Optimierung der Merkmalberechnung für die automatische Spracherkennung

Studienarbeit im Fach Informatik

vorgelegt
von

Christian Hacker

Geboren am 08.09.1975 in Amberg

Angefertigt am

Lehrstuhl für Mustererkennung (Informatik 5)
Institut für Informatik
Friedrich-Alexander-Universität Erlangen-Nürnberg.

Betreuer: Dipl.Inf. Georg Stemmer, Dr.-Ing. Elmar Nöth

Beginn der Arbeit: 01.03.2001

Abgabe der Arbeit: 29.11.2001

Ich versichere, dass ich die Arbeit ohne fremde Hilfe und ohne Benutzung anderer als der angegebenen Quellen angefertigt habe und dass die Arbeit in gleicher oder ähnlicher Form noch keiner anderen Prüfungsbehörde vorgelegen hat und von dieser als Teil einer Prüfungsleistung angenommen wurde. Alle Ausführungen, die wörtlich oder sinngemäß übernommen wurden, sind als solche gekennzeichnet.

Die Richtlinien des Lehrstuhls für Studien- und Diplomarbeiten habe ich gelesen und anerkannt, insbesondere die Regelung des Nutzungsrechts.

Erlangen, den 29. November 2001

Übersicht

Ausgangspunkt der Studienarbeit ist der 24-dimensionale Mel-Cepstrum basierte Merkmalvektor, der bisher am LME für die automatische Spracherkennung eingesetzt wird. Es wird untersucht, ob die Merkmalberechnung mit der folgenden Vorgehensweise optimiert werden kann: In verschiedenen Experimenten werden höher-dimensionale Vektoren dadurch erzeugt, dass man die Merkmale parallel über verschiedene Zeitauflösungen berechnet bzw. Produkte höherer Ordnung aus den bisherigen Merkmalkomponenten bildet. Danach wird die Dimension mit Hilfe der Karhunen-Loève-Transformation (KLT) wieder auf 24 reduziert.

Durch getrennte oder gemeinsame Behandlung der unterschiedlichen Merkmalgruppen (statische und dynamische Merkmale) sowie durch Hinzufügen oder Weglassen vieler optionaler Rechenschritte ergeben sich zahlreiche Variationen der Experimente. Nach einer Vorstellung der Grundlagen und neuerer Veröffentlichungen zur Merkmalberechnung werden Verfahren der problemabhängigen Reihenentwicklung, insbesondere die KLT, erläutert und der Versuchsaufbau beschrieben. Anschließend werden die verschiedenen Untersuchungen motiviert und die Ergebnisse diskutiert. Ausgewählte erfolgreiche Experimente werden zudem mit der PPCA (“Probabilistic Principal Component Analysis”) durchgeführt.

Inhaltsverzeichnis

Kapitel 1

Einleitung

1.1 Spracherkennung heute

Sprache ist die wichtigste Verständigungsform unter Menschen. Auch zwischen Mensch und Maschine gewinnt die Sprache an zunehmender Bedeutung. Diktiersysteme kann man bereits für wenig Geld im nächsten Kaufhaus erwerben und das Tippen lästig langer Telefonnummern wird oft durch kurzes Sprechen des gewünschten Namens ersetzt. Automatische Dialogsysteme geben dem Anrufer der jeweiligen Hotline zu jeder Tageszeit Auskunft, und so erhält man beispielsweise die Nahverkehrsverbindung von Amberg-Raigering nach Erlangen, Technische Fakultät, samt der vier Umsteigepunkte nach wenigen Minuten geduldigen Durchfragens[1]. Eine Kinoauskunft für den Großraum Nürnberg gibt FRÄNKI[2], das am Lehrstuhl für Mustererkennung (LME) der Universität Erlangen-Nürnberg entwickelte Kinoauskunftsystem, dank seiner ausgeklügelten Dialogform schon nach kurzem Gespräch.

Insbesondere im Bereich Mobilfunk wird Sprache wohl *das* Mittel zur Mensch-Maschine-Interaktion werden. Denkt man nämlich an immer kleiner und immer komplexer werdende Mobilfunk-Endgeräte, so ist bei gleich bleibender durchschnittlicher Fingerdicke des Menschen das Aus für die kleinen Tasten schon vorhersehbar.

1.2 FRÄNKI - ein Kinoauskunftsystem

Ein automatisches Dialogsystem, das am Lehrstuhl für Mustererkennung (LME) der Universität Erlangen-Nürnberg entwickelt wurde ist EVAR (**E**rkennen,**V**erstehen,**A**ntworten,**R**ückfragen),

[1] Telefonische Fahrplanauskunft des VGN: 01802/993399
[2] Kinoauskunft FRÄNKI: 09131/610016

ein Zugfahrplan-Auskunft-System [Gal98]. Von Sympalog, einer Ausgründung des LME, wurde der Dialogmanager von EVAR völlig neu implementiert und aufgaben- sowie sprachunabhängig gestaltet. Mehrere verschiedene Dialogsysteme wurden daraus entwickelt, teilweise in nur wenigen Tagen. FRÄNKI ist das **fränk**ische **Ki**noauskunftsystem, Stocki ein Börsenkurs-Informationssystem (**Stock I**nformation). Aktueller Stand und Herausforderungen für die Forschung werden in [Nöt01] beschrieben.

FRÄNKI gibt Auskunft über das aktuelle Kinoprogramm der ca. 60 Kinos im Großraum Nürnberg. Die Anfrage erfolgt per Telefon in deutscher Sprache. Es handelt sich also um ein monolinguales System, im Gegensatz zu STOCKI, das auch Anfragen auf Englisch oder Französisch bearbeiten kann. Das Vokabular umfasst etwa 1500 Wortformen, die für solch spezielle Dialoge ausreichen. Einmal pro Woche wird das Programm aus dem Internet auf den aktuellen Stand gebracht. Die neuen Filmtitel werden in semiautomatischer Weise dem Lexikon des Erkenners hinzugefügt. Dazu müssen die neuen Wörter halbmanuell von einem Experten phonetisiert und gesprochen werden. Dieser Aufwand wird in Kauf genommen, da nur einmal pro Woche wenige neue Filme hinzukommen.

Das System zeichnet sich unter anderem durch die flexible Dialogform aus. Der Benutzer kann seine Anfrage frei formulieren, während andere verbreitete Systeme oft ja/nein-Antworten erzwingen. Auch kann der Anrufer die Reihenfolge der Teilziele (z.B. Kinoname, Uhrzeit, Filmtitel) frei wählen und mehrere Informationen in einem Satz geben. So sind z.B. folgende Dialoge möglich:

System: Hallo hier ist FRÄNKI, wie kann ich ihnen helfen?
Benutzer: Hallo, ich möchte in Erlangen ins Kino gehen.
System: Sie möchten in Erlangen ins Kino?
Um wieviel Uhr möchten sie ins Kino?
Benutzer: So um halb acht.
System: Sie möchten gegen 19:30 Uhr ins Kino?
Benutzer: Ja.
System: Ich nenne Ihnen nun die in Frage kommenden Filmtitel ...

System: Hallo hier ist FRÄNKI, wie kann ich ihnen helfen?
Benutzer: Hallo, ich möchte heute gegen 19:30 Uhr in Erlangen ins Kino Manhattan.
...

Die flexible Dialogführung erhöht aber auch die Wahrscheinlichkeit, dass der Benutzer missverstanden wird. Dieses Problem wird dadurch gelindert, dass das System gegebenfalls die

Antwort geschickt nochmal aufgreift. Der Benutzer kann falsch Verstandenes verbessern bzw. seine Anfrage korrigieren.

System: Hallo hier ist FRÄNKI, wie kann ich ihnen helfen?
Benutzer: Hallo, ich möchte in Erlangen ins Kino gehen.
System: Sie möchten in Erlangen ins Kino.
Um wieviel Uhr möchten sie ins Kino?
Benutzer: Nein, lieber doch in Nürnberg, um 20:00 Uhr.
System: Sie möchten gegen 20:00 in Nürnberg ins Kino?
Benutzer: *(unverständlich)*
System: Entschuldigung, ich habe sie leider nicht verstanden.
...

FRÄNKI kann auch auf manche Wörter die eigentlich außerhalb des Vokabulars liegen (OOV: “out of vocabulary”) geeignet reagieren, sowie auf manche Anfragen die außerhalb des Aufgabengebietes liegen (OOD: “out of domain”):

Benutzer: Hallo, ich möchte in Regensburg ins Kino gehen.
System: Das Kinoprogramm für Regensburg ist mir zur Zeit
nicht bekannt.

Benutzer: Hallo, ich möchte einen Aktion-Film sehen.
System: Über das Genre, den Inhalt oder die Qualität der
Filme kann ich leider keine Auskunft geben.

Forschungsgebiete für die Zukunft sind gemäß [Nöt01] u.a. automatisches Finden von OOD oder OOV Situationen in Tests oder Berücksichtigung von Emotionen des Benutzers.

