbilden. In der folgenden Evaluierung werden jeweils alle Wortbilder durch das jeweilige Word Spotting Verfahren nach Relevanz sortiert, d.h. die Länge der Retrieval-Liste entspricht der Anzahl aller Wortbilder. Daher ist der Recall immer 100% und die Precision beschreibt den Anteil der Treffer im gesamten Datensatz. Beide Gütemaße reichen für die Bewertung an dieser Stelle somit nicht aus. In der Word Spotting Literatur wird oftmals die Precision-Recall-Kurve (PR-Kurve) eines Retrieval Modells zur Beurteilung von dessen Güte angegeben (z.B. [LOLE09, FKFB12]). Da der ausschlaggebende Wert für die Beurteilung der Erkennungsleistung eines Word Spotting Verfahrens die Fläche unter der PR-Kurve ist, werden im weiteren Verlauf der vorliegenden Arbeit keine PR-Kurven angegeben. Stattdessen wird die ebenfalls beliebte *Mean Average Precision* (mAP) als Gütemaß verwendet, welche ein Maß für

die Fläche unter der PR-Kurve ist [AGFV14b]. Die *Average Precision* (AP) ist definiert durch [LRF+12]

$$\text{AP} = \frac{\sum_{k=1}^{n} \text{Precision}(k) \cdot \text{rel}(k)}{\text{Anzahl tatsächlich relevanter Wortbilder}}. \tag{21}$$

Dabei ist n die Länge der Retrieval-Liste und $\text{rel}(k)$ ist eine binäre Funktion, welche angibt, ob das Wortbild, welches an k-ter Stelle der Retrieval-Liste steht, ein Treffer ist. Die mAP ist die durchschnittliche AP über alle Queries einer Evaluierung. Während durch Precision und Recall jeweils das Verhältnis von Treffern zur Länge der Retrieval-Liste, bzw. zu allen relevanten Wortbildern dargestellt wird, bietet die Average Precision Informationen zur Reihenfolge der zurückgelieferten Ergebnisse, da sie die Precision an jeder Stelle der Retrieval-Liste aufsummiert. Je weiter hinten die Treffer in der Retrieval-Liste positioniert sind, desto geringer ist die Precision an jeder Position eines Treffers. Die Average Precision ist genau dann 100%, wenn alle k Ergebnisse, welche das angefragte Wort tatsächlich darstellen, an den ersten k Positionen der Retrieval-Liste stehen.

Für die Evaluierung der Query-by-Example und Query-by-String Verfahren (vgl. Abschnitt 5.3.3 und Abschnitt 5.3.4) werden die 4860 Wortbilder des GW-Datensatzes (Abschnitt 7.1.1) verwendet. Das Vorgehen orientiert sich an dem Benchmark aus [ARTL13] bzw. [AGFV14a]. Dabei wird eine Kreuzvalidierung durchgeführt, wobei die 20 Seiten des GW-Datensatzes in vier Partitionen geteilt werden. Das jeweilige Word Spotting Modell wird in vier Durchläufen mit je drei Partitionen (Trainingsset) trainiert und auf der letzten Partition (Testset) getestet. Für die Auswertung der QbE Verfahren wird jedes Wortbild des Testsets einmal als Anfrage verwendet um in den verbleibenden Wortbildern des Testsets nach visuell ähnlichen Vorkommen des dargestellten Wortes zu suchen. Für die QbS Verfahren wird jedes Wort aus den Transkriptionen der Wortbilder des Testsets einmal angefragt. Doppelt auftretende Wörter werden nur einmal angefragt. Die mAP-Werte werden jeweils über alle vier Durchläufe gemittelt.

Eine ähnliche Vorgehensweise wird für die Evaluierung der schreiberabhängigen QbO-Verfahren (Abschnitt 7.5.1) eingesetzt, wobei zusätzlich die Trajektorien des GWO-Datensatzes (Abschnitt 7.1.2) verwendet und analog in vier Partitionen geteilt werden. Dies bedeutet, dass sowohl im Trainings-, als auch Testset zu jedem Wortbild genau eine passende Online-Handschrift Trajektorie vorliegt. Beim Test werden alle Online-Handschrift Trajektorien des Testsets einmal als Anfrage verwendet um in den Wortbildern des Testsets zu suchen. Die schreiberabhängigen Experimente dienen zudem der Parameteroptimierung der eingesetzten Verfahren. Diese wird über Gridsuche durchgeführt, bei der aus einer zuvor gewählten Menge von Parameterbelegungen die beste Kombination bestimmt wird.

Bei den Multischreiber-Experimenten in Abschnitt 7.5.2 wird die gleiche, zuvor beschriebene Datenteilung der GW- und GWO-Datensätze zur Kreuzvalidierung vorgenommen. Zusätzlich wird der UNIPEN-Datensatz verwendet. Für einen Durchgang der Kreuzvalidierung ergeben sich Trainings-, und Testset wie folgt.

- **Trainingsset**. Drei Partitionen des GW-Datensatzes werden verwendet. Zudem werden alle, für das jeweilige Experiment ausgewählte, UNIPEN-Trajektorien benutzt (für den genauen Aufbau dieser Experimente, siehe Abschnitt 7.5.2).
- **Testset**. Alle GWO-Trajektorien der verbleibenden vierten Partition werden einmal als Anfrage verwendet, um in den Wortbildern dieser letzten Partition nach Treffern zu suchen.

Durch diese Aufteilung der Daten werden im Trainingsschritt insbesondere keine Beispieldaten mit dem Handschriftstil des GWO-Datensatzes verwendet. Dadurch werden handschriftliche Anfragen in einem, dem trainierten Word Spotting Modells, unbekannten Handschriftstil gestellt.

7.3 QUERY-BY-EXAMPLE

Der folgende Abschnitt präsentiert die Ergebnisse der evaluierten Baseline-Verfahren für QbE Experimente (siehe Abschnitt 5.3.3 und Abschnitt 5.3.4). Die Ergebnisse geben einen Hinweis darauf, wie bestimmte Komponenten der Verfahren zusammenarbeiten. Besonders im Fall der Embedded Attributes-Methode ist eine Evaluierung der Verwendung der Bag-of-Features und Spatial Pyramid Merkmalsrepräsentation anstelle der in [AGFV14a] vorgeschlagenen FV+K Merkmalsrepräsentation (vgl. Abschnitt 5.3.4) von Interesse. Folgende Varianten wurden evaluiert:

- **BoF+SP**. Direkter Vergleich der Bag-of-Features und Spatial Pyramid Merkmalsrepräsentationen,
- **Att.** Direkter Vergleich der Attribute,
- **Att.+Platt's.** Kalibrierung der Attribute mit Platt's Scaling,
- **Att.+Reg.** Lernen eines Unterraums mit linearer Regression,
- **Att.+CSR.** Lernen eines Unterraums mit Common Subspace Regression.

Bei den Query-by-Example Experimenten wird jedes Wortbild im Testset angefragt. Dabei wird das angefragte Wortbild für diese Anfrage aus dem Testset entfernt, da dieses den Abstand Null von der Anfrage hat und so stets an erster Stelle der Suchergebnisse stehen würde. Wörter, die nur einmal im Testset vorkommen,

	2x1 / 1x1	3x2	Vorgabe
BoF + SP	61.23	66.69	62.72
Att.	**90.99**	**90.27**	89.85
Att. + Platt's	87.06	87.95	**93.04**
Att. + Reg.	86.36	88.15	90.54
Att. + CSR	89.68	90.19	92.46

Tabelle 1: Query-by-Example Baseline Implementierungen. Angaben jeweils mAP in %. Die Vorgaben stammen aus [AGFV14a] und verwenden die FV+K Merkmalsrepräsentation für Wortbilder.

werden dementsprechend nicht angefragt. Die vier Embedded Attributes Varianten werden per Kreuzvalidierung evaluiert. Da beim direkten Vergleich von BoF+SP Vektoren kein Training eines Modells durchgeführt wird, besteht hier der Test in der einmaligen Anfrage aller Wortbilder des GW-Datensatzes – eine Kreuzvalidierung ist nötig. Die Parameterwahl der BoF+SP Konfiguration orientiert sich an Werten aus der Literatur [RATL11, SRF]. Die Ergebnisse sind in Tabelle 1 dargestellt und werden in Abhängigkeit von zwei getesteten Spatial Pyramid Konfigurationen für die visuelle Merkmalsrepräsentation angegeben.