Das Dialogsystem beinhaltet einen Spracherkenner, dessen Erkennungsraten möglichst gut sein sollen. Im folgenden wird erläutert, wie Sprache oder andere Muster klassifiziert werden.

1.3 Ein Klassifikationssystem

Spracherkennung oder z.B. auch Bilderkennung sind Teilgebiete der Mustererkennung. Ziel in jedem Mustererkennungssystem ist es, einem Muster $\boldsymbol{f}^r(\boldsymbol{x})$, etwa dem aufgenommenen Sprachsignal eines Wortes (z.B. “Bahnhof”), eine Klasse Ω_κ, etwa das Wort “Bahnhof”, zuzuordnen.

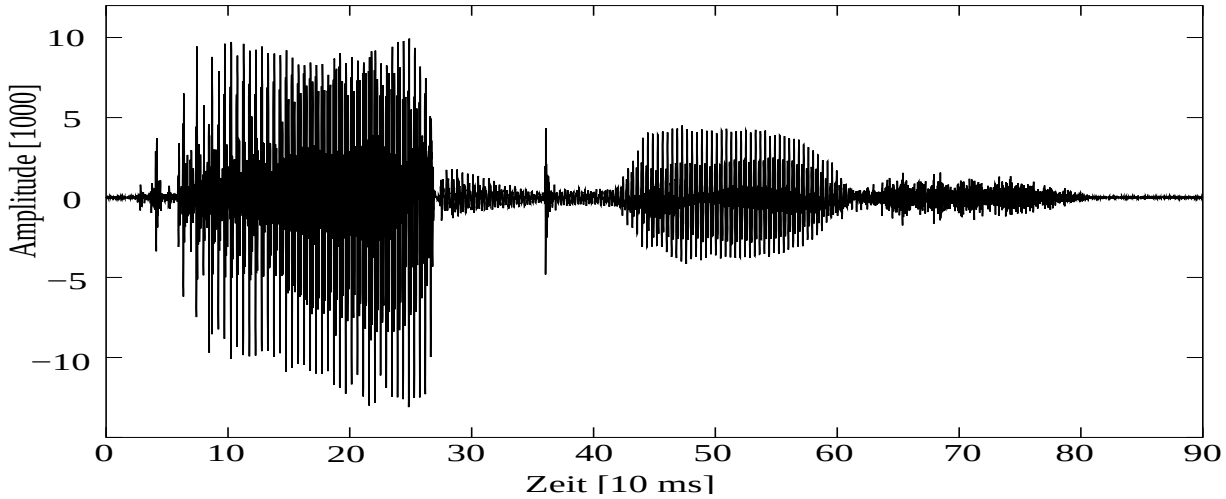

Bild 1.1: Sprachsignal des Wortes “Bahnhof”

Die Klassen Ω_κ sind paarweise disjunkt und bilden zusammen den Problemkreis Ω. Ein Beispiel für ein Sprachsignal zeigt Abbildung 1.1.

Die einzelnen Schritte eines Klassifikationssystems sind in Abbildung 1.2 veranschaulicht. Ein Muster $\boldsymbol{f}$ wird z.B. mit einem Mikrophon aufgenommen und danach gegebenfalls vorverarbeitet, damit man in den folgenden Schritten schneller zu besseren Ergebnissen kommt. Danach werden charakteristische Merkmale berechnet, die im anschließenden Klassifikationsschritt auf eine Klasse Ω_κ abgebildet werden. Für diese letzte Abbildung braucht man Informationen, welche Bereiche im Merkmalraum welcher Klasse zugeordnet werden sollen. Diese gewinnt man aus einer Stichprobe von Mustern in einer Trainings- oder Lernphase [Nie83, S.14f].

Die vorliegende Arbeit beschäftigt sich ausschließlich mit der Merkmalextraktion. Ein Sprachsignal, wie es beispielsweise in Abbildung 1.1 dargestellt ist, wird in Teile zerlegt, die wesentlich kleiner als ein Laut sind. Aus jedem dieser Abschnitte wird eine Gruppe von Merk-

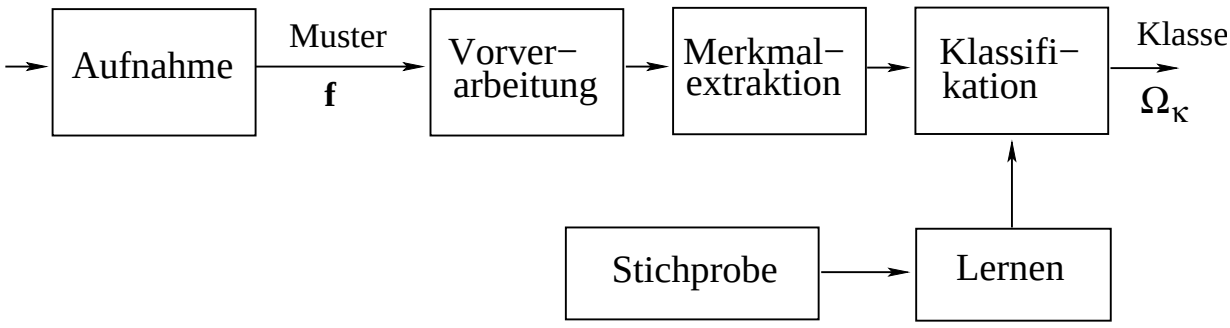

Bild 1.2: Struktur eines Klassifikationssystems. Nach: [Nie83, S.14]

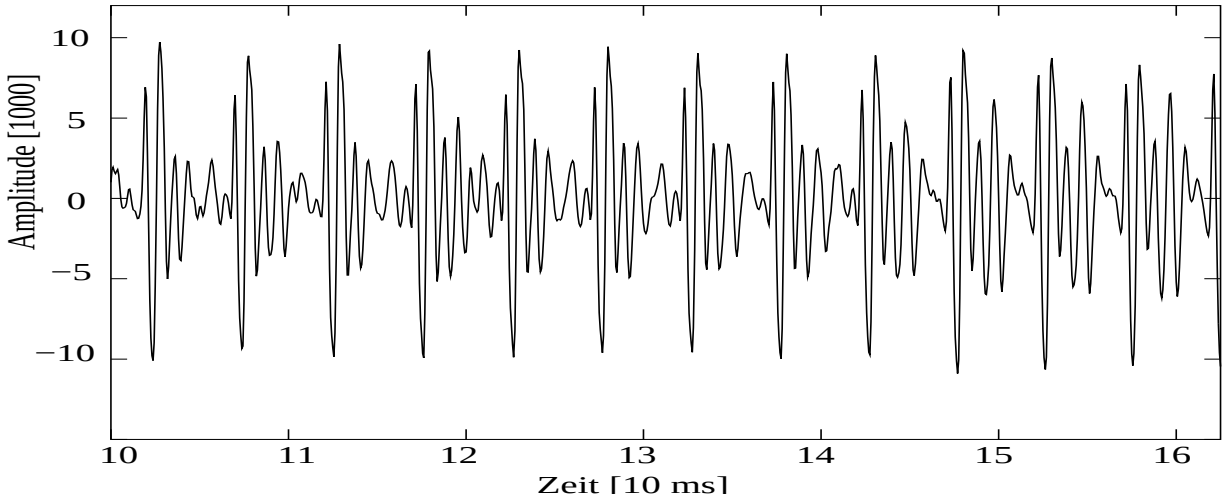

Bild 1.3: Der Laut /a:/ aus “Bahnhof”

malen, ein sogenannter Merkmalvektor, berechnet. Ein Beispiel für ein Merkmal, das auch vom menschlichen Ohr berücksichtigt wird, ist die spektrale Zusammensetzung. Abbildung 1.3 zeigt einen vergrößerten Ausschnitt des Vokals /a:/ aus Abbildung 1.1. Erkennbar ist die regelmäßige Struktur kleiner Teilbereiche eines Sprachsignals. Im Klassifikationsmodul wird der Folge von Merkmalvektoren diejenige Lautkette oder gar Wortkette zugeordnet, die am wahrscheinlichsten gesprochen wurde.

H. Niemann formuliert in [Nie83, S.10ff] sechs Postulate die von jedem Mustererkennungssystem gefordert werden. Nachfolgend werden die für diese Arbeit grundlegenden Postulate zwei und drei zitiert:

> “**Postulat 2:** Ein (einfaches) Muster besitzt Merkmale, die für seine Zugehörigkeit zu einer Klasse charakteristisch sind.
>
> **Postulat 3:** Die Merkmale bilden für Muster einer Klasse einen einigermaßen kompakten Bereich im Merkmalraum. Die von Merkmalen verschiedener Klassen eingenommenen Bereiche sind getrennt.” [Nie83, S.10f]

Weiter heißt es in [Nie83, S.11]:

> “ Das zentrale und allgemein noch ungelöste Problem der Klassifikation besteht darin, solche Merkmale systematisch zu finden, die Postulat 3 genügen. Damit ist hier ein Algorithmus gemeint, der nach Vorgabe einer Stichprobe und eines Maßes für die Leistungsfähigkeit des Systems Merkmale erzeugt, die dieses Maß maximieren (oder minimieren).”

Dieses Problem bleibt ungelöst; Gegenstand dieser Arbeit ist es, den bestehenden heuristischen Merkmalsatz [Rie94, Anhang A], der in Abschnitt 2.3 beschrieben wird, durch systematische Experimente zu verbessern.

1.4 Zielsetzung der Arbeit und Beitrag zur Forschung

Ziel der Studienarbeit ist die Optimierung der Merkmalberechnung für die automatische Spracherkennung. Ausgangspunkt ist der bisher im Spracherkenner am LME verwendete 24-dimensionale Merkmalvektor, der auf dem Mel-Cepstrum basiert. Die Untersuchungen gliedern sich in folgende Punkte:

- Erhöhen der Dimension des Vektors durch Bildung von Produkten höherer Ordnung oder durch parallele Merkmalberechnung über verschiedene Zeitanalyse-Fenster.
- Reduktion der Dimension mit Hilfe einer problemabhängigen Reihenentwicklung.
- Training und Berechnung der Wortakkuratheit des Spracherkenners.

Im Endeffekt soll die Erkennungsrate des Worterkenners (gemessen in Wortakkuratheit und Wortfehlerrate) verbessert werden. Dies gelingt auch in einigen Versuchen wie in [Ste01] berichtet wird.

Die Berechnung der Merkmale in unterschiedlichen Zeitauflösungen wird in verschiedene Untersuchungen aufgeteilt. Da der 24-dimensionale Merkmalvektor aus zwölf statischen Merkmalen besteht, die Informationen über einen engen zeitlichen Abschnitt des Sprachsignals geben, sowie aus zwölf dynamischen, die die Veränderungen der ersten Merkmale über größere Zeitabschnitte berechnen, lassen sich die Untersuchungen mit beiden Teilen des Merkmalvektors getrennt durchführen. Zusätzlich werden die Methoden der Merkmalberechnung variiert. Zu optimieren sind zum einen die in der Spracherkennung bewährten statischen Mel-Cepstrum Merkmale, durch Berücksichtigung mehrerer Zeitauflösungen, zum anderen die dynamischen Merkmale, die, wie sich zeigt, auch in der bisherigen einfachen Auflösung noch Verbesserungsmöglichkeiten bieten. Ein Teil der Experimente mit dem “Multi-Resolution”-Ansatz erweist sich als sehr erfolgreich.

In den Experimenten mit den Produkttermen sollen durch Multiplikation verschiedener Merkmalkomponenten die Ballungsgebiete im Merkmalraum weiter auseinander gezogen und besser getrennt werden. Auch zu diesem Ansatz werden einige Variationen durchgeführt und teilweise Verbesserungen der Erkennungsraten erzielt.

1.5 Aufbau dieser Arbeit

Die Studienarbeit ist wie folgt gegliedert: Im *Kapitel 2* sind die üblichen Verfahren zur Merkmalberechnung aus der Literatur erläutert. Insbesondere ist in einem gesonderten Abschnitt die Vorgehensweise im Programm `fex3_1` (Feature Extraction 3.1), das am LME verwendet wird, zusammengestellt. Dieses Programm wird für die Experimente dieser Studienarbeit modifiziert. Ansätze zur Merkmaloptimierung aus der Literatur folgen.

In allen Untersuchungen dieser Arbeit wird zunächst ein Merkmalvektor mit mehr als 24 Komponenten berechnet und anschließend durch unvollständige Reihenentwicklung wieder auf weniger Dimensionen reduziert. Vorgehensweisen, den theoretischen Hintergrund, und Hinweise zur Implementierung einer solchen Transformation, insbesondere der Karhunen-Loève Transformation (KLT), findet man im *Kapitel 3*. Auch LDA (Lineare Diskriminanzanalyse) und PPCA (Probabilistic Principal Component Analysis) werden kurz vorgestellt.

Eine technische Beschreibung der Experimente gibt *Kapitel 4*. Stichprobe, Trainings- und Testverfahren werden dort erklärt.

In *Kapitel 5* werden sämtliche Experimente motiviert, beschrieben und verglichen. In ersten Versuchen werden die unveränderten ursprünglichen Merkmale mit der KLT transformiert. In weiteren Untersuchungen werden zunächst die dynamischen Merkmale für verschiedene Zeitauflösungen berechnet und danach ähnliche Experimente mit den statischen Merkmalen durchgeführt. Zuletzt werden Versuche mit Produkttermen unternommen. Auch die Begriffe nichtlineare PCA und Kernel-PCA werden dort erläutert.

Nach einem Ausblick in *Kapitel 6* schließt die Arbeit mit der Zusammenfassung (*Kapitel 7*).

Kapitel 2

Merkmalberechnung für die Spracherkennung

In diesem Kapitel werden Verfahren zur Berechnung von Merkmalen aus einem Sprachsignal vorgestellt. Jene sollen die wichtige Information des Gesprochenen enthalten, so dass damit eine korrekte Klassifikation von Lauten und Wörtern durchgeführt werden kann.

In Vorverarbeitungsschritten, auf die hier nicht näher eingegangen wird, wird zunächst das kontinuierliche Sprachsignal diskretisiert, d.h. in zeitlicher Richtung unter Beachtung des Abtasttheorems abgetastet und anschließend quantisiert. Nun ist das Sprachsignal für die Weiterverarbeitung in einem Digitalrechner geeignet und es können die Algorithmen der Merkmalextraktion angewandt werden.

Aus der zeitlichen Folge von Abtastwerten f_n ($n = 0, 1, 2, ...$) berechnet man eine Folge von Merkmalvektoren $\boldsymbol{c}_\tau$ ($\tau = 0, 1, 2, ...$). In die Berechnung eines jeden Merkmalvektors geht eine Vielzahl von Abtastwerten ein. Man erhält letztendlich wesentlich weniger Merkmalvektoren als Abtastwerte, so dass die Datenmenge reduziert wird. Wichtige Informationen werden dabei hervorgehoben und unwichtige wie z.B. Sprecher-Information, Sprechweise und Umgebungseinflüsse ausgeblendet [ST95, S.45]. Merkmalvektoren einer Klasse sollen im Raum einen kompakten Bereich einnehmen und möglichst getrennt von den Merkmalen anderer Klassen liegen [Nie83, S.11].

In den ersten beiden Abschnitten werden übliche grundlegende Vorgehensweisen der Merkmalberechnung vorgestellt. Die Vorgehensweise im Programm `fex3_1`, das am LME verwendet wird, wird im zweiten Abschnitt erläutert. Danach sind einige Experimente aus [Rie94] und [Fis88] zur Verbesserung der Merkmale zusammengestellt. Zuletzt werden neuere Ansätze aus der Literatur aufgelistet.