Die erste Tabellenzeile (siehe Tabelle 1) vergleicht die Verwendung der BoF+SP Merkmalsrepräsentation mit den in [AGFV14a] verwendeten FV+K Merkmalsrepräsentation. Es wird deutlich, dass bereits mit kleiner Auflösung der Spatial Pyramid die Vorgabewerte erreicht und sogar überschritten werden. Für die vier Varianten des Embedded Attributes Verfahren zeigt sich, dass BoF+SP ein geeigneter Ersatz für FV+K ist. Dies bestätigt eine Aussage in [AGFV14a], wonach das Embedded Attributes Verfahren nicht abhängig von der konkret gewählten visuellen Merkmalsrepräsentation ist. Beim Platt's Scaling ist die Abweichung des Baseline-Verfahrens von der Vorlage mit $\sim$ 6% am größten. Neben der, von der Vorlage verschiedenen, Merkmalsrepräsentation, könnte dies an der Verwendung eines Verfahrens mit Maximum Likelihood Schätzung [LLW07] zum Platt's Scaling, anstatt eines Extremwerttheorie-Verfahrens [SKBB12, AGFV14a] liegen. Die beiden Regressions-Varianten weichen in der besten Konfiguration („3x2" Spatial Pyramid) um etwa 2% von der Vorlage ab. Die beste Query-by-Example Methode der evaluierten Baseline-Verfahren ist somit der direkte Vergleich der Attribut-SVM-Scores.

7.4 QUERY-BY-STRING

Die Query-by-String Verfahren aus [ARTL13] und [AGFV14a] dienen als Vorlage für die Umsetzung von QbO-Verfahren. Die Ergebnisse

der beiden Anfrage-Varianten QbS und QbO sind nicht direkt miteinander vergleichbar, da Anfragen per Online-Handschrift eine weitaus höhere Variabilität aufweisen, als Anfragen per Zeichenkette. Die textuelle Merkmalsrepräsentation einer Zeichenkette, sowohl basierend auf n-Grammen [ARTL13], als auch PHOC-Vektoren [AGFV14a], ist nicht abhängig vom Schreiber, was bei Online-Handschrift Merkmalsrepräsentation der Fall ist. Die QbS-Ergebnisse dienen aufgrund der niedrigeren Variabilität allerdings als Orientierung bzw. obere Schranke für die zu erwartenden Werte bei QbO.

Im Folgenden werden die Evaluierungsergebnisse der Query-by-String Baseline-Verfahren

- **LSA**. Latent Semantic Analysis Methode,
- **Att.** Direkter Vergleich der Attribute,
- **Att.+Platt's.** Kalibrierung der Attribute mit Platt's Scaling,
- **Att.+Reg.** Lernen eines Unterraums mit linearer Regression,
- **Att.+CSR.** Lernen eines Unterraums mit Common Subspace Regression

beschrieben (siehe Abschnitt 5.3.3 und Abschnitt 5.3.4). Bei der Anwendung der QbS Verfahren ist sowohl für die LSA-, als auch die Embedded Attributes Methode ein Trainingsschritt nötig, welcher einen Vektorraum der verschiedenen Merkmalsrepräsentationen berechnet. Dieses Training findet bei der Kreuzvalidierung auf drei der vier Partitionen der Daten statt (siehe Abschnitt 7.2). Die Baseline-Verfahren werden auf der jeweils verbleibenden vierten Partition der Daten getestet. Es werden alle Wörter des Testsets, ermittelt anhand der Transkriptionen der Wortbilder, angefragt. Bei doppelten Vorkommen eines Wortes wird dieses nur einmal angefragt.

Die Ergebnisse der Auswertung der LSA-Methode sind in Tabelle 2 dargestellt, jeweils in Abhängigkeit der verwendeten Spatial Pyramid Konfiguration und der Anzahl der verwendeten Topics. Die Vorgaben stammen aus [ARTL13], wobei alle Werte, ausser für 2048 Topics, aus einer dort präsentierten Grafik abgelesen wurden. Die Spatial Pyramid Konfiguration der Vorlage ist „9x2/3x2". Es fällt auf, dass die implementierte Baseline für alle dargestellten Parameterkonfigurationen besser, als die Vorgabe ist. Die genauen Unterschiede zwischen Baseline- und Referenzimplementierung konnten dabei nicht ermittelt werden. Der einzig bekannte Unterschied im Vergleich zur Vorlage besteht in der Berechnung der SIFT-Deskriptoren. In [ARTL13] werden SIFT-Deskriptoren in drei Größen verwendet und zudem Deskriptoren in konstrastarmen Regionen der Dokumentbilder verworfen. Zudem wird für die Quantisierung anhand der Visual Words

	3x2/2x1	9x2/3x2	Vorgabe
LSA (64 Topics)	50.44	44.14	~ 31.00
LSA (128 Topics)	62.93	56.25	~ 40.00
LSA (256 Topics)	70.75	64.24	~ 45.00
LSA (512 Topics)	71.95	67.21	~ 47.50
LSA (1024 Topics)	**73.00**	**67.84**	~ 49.00
LSA (2048 Topics)	71.61	66.99	**56.65**
LSA (4096 Topics)	68.52	65.28	-

Tabelle 2: Query-by-String LSA-Baseline. Angaben in mAP in % über alle Wörter. Vorgaben aus [ARTL13], Werte für Topics (ausser 2048) aus Grafik entnommen.

	3x2	3x2/2x1	Vorgabe
Att.	68.41	71.25	67.64
Att. + Platt's	83.31	85.28	**91.29**
Att. + Reg.	87.14	88.92	87.02
Att. + CSR	**87.39**	**89.16**	90.81

Tabelle 3: Query-by-String Embedded Attributes Baseline Implementierungen. Angaben jeweils mAP in % über alle Testwörter. Vorgaben aus [AGFV14a]. Die Vorgaben verwenden eine FV+K Merkmalsrepräsentation, anstelle von BoF+SP.

Locality-constrained Linear Coding verwendet (LLC, siehe auch Abschnitt 4.2.2). In dieser Arbeit wird nur eine Deskriptor-Größe verwendet und die Quantisierung anhand der euklidischen Distanz zum nächsten Nachbarn durchgeführt. Die beste Parameterkonfiguration der Vorlage ist durch 2048 Topics und die oben beschriebene Spatial Pyramid Konfiguration mit einer mAP von 56.65% gegeben. Das beste Ergebnis der Baseline-Implementierung ist 73.00% und verwendet eine „3x2/2x1“ Spatial Pyramid Konfiguration und 1024 Topics.

In Tabelle 3 sind die Ergebnisse der Evaluierung der vier Varianten der Attribute Embedding QbS-Methode aufgeführt. Die Ergebnisse werden in Abhängigkeit von zwei ausgewählten Spatial Pyramid Konfigurationen präsentiert. Eine Erhöhung oder Reduzierung der gezeigten Auflösungen, d.h. die Verwendung von mehr oder weniger Regionen pro Level, zeigte keine Verbesserung dieser Ergebnisse. Von den vier Varianten der Baseline-Implementierung ergibt die Common Subspace Regression (CSR) mit einer „3x2/2x1“ Spatial Pyramid das beste Ergebnis von 89.16%. Der direkte Vergleich der Attribute (Att.) und die lineare Regression (Att.+Reg.) erreichen bessere mAP-Werte, als die Vorgabe. Die größte Abweichung ist, wie bereits die QbE-Experimente gezeigt haben, die Platt's Scaling Variante. Für die bessere Vergleichbarkeit der vorgestellten Verfahren aus [ARTL13, AGFV14a], wurde in dieser Arbeit einheitlich

die BoF+SP Merkmalsrepräsentation verwendet. Eine Verwendung der FV+K Merkmalsrepräsentation könnte die Erkennungsraten weiter verbessern, der Fokus der vorliegenden Arbeit liegt jedoch nicht auf der Evaluierung visueller Merkmalsrepräsentationen. Bemerkenswerterweise sind die vorgestellten Ergebnisse jedoch bereits, trotz unterschiedlicher visueller Merkmalsrepräsentation, in der gleichen Größenordnung, wie in der Vorlage.