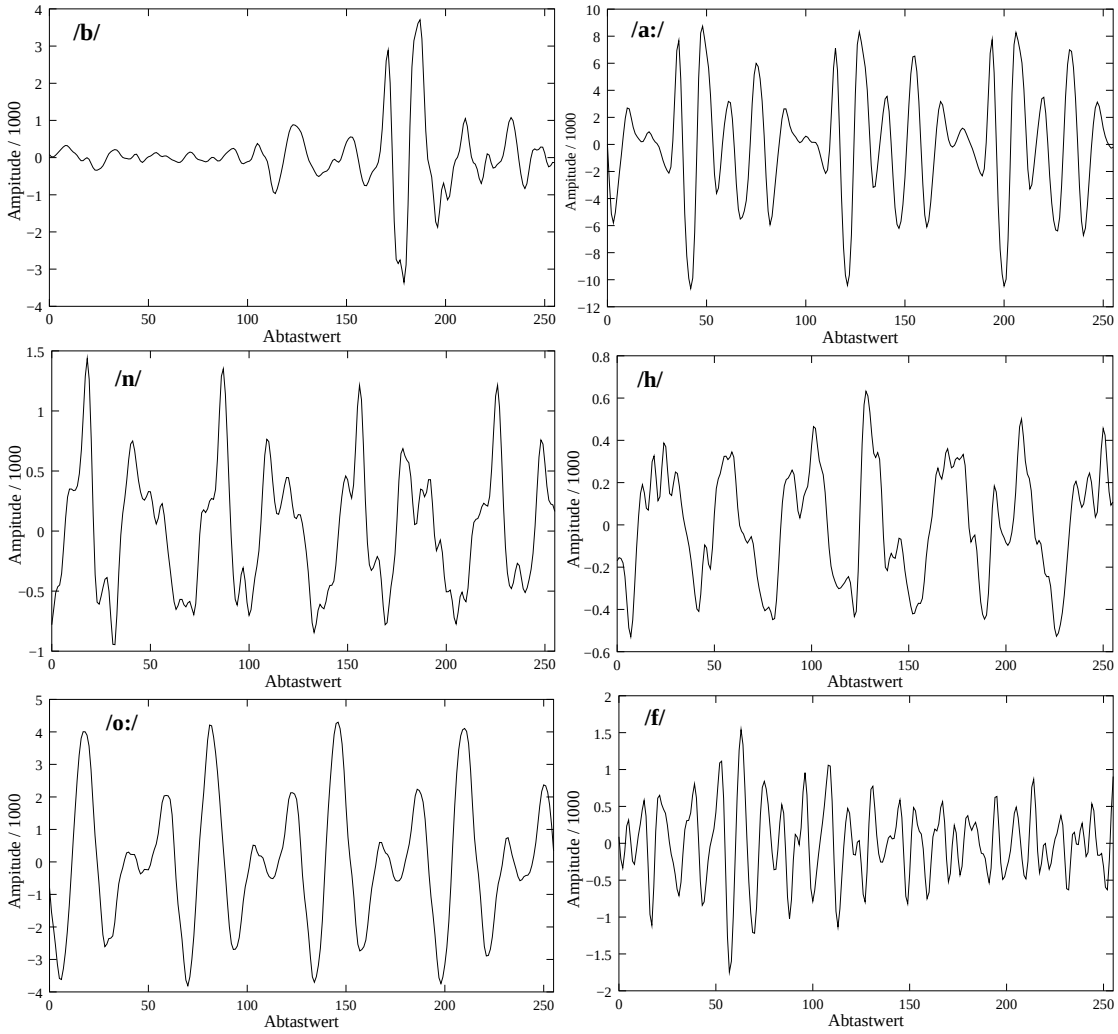

Bild 2.1: Ausschnitte aus dem Zeitsignal des Wortes “Bahnhof”. Achtung: unterschiedliche Skalierung der Amplituden-Achsen.

2.1 Statische Merkmale

In diesem ersten Abschnitt werden Merkmale beschrieben, die aus Zeitfenstern des Sprachsignals gewonnen werden und nur Informationen aus diesem jeweiligen Fenster wiedergeben. Man spricht von statischen Merkmalen, im Gegensatz zu den dynamischen, die im nächsten Abschnitt vorgestellt werden. Ausführlichere Erläuterungen hierzu findet man in [ST95, S.45 - 68].

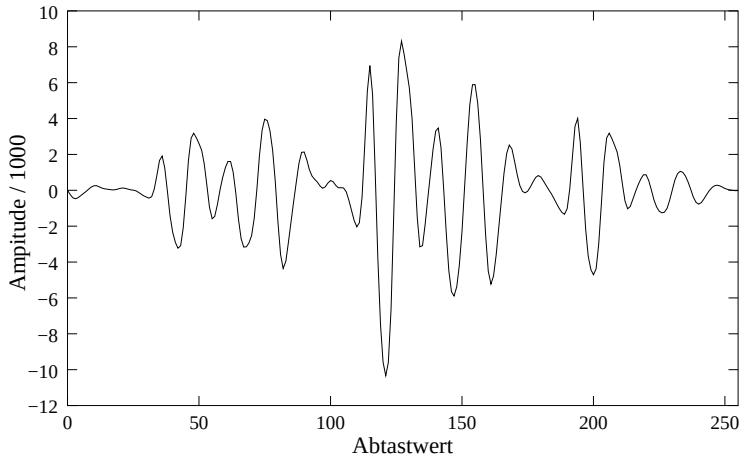

Bild 2.2: Das Kurzzeitanalysefenster des Lautes /a:/ aus dem Wort “Bahnhof”, ausgeschnitten mit einem Hamming-Fenster

2.1.1 Kurzzeitanalyse

Statt die gesamte Folge von Abtastwerten f_n ($n = 0, 1, 2, ...$) zu analysieren, zerlegt man diese erst in eine Folge kleiner evtl. auch überlappender Fenster. Bei der Kurzzeitanalyse wird also das Signal mit einer Fensterfunktion multipliziert bzw. beide Spektren gefaltet. Da ein Rechteckfenster steile Kanten und sein Spektrum folglich hohe Frequenzen besitzt, verschmiert es bei der Faltung das Spektrum des Sprachsignals stark. Deshalb wird oft vom Hamming-Fenster Gebrauch gemacht:

$$w_n = 0.54 - 0.46 \cos\left(\frac{2\pi n}{N-1}\right) \tag{2.1}$$

N ist die Anzahl der Abtastwerte im Fenster, für $n < 0$ bzw. $n \geq N$ gilt $w_n = 0$. Alternativen sind z.B. das Hanning-, Gauß- oder Parabel-Fenster [ST95, S.49].

Die Abbildung 2.1 zeigt einzelne Laute aus dem Wort “Bahnhof”, die mit einem Rechteckfenster aus dem Zeitsignal aus Abbildung 1.1 ausgeschnitten sind. Die Fensterbreite beträgt jeweils 256 Abtastwerte. Der Laut /b/ beginnt bei 30 ms, /a:/ bei 150 ms, /n/ bei 300 ms, /h/ bei 380 ms, /o:/ bei 500 ms und /f/ bei 650 ms. Abbildung 2.2 zeigt nochmal das Fenster mit dem Laut /a:/, jedoch diesmal mit einem Hamming-Fenster ausgeschnitten.

Aus jedem Kurzzeitanalysefenster werden mehrere statische Merkmale berechnet und in einem Vektor gespeichert. Sind die Fenster mit τ nummeriert, so erhält man eine Folge von Merkmalvektoren $\boldsymbol{c}_\tau$ ($\tau = 0, 1, 2, ...$).

2.1.2 Zeitbereichsmerkmale

In der Spracherkennung werden heute kaum mehr Merkmale verwendet, die direkt aus den Abtastwerten des Zeitsignals f_n ($n = 0, 1, 2, ...$) berechnet werden. Eine Ausnahme ist die Kurzzeitenergie. Sie ist ein Maß für die Energie in einem Kurzzeitanalysefenster. Das Fenster Nummer τ startet mit Abtastwert m.

$$E_\tau = \sum_{n=0}^{N-1} |f_{n+m}|^2 \tag{2.2}$$

Mit der Kurzzeitenergie lassen sich z.B. Sprache und Stille unterscheiden oder auch grob Silbengrenzen erkennen.

Andere Zeitbereichsmerkmale sind die Autokorrelationsfunktion, die verwendet werden kann, um Periodizitäten aufzudecken [ST95, S.52f], oder die Nulldurchgangsrate, die geeignet ist zwischen stimmhaften und stimmlosen Lauten zu unterscheiden.

2.1.3 Spektrum und Mel-Spektrum

Im folgenden werden die Frequenzanalyse eines Sprachsignals, sein Spektrum, und das Mel-Spektrum diskutiert.

Durch die Fourier-Transformation FT eines Zeitsignals erhält man sein Spektrum, das über seine Zusammensetzung aus Frequenzen Aufschluss gibt. Die Kurzzeitanalyse-Fenster des Signals werden einzeln transformiert, da man Änderungen der spektralen Eigenschaften von Laut zu Laut beobachten möchte.

Seien $f_{n,m} = f_{n+m} \cdot w_n$ ($n = 0, 1, ..., N-1$) die Abtastwerte im Fenster τ, das an der Stelle m beginnt, dann berechnet sich die Diskrete Fourier Transformation (DFT) nach der Formel

$$S_{k\tau} = \sum_{n=0}^{N-1} f_{n,m} \cdot e^{-\frac{2\pi i n k}{N}}; \tag{2.3}$$

$k = 0, ..., N-1$; N gibt die Anzahl der Abtastwerte im Fenster an.

Die Abbildung 2.3 zeigt die 6 Spektren der Laute aus Abbildung 2.1. Da es sich um Telefon-Signale handelt, sind keine Frequenzen über 4000 Hz vorhanden. Die Amplitude ist logarithmiert.