7.5 QUERY-BY-ONLINE-TRAJECTORY

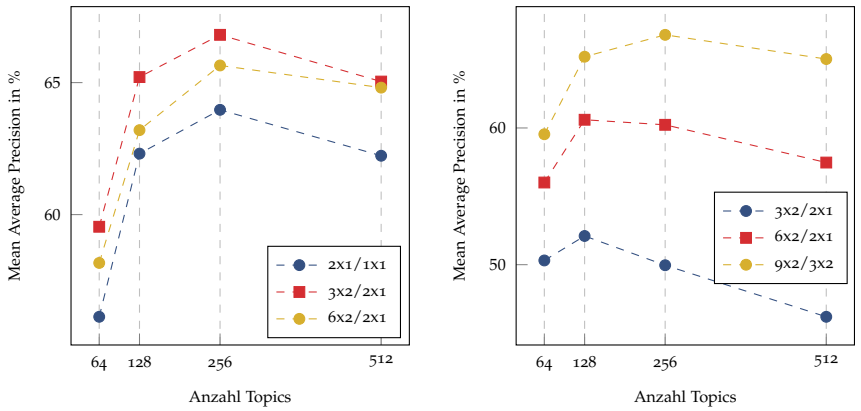
In diesem Abschnitt werden die Evaluierungsergebnisse der in dieser Arbeit zum ersten Mal vorgestellten Word Spotting Verfahren mit Online-Handschrift Anfragen präsentiert, welche in Kapitel 6 beschrieben wurden. Folgende Varianten der vorgestellten Verfahren werden evaluiert:

- **LSA**. Latent Semantic Analysis Methode,
- **Att.** Direkter Vergleich der Attribute,
- **Att.+Platt's.** Kalibrierung der Attribute mit Platt's Scaling,
- **Att.+Reg.** Lernen eines Unterraums mit linearer Regression,
- **Att.+CSR.** Lernen eines Unterraums mit Common Subspace Regression.

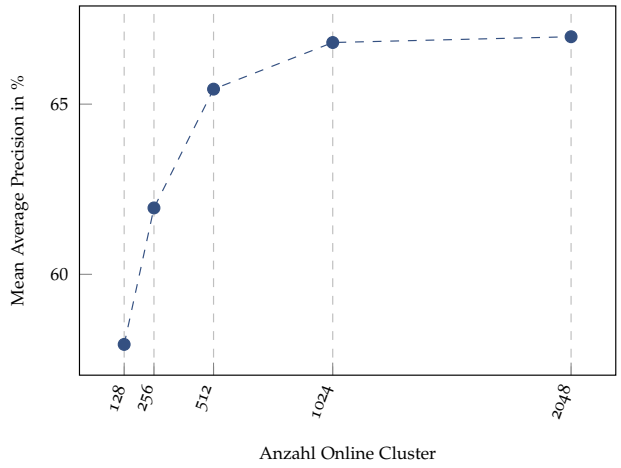
Die Evaluation wurde dabei in zwei Bereiche unterteilt. In Abschnitt 7.5.1 wird die schreiberabhängige Evaluation beschrieben, bei der im Training und bei der Auswertung der Verfahren die Online-Handschrift von nur einem Schreiber verwendet wird. Abschnitt 7.5.2 zeigt die Evaluierungsergebnisse bei der Verwendung von Online-Handschrift Trajektorien mehrerer Schreiber. Diese Evaluation ist praxisorientierter, da hier Modelle evaluiert werden, welche auf Seite der Anfragen im Idealfall unabhängig von verschiedenen Schriftstilen sind. In der Praxis wird ein Word Spotting Verfahren von mehr als nur einem Anwender verwendet. Daher ist die Unterstützung unterschiedlicher Handschriftstile eine Anforderung für Verfahren, die in der Praxis eingesetzt werden sollen.

7.5.1 *Schreiberabhängiges Modell*

Im Folgenden wird die Evaluierung der in Kapitel 6 beschriebenen Verfahren für den schreiberabhängigen Fall mit dem GWO-Datensatz (siehe Abschnitt 7.1.2) beschrieben. Die Online-Handschrift Trajektorien stammen somit bei der Modellbildung sowie der Auswertung der verschiedenen QbO Varianten von einem einzigen Schreiber. Dadurch wird die große Variabilität der Handschrift, welche normalerweise durch mehrere Schreiber verursacht wird, reduziert. Das berechnete



(a) Visuelle Spatial Pyramid (b) Online-Handschrift Spatial Pyramid



(c) Anzahl Online-Cluster

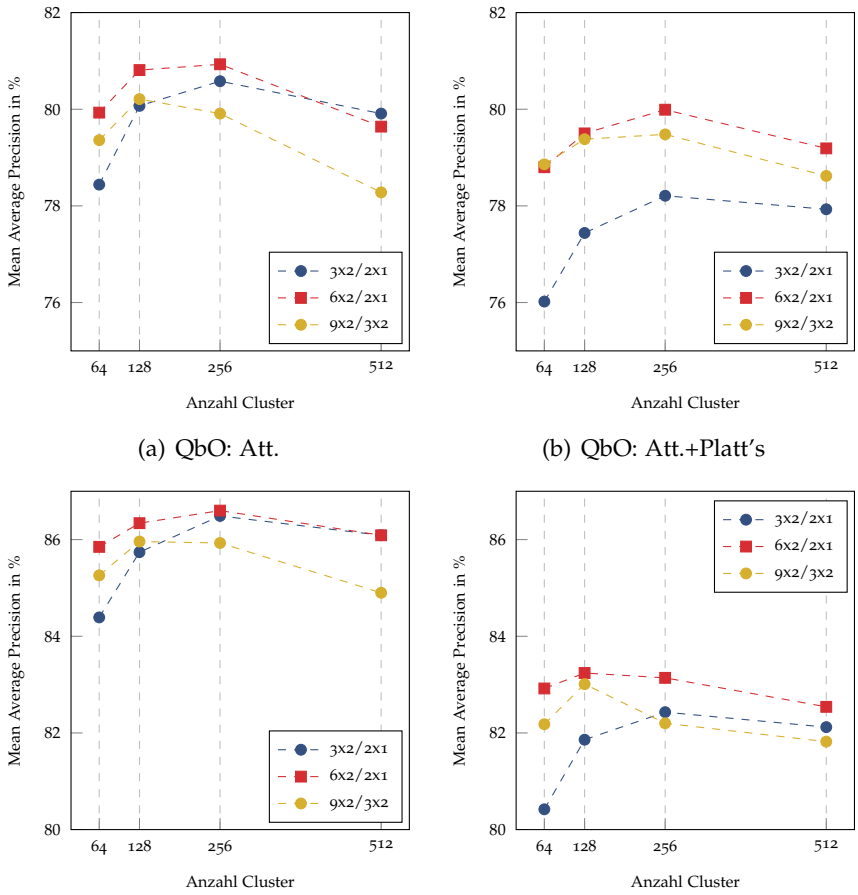
Abbildung 31: QbO-LSA: Optimierung der visuellen Spatial Pyramid Konfiguration (a), der Online-Handschrift Spatial Pyramid Konfiguration (b) und der Anzahl der Online-Handschrift Cluster (c). Die Werte der Messpunkte und x-Achseneinträge sind diskret. Zwischen Messpunkten wurde linear interpoliert.

Modell muss somit bei Anfragen nicht zwischen unterschiedlichen Handschriftstilen unterscheiden.

Wie in Abschnitt 7.4 bereits beschrieben wurde, liegt der Fokus dieser Arbeit nicht auf der Evaluierung von visuellen Merkmalsrepräsentationen. Daher wurden hier Parameter, wie z.B. die Anzahl der Visual Words sowie die Größe und der Abstand zwischen SIFT-Deskriptoren (siehe Abschnitt 4.2.1), nicht optimiert. Die Konfiguration dieser Parameter basiert auf Experimenten aus [ARTL13], in denen diese Parameter bereits für den Einsatz auf dem GW-Datensatz optimiert wurden. Bei der Evaluierung der LSA-Methode für schreiberabhängiges Word Spotting sind somit vier

Parameter von Bedeutung: die Konfiguration der visuellen- und Online-Handschrift Spatial Pyramid, die Anzahl der Topics und die Anzahl der Online-Cluster. Die beste, durch Gridsuche (vgl. Abschnitt 7.2) ermittelte Kombination der vier optimierten Parameter ist die Verwendung von 1024 Online-Handschrift Clustern, 128 Topics und den Konfigurationen „3x2/2x1“ bzw. „9x2/3x2“ für die visuelle- bzw. Online-Handschrift Spatial Pyramid respektive, welche eine mAP von 66.81% erreicht. Es soll kurz aufgezeigt werden, welchen Einfluss die Variation dieser Parameter bei sonst fester, optimaler Belegung der anderen Parameter hat. Für die Anzahl der Online-Handschrift Cluster ist dies in Abbildung 7.31(c) dargestellt. Die Anzahl wurde dabei exponentiell erhöht. Es ist deutlich zu sehen, dass bis zu einer Anzahl von 1024 Clustern die Erhöhung der Anzahl eine große Verbesserung der Erkennungsleistung bewirkt. Die Steigerung von 1024 auf 2048 Cluster verbessert hingegen das Ergebnis nicht wesentlich (66.98% mAP) und erhöht zudem die Berechnungszeit des Topic Raumes enorm, da die Größe der Online-Handschrift Merkmalsrepräsentation direkt abhängig von der Anzahl der Cluster ist. Aus diesem Grund werden im Folgenden 1024 Cluster verwendet. Die stark abfallende Erkennungsleistung bei Verwendung von weniger als 1024 Clustern ist durch eine weniger spezifische Merkmalsrepräsentation zu erklären, die mit der Veringerung der Cluster-Anzahl einhergeht. Bei Reduzierung der Cluster-Anzahl und gleichbleibender räumlicher Verteilung der zu quantisierenden Deskriptoren steigt der Quantisierungsfehler (siehe Abschnitt 2.1), da auch unähnliche Deskriptoren zu gleichen Clustern zugewiesen werden.

Abbildung 7.31(a) zeigt die Auswirkungen der visuellen Spatial Pyramid Konfiguration in Abhängigkeit der Anzahl der Topics. Wie schon in Tabelle 2 zu beobachten war, führt eine naive Erhöhung der visuellen Spatial Pyramid Auflösung nicht automatisch zu besseren Ergebnissen. Vielmehr muss ein geeigneter Kompromiss zwischen Auflösung der visuellen Spatial Pyramid und der Online-Handschrift Spatial Pyramid gefunden werden. Die optimale Auflösung ist hier „3x2/2x1“ bei 256 Topics. In Abbildung 7.31(b) ist die Variation der Online-Handschrift Spatial Pyramid Konfiguration in Abhängigkeit der Anzahl der Topics aufgezeigt. Im Gegensatz zur visuellen Konfiguration sorgt hier eine Spatial Pyramid mit mehr Regionen für eine bessere Erkennungsleistung. Auffällig ist, dass für kleinere Konfigurationen (rote und blaue Kurve) eine geringere Anzahl an Topics optimal ist. Dieses Verhalten ist darauf zurückzuführen, dass die Anzahl der Spatial Pyramid Regionen einen direkten Einfluss auf die Größe der visuellen Merkmalsrepräsentation hat. Bei einer kleineren Repräsentation reichen weniger Topics aus, die entscheidenden Korrelationen zwischen visuellen- und Online-Handschrift Merkmalen zu erfassen. Zudem ist für die Topics ein



(a) QbO: Att.

(b) QbO: Att.+Platt's

(c) QbO: Att.+Reg.

(d) QbO: Att.+CSR

Abbildung 32: QbO-Embedded Attributes: Optimierung der Online Spatial Pyramid Konfiguration und Anzahl der Online Cluster bei fester „3x2/2x1“ visueller Spatial Pyramid. Die Werte der Messpunkte und x-Achseneinträge sind diskret. Zwischen Messpunkten wurde linear interpoliert. Aus Gründen der Übersicht wird in den Grafiken (c) und (d) ein anderer y-Achsen Abschnitt, als in (a) und (b) dargestellt.

ähnliches Verhalten, wie bei der Cluster-Anzahl zu beobachten. Für geringere Größen der Topic-Anzahl sinkt die Erkennungsleistung, da auch hier die Merkmalsrepräsentation zu unspezifisch wird.

Bei den vier Varianten der Embedded Attributes Methode sind die visuelle- und Online-Handschrift Spatial Pyramid Konfiguration sowie die Anzahl der Online-Cluster zu optimieren. Dies wurde erneut durch Gridsuche erreicht. Für die Konfiguration der visuellen Spatial Pyramid wurde sich an den Ergebnissen der QbS-Evaluierung orientiert. Da für alle vier Varianten eine Abweichung von der dort optimalen Wahl „3x2/2x1“ keine nennenswerte Verbesserung erzielte, wird diese Konfiguration im Folgenden auch für die QbO-Evaluierung

Variante	mAP
LSA	66.81
Att.	81.37
Att. + Platt's	79.99
Att. + Reg.	**86.49**
Att. + CSR	83.59

Tabelle 4: Vergleich der jeweils optimalen Konfiguration aller evaluierten QbO Varianten für das schreiberabhängige Word Spotting.

verwendet. In Abbildung 32 sind die Evaluierungsergebnisse aller vier Varianten in Abhängigkeit der Online Spatial Pyramid Konfiguration und der Anzahl der Online Cluster abgebildet. Beim Vergleich der vier Varianten untereinander ist zu beachten, dass in Abbildung 7.32(a) und Abbildung 7.32(b) aus Gründen der Übersicht ein anderer y-Achsen Abschnitt dargestellt ist, als in Abbildung 7.32(c) und Abbildung 7.32(d). Es ist zunächst ersichtlich, dass die „6x2/2x1" Spatial Pyramid für alle Varianten die beste Konfiguration darstellt. Für den direkten Vergleich der Attribute, die Platt's Scaling-Variante und die lineare Regression ist eine Anzahl von 256 Online-Cluster optimal, für die Common Subspace Regression Variante liegt das Optimum bei 128 Clustern. Hier ist das gleiche, schon bei der LSA-Methode beschriebene, Verhalten bei Änderung der Cluster-Anzahl zu beobachten. Eine geringe Anzahl an Clustern führt zur Zuweisung auch unähnlicher Deskriptoren zu einem gleichen Cluster, wodurch die Erkennungsleistung sinkt. Eine zu hohe Anzahl an Clustern führt ebenfalls zu einer verringerten Erkennungsleistung, da hier ähnliche Deskriptoren zu unterschiedlichen Clustern zugewiesen werden. Beide Regressions-Varianten liefern bessere Ergebnisse als der Attribut-Vergleich mit bzw. ohne Platt's Scaling. Die beste der vier Varianten ist die lineare Regression mit 86.49%.

Tabelle 4 zeigt zusammenfassend die Ergebnisse der fünf evaluierten QbO-Varianten anhand ihrer jeweils besten Parameterkombination. Die LSA-Variante erreicht deutlich schlechtere Erkennungsergebnisse, als die Varianten des Attribute Embedding. Zudem ist anzumerken, dass die Berechnungszeit der LSA-Variante aufgrund der Singulärwertzerlegung deutlich über denen der anderen Varianten liegt, und sie somit nur bedingt für einen Einsatz in der Praxis geeignet ist. Für das schreiberabhängige Word Spotting mit Online-Handschrift ist somit die Variante mit linearer Regression am besten geeignet.

7.5.2 *Multischreiber Modell*

In diesem Abschnitt werden die Experimente für Modelle beschrieben, die schreiberunabhängige Anfragen bearbeiten können. Das Ziel dabei ist, im Idealfall, ein Modell zu trainieren, welches Anfragen eines beliebigen Schreibers bzw. eines beliebigen Schriftstils verarbeiten kann. Als Quelle für Online-Handschrift Trajektorien von mehreren Schreibern wird eine Teilmenge des UNIPEN-Datensatzes (siehe Abschnitt 7.1.3) verwendet. Zusätzlich kommen weiterhin, wie in der Evaluierung schreiberabhängiger Modelle in Abschnitt 7.5.1 bereits gesehen, die Trajektorien des GWO-Datensatzes (Abschnitt 7.1.2) zum Einsatz. Zusammen ergibt dies eine Menge von 63 Schreibern.

In Abschnitt 6.4 wurde diskutiert, dass für die LSA- und Regressions-Varianten der vorgestellten QbO-Methoden die Trainingsdaten in Paaren (vgl. Abschnitt 7.1.1) vorliegen müssen, bei denen jeweils ein Wortbild und eine Trajektorie, welche das gleiche Wort repräsentieren, zusammengehören. Da diese Voraussetzung unter Hinzunahme des UNIPEN-Datensatzes nicht mehr erfüllt ist [1], kommen im Folgenden lediglich die Varianten „Att." und „Att.+Platts" zum Einsatz. Die Parameter werden entsprechend der optimalen Kombination aus der schreiberunabhängigen Evaluation (Abschnitt 7.5.1) gewählt. Dies entspricht einer „3x2/2x1" visuellen Spatial Pyramid, einer „6x2/2x1" Online-Handschrift Spatial Pyramid und einer Anzahl von 256 Online-Handschrift Clustern. Aufgrund von Unregelmäßigkeiten bei der Erfassung des Ab- und Aufsetzen des Stiftes im UNIPEN-Datensatz, werden in den folgenden Experimenten Trajektorien bei der Normalisierung nicht von Delayed Strokes befreit (vgl. Abschnitt 3.2.7). Dementsprechend werden auch das Stiftzustand- und Delayed Stroke-Merkmal nicht verwendet (vgl. Abschnitt 3.3.3). Einige informelle Experimente haben gezeigt, dass dies die obige Wahl der optimalen Parameter nicht beeinflusst.