Nach dem Unschärfeprinzip gilt, dass sich die Frequenzauflösung umgekehrt proportional zur Zeitauflösung verbessert bzw. verschlechtert. Zum Erkennen kurzer Plosive wäre eine hohe Zeitauflösung wünschenswert, um Formanten zu unterscheiden jedoch eine hohe Frequenzauflösung. Die bei der Kurzzeitanalyse gewonnene zeitliche Folge von Spektren wird in Spektrogrammen dargestellt. Abbildung 2.4 zeigt oben ein Breitbandspektrogramm mit hoher, und

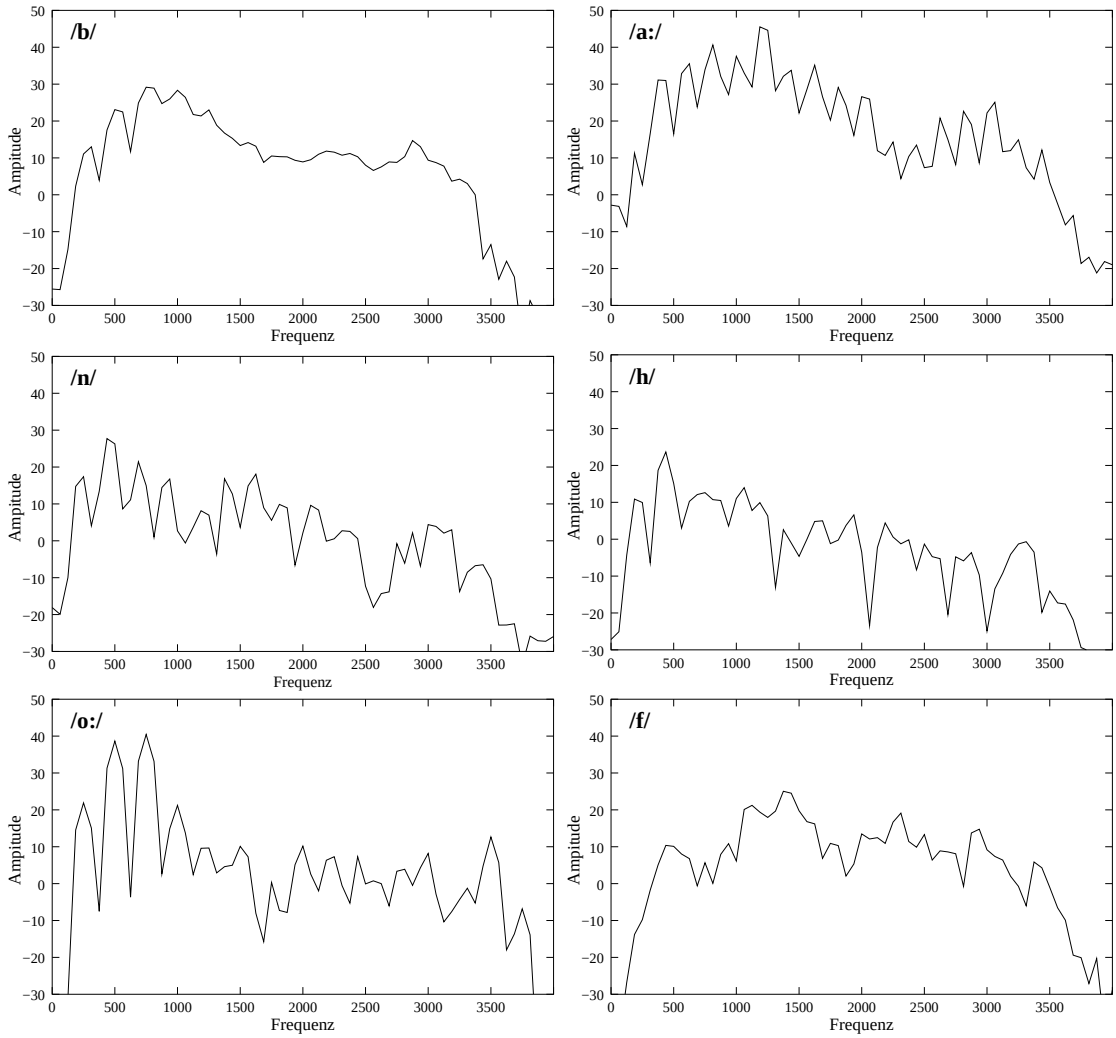

Bild 2.3: Die logarithmierten Spektren der Laute im Wort “Bahnhof”. Amplitude in Dezibel.

unten ein Schmalbandspektrogramm mit niedriger Frequenzauflösung für das Wort “Bahnhof”. Dort findet man auch die Spektren aus Abbildung 2.3 wieder. Exemplarisch wurde als /b/ dasjenige Zeitfenster im Spektrogramm ausgewählt, das bei 30 ms beginnt; /a:/ startet bei 150 ms, /n/ bei 300 ms, /h/ bei 380 ms, /o:/ bei 500 ms und /f/ bei 650 ms.

Nach dem Parsevalschen Theorem [Opp83, S.326 f.] und unter der Annahme der zeitlich periodischen Fortsetzung des Fensters Nummer τ, das mit Abtastwert m beginnt, lässt sich die Signalenergie aus dem letzten Unterabschnitt auch aus dem Spektrum berechnen.

$$E_\tau = \sum_{n=0}^{N-1} |f_{n+m}|^2 \approx \frac{1}{N} \sum_{k=1}^{N-1} |S_{k\tau}|^2 \tag{2.4}$$

Für reellwertige Funktionen, wie es Sprachsignale sind, gilt zudem $|S_{k\tau}|^2 = |S_{-k\tau}|^2$. Das

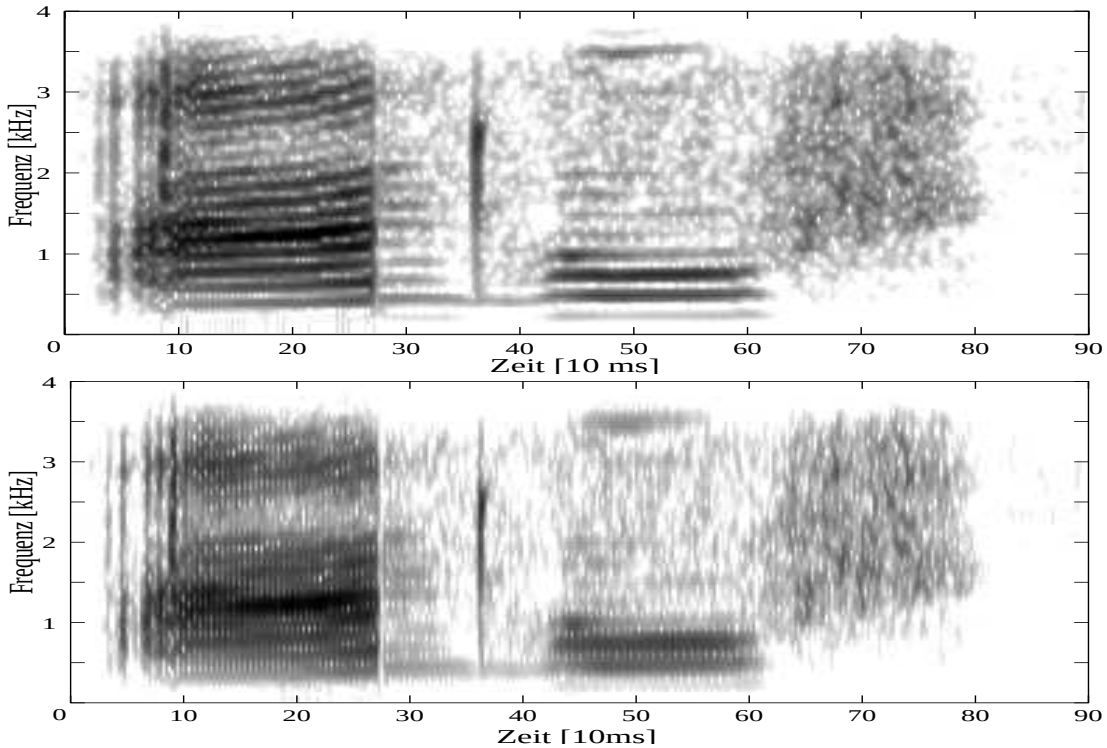

Bild 2.4: Spektrogramme des Wortes “Bahnhof”

Betragsquadratspektrum $|S_{k\tau}|^2$ wird also durch $N/2$ Koeffizienten pro Kurzzeitfenster beschrieben.

Aus dem Spektrum wird das Mel-Spektrum berechnet. Es ist eine spezielle Form des Band-Spektrums. Benötigt wird zunächst eine Bank von Bandpass-Filtern, die wichtige Teile des Spektrums abdecken. Durch Multiplikation der Bandpass-Filter mit dem Spektrum wird dieses in Fenster aufgeteilt. Die Bandpass-Energie wird wie in Gleichung 2.4 durch Summieren der Koeffizienten eines Bandes berechnet. Die Energiewerte der Bänder bilden das Bandspektrum.

Als Bandpass-Filter werden üblicherweise Dreiecks-, Trapez- oder Rechteckfilter verwendet. Eine Filterbank mit trapezförmigen Filtern zeigt Abbildung 2.5. Wenn diese Filter auf der Mel-Skala äquidistant liegen erhält man das Mel-Spektrum. Auf diese Weise erfolgt eine Kompression des Spektrums, die sich an der menschlichen Tonhöhenempfindung orientiert. In [Bur01] wird datengetrieben eine optimale Filterbank unter der Annahme der Normalverteilung der Laute berechnet, die mit der Mel-Filterbank sehr ähnlich ist und infolgedessen diesen psychoakustischen Ansatz bestätigt. Da in dieser Arbeit nur mit Sprachdaten in Telefonqualität gearbeitet wird, sind die beiden Bandpass-Filter im Bereich über 4kHz ungeeignet um wichtige Informationen auszufiltern. Die Filterbank aus Bild 2.5 wird jedoch bisher im Programm `fex3_1` (siehe Abschnitt 2.3), das am Lehrstuhl für Mustererkennung (LME) der Universität Erlangen-Nürnberg

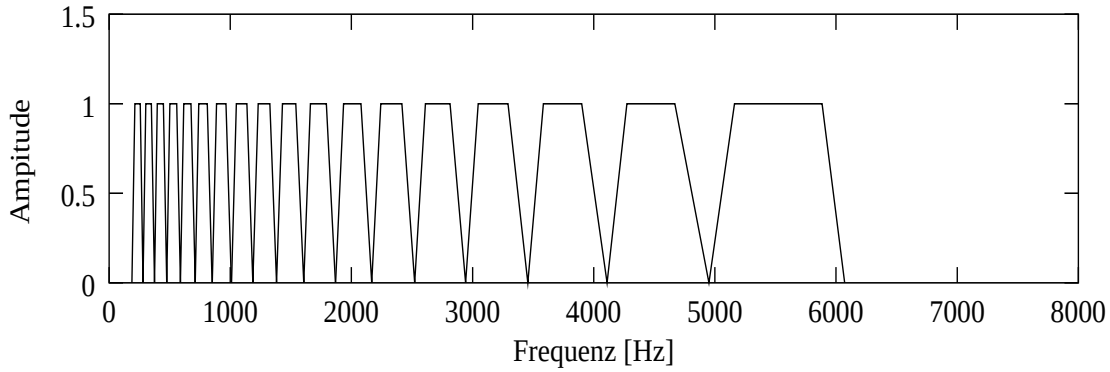

Bild 2.5: Mel-Filterbank mit 18 trapezförmigen Bänken

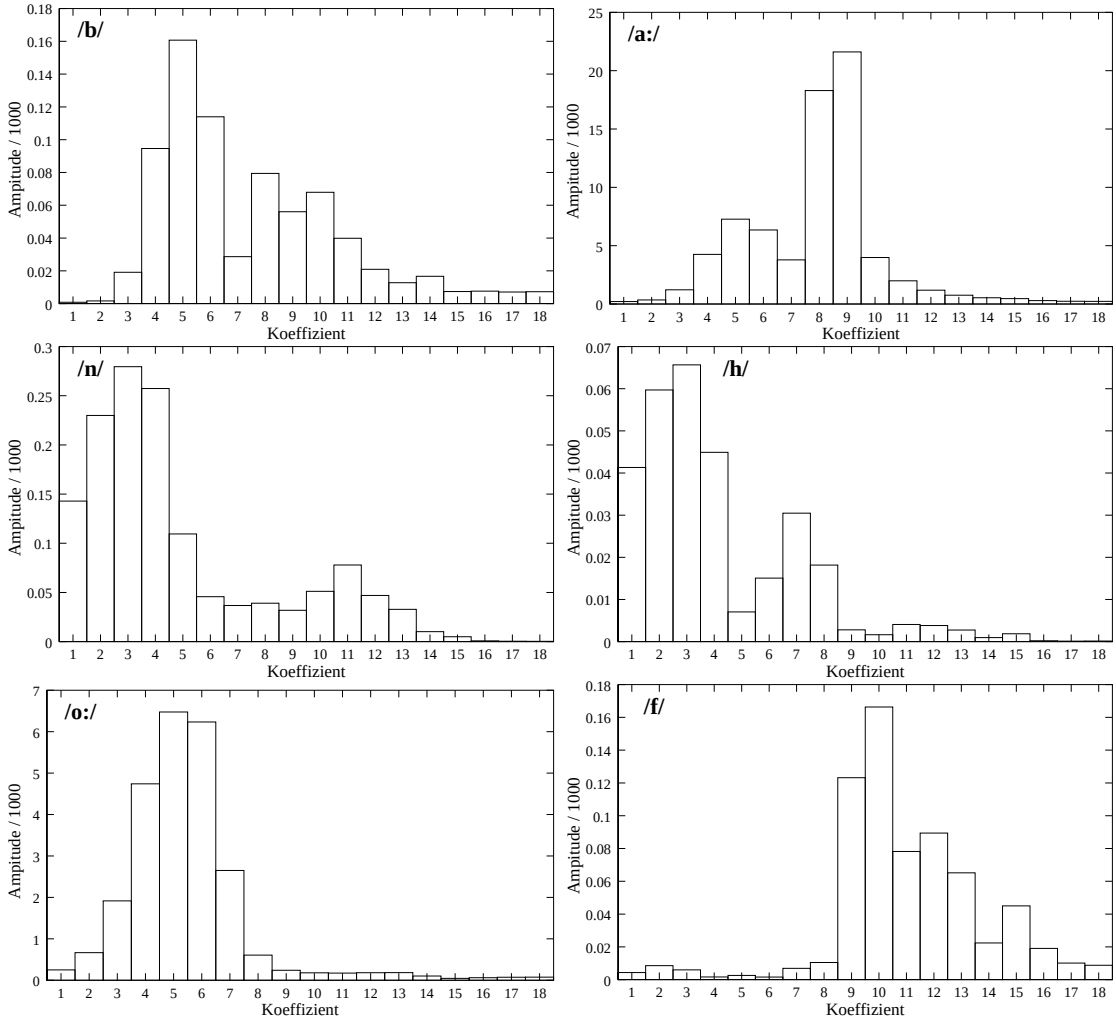

Bild 2.6: Die Mel-Spektrum Koeffizienten der Laute im Wort “Bahnhof”. Man beachte die Unterschiedliche Skalierung der Amplituden-Achsen.

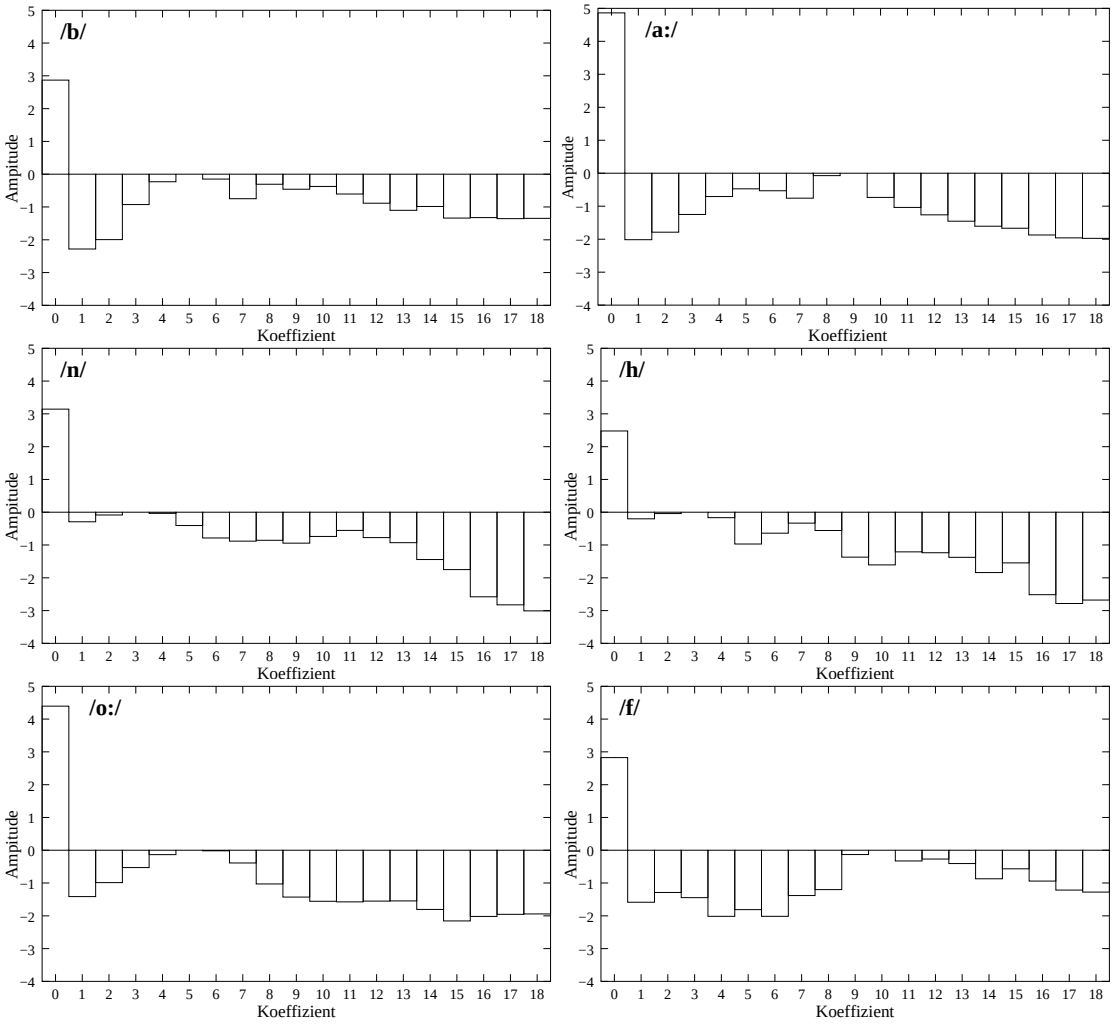

Bild 2.7: Die logarithmierten Mel-Spektrum Koeffizienten der Laute im Wort “Bahnhof”. Merkmal 0 ist die logarithmierte Gesamtenergie.

zur Merkmalberechnung verwendet wird, eingesetzt. Um die berechneten Merkmale direkt vergleichen zu können, wird diese Filterbank jedoch in dieser Arbeit beibehalten. Abbildung 2.6 zeigt die 18 Mel-Spektrum-Koeffizienten der Laute aus Abbildung 2.1.