Die Evaluierung der Multischreiber-Modelle wird anhand von vier Experimenten durchgeführt, bei denen die verwendete Auswahl der in Wortbildern und Trajektorien dargestellten Wörter variiert wird. Dies soll den Einfluss auf die Erkennungsleistung der Verfahren demonstrieren, den die Verfügbarkeit von verschiedenen Trainingsdaten hat. Gerade in der Praxis ist es recht unwahrscheinlich, dass zu einem zu durchsuchenden Dokument, aus dem Wortbilder extrahiert werden, auch die passenden Online-Handschrift Trajektorien vorliegen. Die Wortauswahl für jedes Experiment ist in Abbildung 33 dargestellt. Der gemeinsame Wortschatz von GW-Datensatz und UNIPEN-Datensatz beträgt 223 Wörter.

1 Im folgenden Experiment 2 könnten solche Paarungen erstellt werden, was einen zusätzlichen Berechnungsschritt erfordern würde. Allerdings ist die Suche solcher Paarungen nicht trivial, beispielsweise in Fällen mit ungleicher Anzahl an Wortbildern und Trajektorien zum selben dargestellten Wort.

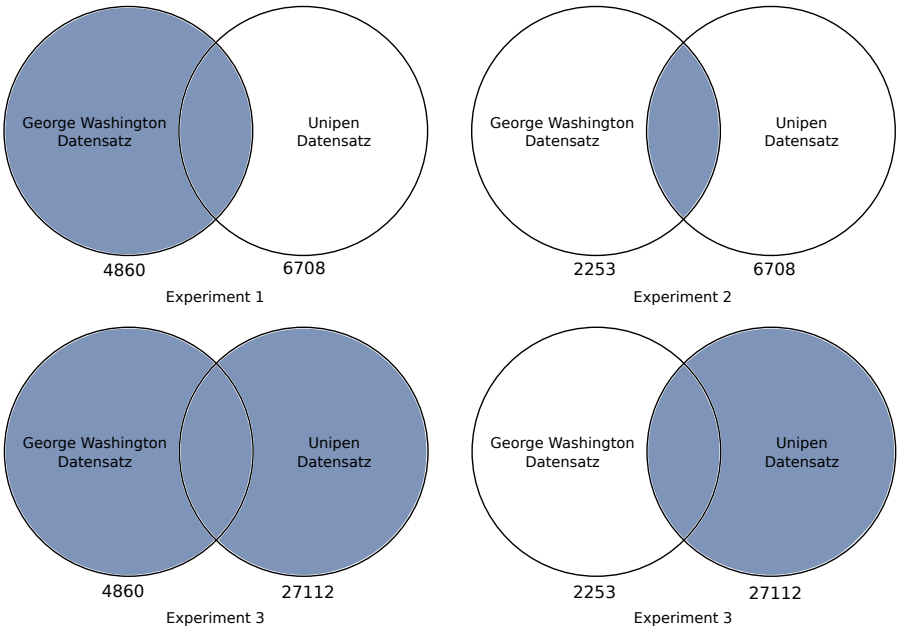


Abbildung 33: Verwendete Daten in Multischreiber Experimenten. Dargestellt sind die Wortmengen aus der George Washington- und Unipen-Datenbank. Ein Kreis steht für die Menge aller verschiedenen Wörter im jeweiligen Datensatz. Farbig markiert ist die Auswahl aus diesen Wörtern, welche für das jeweilige Experiment verwendet wird. Unter jedem Kreis ist die vollständige verwendete Anzahl (inkl. Duplikate) der Wortbilder bzw. Trajektorien in diesem Experiment angegeben, welche aus der Wortauswahl resultiert.

Experiment 1. Alle Wortbilder, 6708 Unipen-Trajektorien aus dem gemeinsamen Wortschatz.
Experiment 2. Nur Wortbilder und Trajektorien aus dem gemeinsamen Wortschatz
Experiment 3. Alle Wortbilder und UNIPEN-Trajektorien.
Experiment 4. Alle UNIPEN-Trajektorien, 2253 Wortbilder aus dem gemeinsamen Wortschatz.
Jedes Experiment wird als Kreuzvalidierung durchgeführt (siehe Abschnitt 7.2). Dazu werden die verwendeten Wortbilder des GW-Datensatzes und die zugehörigen Trajektorien des GWO-Datensatzes in vier Teilmengen aufgeteilt, wobei die Paarungen von Wortbildern und Trajektorien, welche das gleiche Wort repräsentieren, bestehen bleiben (vgl. Abschnitt 7.1.2). Mit den Wortbildern von jeweils drei Teilmengen werden die visuellen Attribut-SVMs trainiert. Alle gewählten UNIPEN-Trajektorien werden für das Training der Online-Handschrift Attribut-SVMs verwendet - die drei entsprechenden Teilmengen des GWO-Datensatzes kommen hier nicht zum Einsatz. Auf der verbleibenden Teilmenge des GW- und GWO-Datensatzes wird

		Variante	mAP
	Experiment 1	Att.	10.79
		Att. + Platt's	23.84
	Experiment 2	Att.	39.07
		Att. + Platt's	45.64
	Experiment 3	Att.	10.90
		Att. + Platt's	24.20
	Experiment 4	Att.	35.78
		Att. + Platt's	41.78

Tabelle 5: Ergebnisse der Multischreiber-Experimente. Die erste Spalte stellt die in Abbildung 33 gezeigte Auswahl der verwendeten Wortbilder und Online-Handschrift Trajektorien dar.

das berechnete Modell getestet. Die Ergebnisse der im Folgenden beschriebenen Experimente sind in Tabelle 5 zusammengefasst.

In Experiment 1 werden alle Wortbilder verwendet sowie 6708 UNIPEN-Trajektorien, welche Wörter repräsentieren, die auch im GW-Datensatz vorkommen. Die „Att." Variante erreicht dabei eine mAP von 10.79%, die „Att.+Platt's" Variante ist etwas besser mit 23.86% mAP. Die erhöhte Variabilität der Handschrift erzeugt im Gegensatz zu den Ergebnissen der schreiberabhängigen Experimente (81.37% bzw. 79.99%) eine deutliche Verschlechterung der Erkennungsleistung. Dadurch kann keine der beiden evaluierten Varianten von den in den Trainingsbeispielen enthaltenen Handschriftstilen effektiv auf die Eigenschaften der Handschrift der Testtrajektorien schliessen.

Das zweite Experiment verwendet nur Wortbilder und Trajektorien des gemeinsamen Wortschatzes aus 223 Wörtern. Durch den kleineren Wortschatz wird sowohl im Training als auch beim Test das Rauschen reduziert, welches durch Wortbilder erzeugt wird, für die keine entsprechende Trajektorie vorhanden ist, welche das gleiche Wort repräsentiert. Das Rauschen bezieht sich hierbei auf Wortbilder, welche nicht das angefragte Wort repräsentieren, aber trotzdem ähnliche visuelle Merkmale aufweisen. Solche Wortbilder werden durch das Verfahren eher als relevant eingestuft, wenn sie nicht in den Trainingsbeispielen enthalten sind. Für die gewählte Konfiguration der verwendeten Wortbilder und Trajektorien bedeutet dies, dass das Modell das Retrieval nur in Wortbildern durchführen muss, deren abgebildete Wörter bereits in Trainingsbeispielen enthalten waren. Dadurch verbessert sich das Ergebnis im Vergleich zu Experiment 1 deutlich auf 39.07% mAP bzw. 45.64% mAP.

In Experiment 3 werden alle zur Verfügung stehenden Wortbilder und Trajektorien verwendet. Dies entspricht am ehesten der Situation, die in der Praxis vorzufinden ist, bei der eine Menge von Trajektorien und die Wortbilder eines zu durchsuchenden

Dokuments gegeben sind, die lediglich in einer kleinen Menge von Wörtern übereinstimmen. Die mAP-Werte sind vergleichbar mit den Ergebnissen aus Experiment 1.

Experiment 4 verwendet alle Unipen-Trajektorien und die Wortbilder aus dem gemeinsamen Wortschatz. Die Ergebnisse sind mit 35.78% mAP bzw. 41.78% mAP ähnlich zu den Ergebnissen in Experiment 2. Die leichte Verschlechterung ist zurückzuführen auf die Verwendung von mehr Trajektorien in Experiment 4. Dadurch wird mehr Rauschen erzeugt, als wenn nur die Trajektorien im Modelltraining verwendet werden, welche auch im Test angefragt werden.

Durch die Experimente wird ein guter Überblick über das Verhalten der schreiberunabhängigen Modelle in Abhängigkeit der Trainingsdaten gegeben. Experiment 1 und 3 zeigen, dass bei Verwendung aller Wortbilder die Erkennungsleistung niedrig ist. Die Erkennungsleistung steigt nahezu unabhängig von der Menge der verwendeten Trajektorien, wenn weniger Wortbilder verwendet werden (siehe Experiment 2 und 4). Daher ist davon auszugehen, dass die Qualität der berechneten Abbildung zwischen der Online-Handschrift Merkmalsdomäne und der visuellen Merkmalsdomäne nicht ausreicht, um zuverlässig Wortbilder zu durchsuchen, deren repräsentierte Wörter nicht in den Trainingsbeispielen enthalten sind. Da die Ergebnisse aus Experiment 2 und 4 sehr ähnlich sind wird zusammen mit der zuvor beschriebenen Beobachtung deutlich, dass es im Training weniger darauf ankommt, viele Trajektorien pro Schreiber zu verwenden, sondern gerade die Trajektorien, deren repräsentierte Wörter auch im zu durchsuchenden Dokument vorkommen. Ist der repräsentierte Wortschatz von Wortbildern und Trajektorien gleich, ist „Att.+Platt's" mit 45.64% mAP die bessere der beiden evaluierten Varianten.

7.6 Zusammenfassung

In diesem Kapitel wurden die Experimente zur Auswertung der in dieser Arbeit vorgeschlagenen Word Spotting Verfahren zum neuen Word Spotting mit Online-Handschrift Anfragen beschrieben. Zunächst wurde anhand der Baseline-Experimente zu den zwei vorgestellten Query-by-Example und Query-by-String Word Spotting Verfahren die Korrektheit der erstellten Implementierungen festgestellt. Im Fall der LSA-Methode konnte eine deutliche Steigerung der Erkennungsleistung im Gegensatz zur Vorlage in [ARTL13] erreicht werden. Für die Embedded Attributes-Methode wurde gezeigt, dass der Bag-of-Features Ansatz mit einer Spatial Pyramid als Merkmalsrepräsentation für Wortbilder eine geeignete Alternative zu den in [AGFV14a] verwendeten Fisher Vektoren mit Koordinatenerweiterung darstellt.

Für die zum ersten Mal vorgestellten Query-by-Online-Trajectory Word Spotting Verfahren wurden in einem schreiberabhängigen

Experiment herausragende Ergebnisse erzielt. Dabei liegen die erreichten Werte nur geringfügig unter den Werten der Query-by-String Verfahren. Dies ist aufgrund der viel höheren Variabilität von Online-Handschrift gegenüber Zeichenketten beeindruckend. Zwischen zwei Zeichenketten, welche das selbe Wort repräsentieren, gibt es keine Variabilität, da dieses Wort stets aus den gleichen Zeichen zusammengesetzt ist. Die Ergebnisse zeigen zum einen, dass die neue Bag-of-Online-Features Merkmalsrepräsentation eine geeignete Wahl zur Merkmalsrepräsentation von Online-Handschrift Trajektorien ist. Zum anderen bestätigt es die Theorie, dass sich sowohl die LSA- als auch Embedded Attributes-Methode für das QbO Word Spotting erfolgreich einsetzen lassen.

Diese Methoden wurden zudem in vier Experimenten mit schreiberunabhängigen Anfragen eingesetzt, die ebenfalls vielversprechende Ergebnisse lieferten. Diese Experimente zeigen, dass die Qualität der berechneten schreiberunabhängigen Word Spotting Modelle abhängig ist von den im Training beobachteten Wörtern. Für den eingeschränkten Fall, in dem Wortbilder und Online-Handschrift Trajektorien Wörter aus dem selben Wortschatz repräsentieren, wurde durch die Platt's Scaling-Variante mit 45.64% mAP die beste Erkennungsleistung erzielt. Aufgrund der Anforderung an schreiberunabhängige Verfahren, auch Anfragen von Handschriftstilen bearbeiten zu können, welche nicht in den Trainingsbeispielen enthalten sind, ist dieses Ergebnis bemerkenswert. In den schreiberabhängigen Experimenten lagen für den einzigen verwendeten Handschriftstil im Training jeweils etwa 3600 Beispieltrajektorien vor. Bei den Multischreiber-Experimenten lag die Anzahl der Beispieldaten pro Schreiber nur etwa bei 430 Trajektorien unter Verwendung aller UNIPEN-Trajektorien. Deutlich weniger Beispiele wurden in den Experimenten 1 und 2 verwendet. Unter diesem Gesichtspunkt sind die erreichten Erkennungszahlen beeindruckend.

8 ZUSAMMENFASSUNG

Beim Word Spotting wird eine Menge von Dokumenten nach relevanten Vorkommen von Wörtern zu einer gegebenen Anfrage durchsucht. Die Anfrage besteht dabei in der Regel aus einem durch einen Anwender gewählten Wortbild (Query-by-Example) oder der Eingabe eines Wortes als Zeichenkette (Query-by-String). Word Spotting ist den Bereichen des Bild-Retrieval und der Dokumentenanalyse zuzuordnen. In dieser Arbeit wurde eine neue, dritte Anfrageart motiviert und evaluiert, die Anfrage mit Online-Handschrift Trajektorien. Diese eignet sich für Situationen, in denen das manuelle Markieren eines Wortbildes zur Anfrage oder das Benutzen einer Tastatur für eine Eingabe als Zeichenkette nicht gewünscht ist. Durch den Benutzer wird die Anfrage dabei beispielsweise auf einem Touchscreen oder Smartboard handschriftlich verfasst.

Zur Evaluierung dieser neuen Methode wurden zwei Query-by-String Verfahren aus [ARTL13] und [AGFV14a] in Kapitel 5 vorgestellt und Änderungen für Anfragen mit Online-Handschrift in Kapitel 6 diskutiert. Beide Verfahren arbeiten segmentierungsbasiert und repräsentieren Wortbilder anhand des Bag-of-Features Ansatzes und einer Spatial Pyramid. In dieser Arbeit wurde der Bag-of-Features Ansatz zum ersten Mal auch für Online-Handschrift Trajektorien verwendet. Dafür wird durch Clustering ein Online-Handschrift Vokabular aus verschiedenen Ausprägungen von Online-Handschrift Merkmalsvektoren berechnet, welche für jeden Punkt einer Trajektorie aus einer Menge von Beispieltrajektorien berechnet werden. Für eine neue Online-Handschrift Trajektorie wird je Punkt ein Merkmalsvektor berechnet und anhand des Online-Handschrift Vokabulars quantisiert. Die Merkmalsrepräsentation wird bestimmt, indem ein Histogramm über die Vorkommen der quantisierten Merkmalsvektoren in der Trajektorie erstellt wird. Diese neue Merkmalsrepräsentation wird *Bag-of-Online-Features* genannt. Wie bei der Erweiterung einer Bag-of-Features Repräsentation durch eine Spatial Pyramid, werden anschliessend auch zur Bag-of-Online-Features Repräsentation Lokalitätsinformationen über eine Spatial Pyramid hinzugefügt.

Um eine Online-Handschrift Anfrage zu beantworten, werden die Merkmalsrepräsentationen von Trajektorie und Wortbildern verglichen. Da diese jedoch aus unterschiedlichen Merkmalsdomänen stammen, ist dazu ein Modelltrainingsschritt notwendig, der einen Vektorraum berechnet, über den der domänenübergreifende Vergleich er-

möglicht wird. Bei dem Verfahren nach [ARTL13] wird mittels Latent Semantic Analysis ein Topic Raum bestimmt, in welchem Korrelationen zwischen visuellen Merkmalen und Online-Handschrift Merkmalen berechnet werden. Die in den Topic Raum projizierten Merkmalsrepräsentationen können anschliessend verglichen werden, da Trajektorien und Wortbilder, welche das selbe Wort repräsentieren, den gleichen Topics angehören. Beim Attribute Embedding Verfahren nach [AGFV14a] werden für die Merkmalsrepräsentation von Wortbildern und Online-Handschrift Trajektorien zwei separate Mengen von Support Vector Machines für Attribute gelernt. Attribute werden über die textuellen Transkriptionen von Beispieldaten bestimmt und spiegeln das Vorkommen von unterschiedlichen Buchstaben in Regionen des Wortbildes bzw. der Trajektorie wider. Durch Bestimmung des SVM-Scores für eine entsprechende Merkmalsrepräsentation wird berechnet, ob ein Wortbild bzw. eine Trajektorie das, der SVM entsprechende, Attribut besitzt. Für den Vergleich eines Wortbildes und der angefragten Trajektorie werden Vektoren aus SVM-Scores verglichen, was dem Vergleich von Attributen entspricht. Verschiedene Varianten dieser Methode normalisieren die Scores, um bessere Ergebnisse zu erzielen.