Nach dem Source-Filter-Modell [ST95, S.33] ist das Sprachsignal mit Abtastwerten f_n aus Faltung des Anregungssignals e_n und der Vokaltraktresonanz h_n entstanden. Das logarithmierte Spektrum ist demnach eine additive Überlagerung der logarithmierten Spektren von e_n und h_n. Sei FT die Fourier-Transformation, so gilt:

$$FT(f_n) = FT(e_n) \cdot FT(h_n) \tag{2.5}$$

$$\log FT(f_n) = \log FT(e_n) + \log FT(h_n) \tag{2.6}$$

Das Spektrum wird logarithmiert, um die Lautheits-Empfindung des Menschen zu modellieren. Für Abbildung 2.7 wurden die 18 Mel-Spektrum Koeffizienten auf das Intervall [0,1] skaliert und danach logarithmiert. Koeffizient 0 ist die logarithmierte Gesamtenergie.

2.1.4 Das Cepstrum

Das Spektrum einer Funktion f wird normalerweise nicht selbst zur Berechnung von Merkmalen verwendet. Stattdessen berechnet man das Cepstrum.

$$C = FT^{-1}\left(\log|FT(f)|\right) \tag{2.7}$$

Die Einheit im Cepstrum ist die Quefrenz, analog zur Frequenz im Spektrum. Mit dem Cepstrum analysiert man also die Quefrenzen des Spektrum, so wie man mit dem Spektrum die Frequenzen des Zeitsignals berechnet. Analog zur Filterung des Zeitsignals spricht man von der Lifterung des Spektrums.

In der Praxis berechnet man die inverse DFT oder die Kosinustransformation der Ordnung L. Letztere ist für das Fenster τ

$$C_{k\tau} = \sum_{l=1}^{L} \log|S_{l\tau}| \cdot \cos\left(\frac{k \cdot (2l-1)\pi}{2L}\right). \tag{2.8}$$

$S_{l\tau}$ sind die Koeffizienten des Spektrums. Die Kosinustransformation lässt sich als eine Hauptachsentransformation interpretieren, welche die Merkmale dekorreliert. Als Mel-Cepstrum bezeichnet man die Kosinustransformation des logarithmierten Mel-Spektrums. Der nullte Koeffizient ist die Kurzzeitenergie (die Faktoren $\cos(...)$ in Gleichung 2.8 sind alle eins) und wird oft durch andere Energiemaße ersetzt.

Die Abbildung 2.8 zeigt die logarithmierte Energie und 11 Mel-Cepstrum-Koeffizienten der Laute aus Abbildung 2.1. Letztere werden kurz als MFCCs (Mel-frequency cepstral coefficients) bezeichnet.

In das Spektrum gehen die Frequenzen des Anregungssignals und die Artikulation durch den Vokaltrakt ein. Das geglättete Spektrum gibt die Charakteristika des Vokaltraktes wieder. Die hohen Quefrenzen des Spektrums resultieren also vom Anregungssignal bei der Spracherzeugung. Das Spektrum kann geglättet, also Tiefpaß-geliftert, werden, indem man die hohen Quefrenzen im Cepstrum abschneidet. Das resultierende Spektrum zeigt deutlich Maxima bei den Vokaltraktresonanzen. Da man an den Eigenschaften des Vokaltraktes interessiert ist werden nur die Mel-Cepstrum Koeffizienten $C_{k\tau}$ für kleine k berechnet. In Abbildung 2.8 ist $k < 12$.

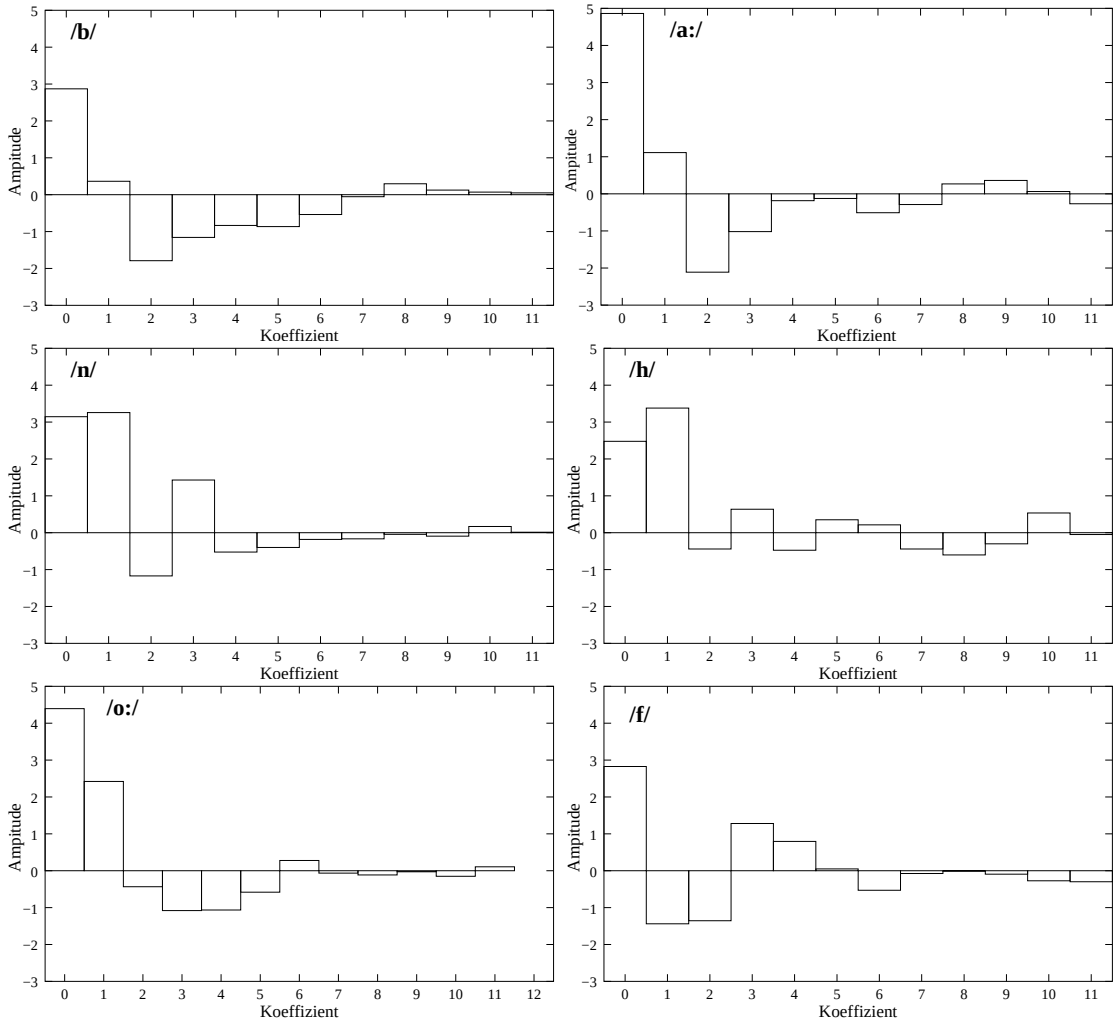

Bild 2.8: Die MFCCs der Laute im Wort “Bahnhof”. Merkmal 0 ist die logarithmierte Gesamtenergie

2.1.5 Lineare Vorhersage und Modellspektrum

Die linearen Vorhersage [Nie83, S.97ff] ist eine Alternative zur Cepstralanalyse. Dabei wird ein Schätzwert für den n-ten Abtastwert als Linearkombination der K vorangegangenen Abtastwerte berechnet:

$$\hat{f}_n = -\sum_{k=1}^{K} a_k \cdot f_{n-k} \tag{2.9}$$

Der Fehler ϵ innerhalb eines Kurzzeitanalysefensters, das mit Abtastwert m beginnt, ist so definiert:

$$\epsilon = \sum_{n=m}^{m+N-1} (f_n - \hat{f}_n)^2 \tag{2.10}$$

Die a_k lassen sich bestimmen, indem man ϵ minimiert. Als Merkmale geeignet sind nun z.B. die a_k selbst, manchmal auch zusammen mit ϵ.