Zur Evaluation der Verfahren in Kapitel 7 wurde zunächst ein schreiberabhängiger Test durchgeführt, bei dem alle Anfragen von dem selben Schreiber stammen. Die LSA-Methode erreichte ein Ergebnis von 66.81% mAP, die beste Variante der Embedded Attributes Methode erreichte 86.49% mAP. Diese hohe bzw. sehr hohe Erkennungsrate bedeutet für einen Anwender des Word Spotting Systems, dass die meisten Suchergebnisse, welche zu einer Anfrage ermittelt werden, das gesuchte Wort darstellen. Durch die erreichten Werte wird deutlich, dass der neue Bag-of-Online-Features Ansatz mit einer Spatial Pyramid eine geeignete Merkmalsrepräsentation für Online-Handschrift Trajektorien im Bereich des Word Spotting darstellt. Zudem wurde gezeigt, dass sich sowohl LSA- als auch Embedded Attribute Methode erfolgreich für das QbO Word Spotting mit schreiberabhängigen Anfragen einsetzen lassen.

Eine Evaluation mit Online-Handschrift Trajektorien von mehreren Schreiber wurde durchgeführt, um die Funktionsweise zweier Varianten des Embedded Attributes Verfahren für ein praxisrelevantes Szenario zu zeigen. In der Praxis wird ein Word Spotting System von mehr als nur einem Anwender benutzt. Daher ist es wichtig, dass das System auch unabhängig vom Handschriftstil Anfragen bearbeiten kann. Das jeweilige Word Spotting Verfahren wurde dazu mit Online-Handschrift Trajektorien von 62 Schreibern aus dem UNIPEN-Datensatz [Uni] trainiert. Die Anfragen stammten weiterhin von nur einem Schreiber, dessen Handschriftstil jedoch nicht in den Beispieldaten zum Training enthalten war. Dadurch wurde dieser Handschriftstil im Trainingsschritt nicht beobachtet und das Ver-

halten der Verfahren bei Bearbeitung eines fremden Handschriftstils evaluiert. Weiterhin wurde die Auswahl der Trainingsdaten in verschiedenen Experimenten variiert, wodurch die Anforderungen an die Trainingsdaten für den Praxiseinsatz der Verfahren evaluiert wurden. Durch diese Experimente wurde die Abhängigkeit der Verfahren von den zum Training vorliegenden Beispieldaten deutlich. Die verwendeten Beispieldaten für Wortbilder und Online-Handschrift Trajektorien müssen im Wortschatz größtenteils übereinstimmen, damit sowohl für die visuelle Merkmalsrepräsentation, als auch die Online-Handschrift Merkmalsrepräsentation die gleichen Attribute gelernt werden können.

Die besten Ergebnisse in den Experimenten erzielte die Variante „Att.+Platt's", welche die SVM-Scores durch Platt's Scaling kalibriert. In einem Experiment mit nur teilweiser Überschneidung des Wortschatzes von Wortbildern und Trajektorien, lag die Erkennungsrate dieser Variante bei lediglich 23.84% mAP, was die zuvor aufgestellte These bestätigt. Wenn Wortbilder und Online-Handschrift Trajektorien zur Verfügung stehen, deren repräsentierte Wörter später angefragt werden, liefert die „Att.+Platt's" Variante jedoch sehr gute Ergebnisse. Mit einem Höchstwert von 45.64% mAP in einem Experiment, bei dem Wortbilder und Online-Handschrift Trajektorien benutzt werden, welche den selben Wortschatz repräsentieren, liegt die Erkennungsrate deutlich unter der, der zuvor beschriebenen schreiberabhängigen Experimenten (Att.+Platt's: 79.99% mAP). Die Ergebnisse der beiden Experimente sind vergleichbar, da jeweils die gleichen Online-Handschrift Trajektorien und Wortbilder als Testset verwendet wurden. Zwei Faktoren sind bei diesem Vergleich jedoch zu beachten. Zum einen liegt eine vielfach höhere Variabilität der Online-Handschrift bei Verwendung von mehreren Handschriftstilen vor. Zum anderen lag die Anzahl der Trainingsbeispiele in dem angeführten Multischreiber-Experiment mit etwa 110 Trajektorien pro Schreiber deutlich unter der verwendeten Anzahl in den schreiberabhängigen Experimenten (etwa 3600 Trajektorien). Daraus wird deutlich, dass die schreiberunabhängigen Verfahren vermutlich erheblich verbessert werden können, indem die Anzahl der Beispieltrajektorien für jeden Handschriftstil im Training erhöht wird. Die erreichten Ergebnisse sind daher durchaus beeindruckend.

Insgesamt zeigen die Ergebnisse dieser Arbeit, dass der Query-by-Online-Trajectoy Ansatz eine geeignete Alternative zu Query-by-Example und Query-by-String darstellt. Durch erfolgreiche Anpassung von zwei bereits bestehenden Query-by-String Word Spotting Verfahren für diese neue Art der Anfrage, die in Experimenten sehr gute Ergebnisse lieferten, wird zudem deutlich, dass bereits bekannte Methoden verwendet werden können, um QbO Word Spotting Verfahren umzusetzen.

ABBILDUNGSVERZEICHNIS

[AGFV14a] Almazan, J. ; Gordo, A. ; Fornes, A. ; Valveny, E.: Word Spotting and Recognition with Embedded Attributes. In: *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 36 (2014), Dec, Nr. 12, S. 2552–2566

[AGFV14b] Almazán, J. ; Gordo, A. ; Fornés, A. ; Valveny, E.: Segmentation-free word spotting with exemplar {SVMs}. In: *Pattern Recognition* 47 (2014), Nr. 12, S. 3967 – 3978

[ARTL13] Aldavert, D. ; Rusinol, M. ; Toledo, R. ; Llados, J.: Integrating Visual and Textual Cues for Query-by-String Word Spotting. In: *Document Analysis and Recognition (ICDAR), 2013 12th International Conference on*, 2013, S. 511–515

[BB12] Basheri, M. ; Burd, L.: Exploring the significance of multi-touch tables in enhancing collaborative software design using UML. In: *Frontiers in Education Conference (FIE)*, 2012, S. 1–5

[Bur98] Burges, C. J. C.: A Tutorial on Support Vector Machines for Pattern Recognition. In: *Data Mining and Knowledge Discovery* 2 (1998), Nr. 2, S. 121–167

[BYRN99] Baeza-Yates, R. A. ; Ribeiro-Neto, B.: *Modern Information Retrieval*. Boston, MA, USA : Addison-Wesley Longman Publishing Co., Inc., 1999

[DDF+90] Deerwester, S. ; Dumais, S. T. ; Furnas, G. W. ; Landauer, T. K. ; Harshman, R.: Indexing by latent semantic analysis. In: *JOURNAL OF THE AMERICAN SOCIETY FOR INFORMATION SCIENCE* 41 (1990), Nr. 6, S. 391–407

[DT05] Dalal, N. ; Triggs, B.: Histograms of Oriented Gradients for Human Detection. In: *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 1 - Volume 01*. Washington, DC, USA : IEEE Computer Society, 2005 (CVPR '05), S. 886–893

[ETF+05] Edwards, J. ; Teh, Y. W. ; Forsyth, D. ; Bock, R. ; Maire, M. ; Vesom, G.: Making latin manuscripts searchable using gHMM's. In: *Advances in Neural Information*

Processing Systems 17: Proceedings of the 2004 Conference 17 (2005), S. 385

[Fino8] Fink, G. A.: *Markov Models for Pattern Recognition: From Theory to Applications*. Berlin, Heidelberg : Springer-Verlag Berlin Heidelberg, 2008

[FKFB12] Fischer, A. ; Keller, A. ; Frinken, V. ; Bunke, H.: Lexicon-free Handwritten Word Spotting Using Character HMMs. In: *Pattern Recognition Letters* 33 (2012), Mai, Nr. 7, S. 934–942. – Special Issue on Awards from {ICPR} 2010

[FR14] Fink, Gernot A. ; Rothacker, Leonard: *Statistical Models for Handwriting Recognition and Retrieval*. Tutorial (invited) presented at Int. Conf. on Frontiers in Handwriting Recognition, Crete, Greece, 2014

[GAC+91] Guyon, I. ; Albrecht, P. ; Cun, Y. L. ; Denker, J. ; Hubbard, W.: Design of a neural network character recognizer for a touch terminal. In: *Pattern Recognition* 24 (1991), Nr. 2, S. 105 – 119

[GW] Manuscript Division, Library of Congress, Washington, D.C.: *George Washington Papers at the Library of Congress, 1741–1799*. – http://memory.loc.gov/ammem/gwhtml/gwhome.html

[HTF09] Hastie, T. ; Tibshirani, R. ; Friedman, J.: *The Elements of Statistical Learning*. 2. New York, NY, USA : Springer New York Inc., 2009 (Springer Series in Statistics)

[JDS11] Jegou, H. ; Douze, M. ; Schmid, C.: Product Quantization for Nearest Neighbor Search. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 33 (2011), Jan, Nr. 1, S. 117–128

[JMW00] Jaeger, S. ; Manke, S. ; Waibel, A.: Npen++: An On-Line Handwriting Recognition System. In: *in 7th International Workshop on Frontiers in Handwriting Recognition*, 2000, S. 249–260

[KAA+00] Kolcz, A. ; Alspector, J. ; Augusteijn, M. ; Carlson, R. ; Viorel Popescu, G.: A Line-Oriented Approach to Word Spotting in Handwritten Documents. In: *Pattern Analysis & Applications* 3 (2000), Nr. 2, S. 153–168. – ISSN 1433–7541

[KGN+07] Konidaris, T. ; Gatos, B. ; Ntzios, K. ; Pratikakis, I. ; Theodoridis, S. ; Perantonis, S. J.: Keyword-guided

word spotting in historical printed documents using synthetic data and user feedback. In: *International Journal of Document Analysis and Recognition (IJDAR)* 9 (2007), Nr. 2-4, S. 167–177

[LB06] Liwicki, M. ; Bunke, H.: HMM-based on-line recognition of handwritten whiteboard notes. In: *in Tenth International Workshop on Frontiers in Handwriting Recognition* Suvisoft, 2006

[LLW07] Lin, H.-T. ; Lin, C.-J. ; Weng, R. C.: A note on Platt's probabilistic outputs for support vector machines. In: *Machine Learning* 68 (2007), Nr. 3, S. 267–276

[LOLE09] Leydier, Y. ; Ouji, A. ; LeBourgeois, F. ; Emptoz, H.: Towards an omnilingual word retrieval system for ancient manuscripts. In: *Pattern Recognition* 42 (2009), Nr. 9, S. 2089–2105

[Low99] Lowe, D. G.: Object recognition from local scale-invariant features. In: *The Proceedings of the Seventh IEEE International Conference on Computer Vision, 1999.* Bd. 2, 1999, S. 1150–1157

[Low04] Lowe, D. G.: Distinctive Image Features from Scale-Invariant Keypoints. In: *International Journal of Computer Vision* 60 (2004), November, Nr. 2, S. 91–110

[LRF+12] Lladós, J. ; Rusiñol, M. ; Fornés, A. ; Fernández, D. ; Dutta, A.: On the influence of word representations for handwritten word spotting in historical documents. In: *International journal of pattern recognition and artificial intelligence* 26 (2012), Nr. 05

[LSP06] Lazebnik, S. ; Schmid, C. ; Ponce, J.: Beyond Bags of Features: Spatial Pyramid Matching for Recognizing Natural Scene Categories. In: *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition* Bd. 2, 2006, S. 2169–2178

[MB01] Marti, U.-V. ; Bunke, H.: Using a statistical language model to improve the performance of an HMM-based cursive handwriting recognition system. In: *International Journal of Pattern Recognition and Artificial Intelligence* 15 (2001), Nr. 01, S. 65–90

[MFW94] Manke, S. ; Finke, M. ; Waibel, A.: Combining bitmaps with dynamic writing information for on-line handwriting recognition. In: *Proceedings of the 12th IAPR International Conference on Pattern Recognition, 1994. Vol. 2*

- *Conference B: Computer Vision & Image Processing.* Bd. 2, 1994, S. 596–598

[MHR96] Manmatha, R. ; Han, Chengfeng ; Riseman, E. M.: Word Spotting: A New Approach to Indexing Handwriting. In: *Proceedings of the 1996 Conference on Computer Vision and Pattern Recognition (CVPR '96)*. Washington, DC, USA : IEEE Computer Society, 1996 (CVPR '96), S. 631–

[OD11] O'Hara, S. ; Draper, B. A.: Introduction to the Bag of Features Paradigm for Image Classification and Retrieval. In: *CoRR* abs/1101.3354 (2011)

[Oxf] *Early Manuscripts at Oxford University, Bodleian library ms. auct. f. 2.13.* – http://image.ox.ac.uk/

[PF09] Plötz, T. ; Fink, G. A.: Markov models for offline handwriting recognition: a survey. In: *International Journal on Document Analysis and Recognition (IJDAR)* 12 (2009), Nr. 4, S. 269–298

[Pla98] Platt, J.: Sequential Minimal Optimization: A Fast Algorithm for Training Support Vector Machines / Microsoft Research. 1998 (MSR-TR-98-14). – Forschungsbericht. – 21 S.

[Pla99] Platt, J.: Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. In: *Advances in large margin classifiers* 10 (1999), Nr. 3, S. 61–74

[PSM10] Perronnin, F. ; Sánchez, J. ; Mensink, T.: Improving the Fisher Kernel for Large-scale Image Classification. In: *Proceedings of the 11th European Conference on Computer Vision: Part IV*. Berlin, Heidelberg : Springer-Verlag, 2010 (ECCV'10), S. 143–156

[PTV05] Pastor, M. ; Toselli, A. ; Vidal, E.: Writing Speed Normalization for On-Line Handwritten Text Recognition. In: *Proceedings of the Eighth International Conference on Document Analysis and Recognition*. Washington, DC, USA : IEEE Computer Society, 2005 (ICDAR '05), S. 1131–1135

[RATL11] Rusinol, M. ; Aldavert, D. ; Toledo, R. ; Llados, J.: Browsing Heterogeneous Document Collections by a Segmentation-Free Word Spotting Method. In: *Document Analysis and Recognition (ICDAR), 2011 International Conference on*, 2011, S. 63–67

[RATL15] Rusiñol, M. ; Aldavert, D. ; Toledo, R. ; Lladós, J.: Efficient segmentation-free keyword spotting in historical document collections. In: *Pattern Recognition* 48 (2015), Nr. 2, S. 545–555

[RM03] Rath, T. M. ; Manmatha, R.: Word image matching using dynamic time warping. In: *Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on* Bd. 2 IEEE, 2003, S. II–521

[RM07] Rath, T. M. ; Manmatha, R.: Word spotting for historical documents. In: *International Journal on Document Analysis and Recognition* (2007), S. 139–152

[RP08] Rodriguez, J. A. ; Perronnin, F.: Local gradient histogram features for word spotting in unconstrained handwritten documents. In: *Proceedings of the 1st International Conference on Handwriting Recognition (ICFHR'08)*, 2008

[RSP12a] Rodriguez-Serrano, J. A. ; Perronnin, F.: Synthesizing queries for handwritten word image retrieval. In: *Pattern Recognition* 45 (2012), Nr. 9, S. 3270–3276

[RSP12b] Rodriguez-Serrano, J.A. ; Perronnin, F.: A Model-Based Sequence Similarity with Application to Handwritten Word Spotting. In: *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 34 (2012), Nov, Nr. 11, S. 2108–2120

[SKBB12] Scheirer, W. ; Kumar, N. ; Belhumeur, P. N. ; Boult, T. E.: Multi-Attribute Spaces: Calibration for Attribute Fusion and Similarity Search. In: *The 25th IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012, S. 2933–2940

[SRF] Sudholt, S. ; Rothacker, L. ; Fink, G. A.: Learning Local Image Descriptors for Word Spotting. – accepted for ICDAR 2015

[SZ03] Sivic, J. ; Zisserman, A.: Video Google: a text retrieval approach to object matching in videos. In: *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*, 2003, S. 1470–1477 vol.2

[Uni] International Unipen Foundation: *The UNIPEN on-line handwritten samples collection release #1*. – http://www.unipen.org/products.html

[Vin02] Vinciarelli, A.: A survey on off-line cursive word recognition. In: *Pattern recognition* 35 (2002), Nr. 7, S. 1433–1446

[WG05] Wu, J. ; Graham, T. C. N.: The software design board: a tool supporting workstyle transitions in collaborative software design. In: *Engineering Human Computer Interaction and Interactive Systems*. Springer, 2005, S. 363–382

[WYY+10] Wang, J. ; Yang, J. ; Yu, K. ; Lv, F. ; Huang, T. ; Gong, Y.: Locality-constrained linear coding for image classification. In: *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, 2010, S. 3360–3367