Das Modellspektrum berechnet man so: Sei f_A die Abtastfrequenz des Zeitsignals und f_R die gewünschte Frequenzauflösung im Modellspektrum, so ergänzt man für jedes Kurzzeitanalysefenster τ den folgenden Vektor $\boldsymbol{a}$ mit Nullen zu einem mindestens (f_A/f_R)-dimensionalen Vektor:

$$\boldsymbol{a}_\tau = (1, a_1, ..., a_K, 0, ..., 0) \tag{2.11}$$

Durch Diskrete Fourier Transformation dieses Vektors erhält man das Modellspektrum, das bei geeignet gewählter Ordnung K wie das gelifterte Spektrum aus Abschnitt 2.1.4 geglättet ist und die Charakteristika des Vokaltraktes wiedergibt. Es wird etwa $K = f_A + 4$ empfohlen.

2.2 Dynamische Merkmale

Das menschliche Gehör berücksichtigt bei der Wahrnehmung von Sprache auch deren zeitlichen Verlauf. So sind Plosive z.B durch ihr abruptes Zeitverhalten charakterisiert. Dynamische Merkmale sind solche Merkmale, die den zeitlichen Kontext berücksichtigen. Ein sehr simples Verfahren zur Kontextberücksichtigung ist z.B. die Konkatenation jedes Vektors mit seinen Nachbarn. In den folgenden Abschnitten werden komplexere Verfahren beschrieben [ST95, S.68ff].

2.2.1 Ableitung

Die Ableitung $\delta x_{k,\tau}$ einer Merkmalkomponente x_k im Kurzzeitanalysefensters τ nach der Zeit lässt sich näherungsweise durch Differenz der Merkmale $\boldsymbol{c}_{\tau\pm i}$ für festes i berechnen. Eine bessere Approximation ist die Regressionsgerade über $2M+1$ Kurzzeitfenster, also M in jeder zeitlichen Richtung:

$$\delta x_{k,\tau} = \frac{\sum_{m=-M}^{M} m \cdot x_{k,\tau+m}}{\sum_{m=-M}^{M} m^2} \tag{2.12}$$

Verdeutlicht sind Differenz und Regression in Abbildung 2.9. Die Regressionskoeffizienten $y_{k,\tau}$, die dem Zähler aus Gleichung 2.12 entsprechen, lassen sich auch rekursiv berechnen:

$$y_{k,\tau} = y_{k,\tau-1} + M(x_{k,\tau+M} - x_{k,\tau-M-1}) - z_{k,\tau-1} \tag{2.13}$$

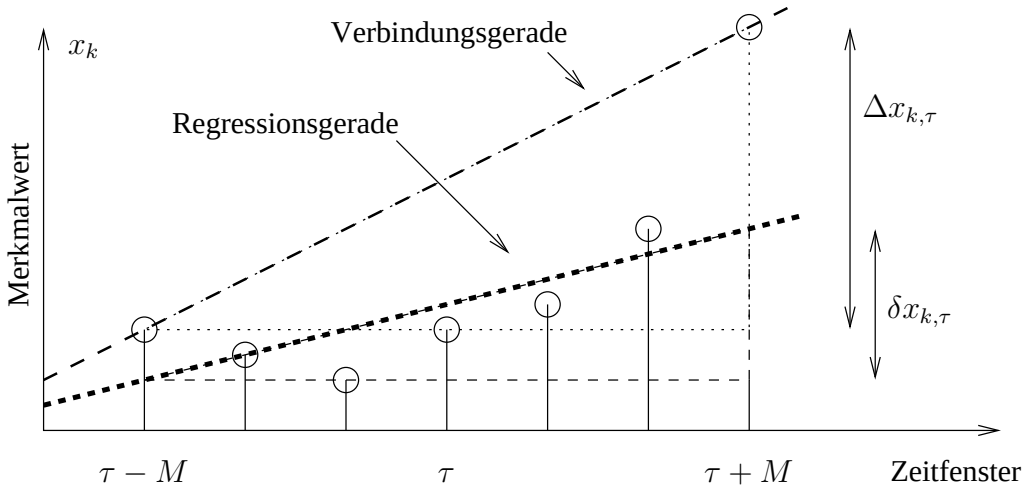

Bild 2.9: Regressionsgerade und Verbindungsgerade aus [ST95, S.70].

$$z_{k,\tau} = z_{k,\tau-1} + (x_{k,\tau+M} - x_{k,\tau-M}) \tag{2.14}$$

Initialisiert wird $y_{k,\tau} = z_{k,\tau} = 0$. Ableitungen höherer Ordnung erhält man durch Berechnung von Regressionen höherer Ordnung, also durch wiederholtes Berechnen der Regression einer Ableitung.

2.2.2 Zweidimensionales Cepstrum

Das Cepstrum wird durch eine inverse Fourier Transformation aus dem logarithmierten Spektrum für jedes Zeitfenster berechnet. Betrachtet man eine zeitliche Folge solcher Cepstra, so erhält man eine Matrix, auf die sich nun auch in Zeitrichtung die Fourier-Transformation anwenden lässt. Die resultierende Matrix wird als 2D-Cepstrum bezeichnet.

In [ST95, S.72f] heißt es, dass diese Merkmale bei der Einzelworterkennung gute Ergebnisse liefern, bei Experimenten mit kontinuierlicher Sprache sich aber die Steigungsmerkmale überlegen zeigten.

2.3 Merkmalberechnung in `fex3_1`

In diesem Abschnitt wird beschrieben, welche Merkmale am Lehrstuhl für Mustererkennung (LME) der Universität Erlangen-Nürnberg für die automatische Spracherkennung verwendet werden. Die Berechnungen sind im Programm `fex3_1` (“Feature Extraction”) implementiert.

Das Sprachsignal wird mit einer Frequenz $f_a = 16$ kHz abgetastet. Bei der Kurzzeitanalyse betrachtet man nun Fenster im Signal, die alle 10 ms berechnet werden und eine Breite von 16 ms haben, was $N = 256$ Abtastwerten entspricht. Aus jedem dieser Fenster wird ein 24-dimensionaler Merkmalvektor berechnet, wie unten noch ausführlich erläutert wird. Die ersten 12 Komponenten sind statische Merkmale, nämlich die Gesamtlautheit und 11 Mel-Cepstrum Merkmale, die restlichen 12 Komponenten sind Ableitungen der 12 statischen Merkmale.

Wie Mel-Spektrum, Gesamtenergie, Mel-Cepstrum und Ableitungen berechnet werden, wird in den nächsten Unterabschnitten detailliert beschrieben. Alle Informationen hierzu wurden aus [Rie94, Anhang A] sowie aus dem Quellcode von `fex3_1` bezogen.

2.3.1 Berechnung des Mel-Spektrums

In diesem Abschnitt werden Formeln und Zahlen zur Berechnung des Mel-Spektrums angegeben. Zunächst wird das Kurzzeitspektrum berechnet und danach werden Frequenzbänder zum Mel-Spektrum zusammengefasst. Motiviert wurde das Vorgehen oben im Abschnitt 2.1.3.

1. Bei der *Kurzzeitanalyse* beginnt das Fenster Nummer τ zum Zeitpunkt

   $t_\tau = \tau \cdot T_f + t_0; \quad \tau = 0, 1, 2, ...$

   Hierbei ist $T_f = 10$ ms die Fortschaltzeit und t_0 die Startzeit des Zeitsignals. Letztere ist o.B.d.A. ab jetzt immer Null. Bei einer Abtastperiode $T_a = 1/f_a = 62.5\mu$s entspricht t_τ dem Abtastwert Nummer

   $z_\tau = \tau \cdot T_f/T_a + t_0/T_a = \tau \cdot 160 + t_0/T_a$.

2. Seien nun f_i die Abtastwerte des Signals und w_j ($j = 0, ...N-1$) die Abtastwerte des Hamming-Fensters. Die Berechnung des *Kurzzeitspektrums* erfolgt dann mit der schnellen Hartley Transformation (FHT):

   $$S_{k\tau} = \sum_{n=0}^{N-1} w_n f_{z_\tau+n} \left[\cos\left(\frac{2\pi nk}{N}\right) + \sin\left(\frac{2\pi nk}{N}\right)\right]; \quad k = 0, ..., N-1. \tag{2.15}$$

   Jedes Fenster τ hat $N = 256$ Abtastwerte. Die FHT wird der Fourier-Transformation vergezogen, da Sprachdaten reellwertig sind, und so keine Berechnungen im Komplexen notwendig sind.

3. Das Spektrum wird in $L = 18$ Frequenzbänder durch Multiplikation mit trapezförmigen Fenstern d_{lk} (siehe Abbildung 2.5) zusammengefasst; man erhält so das *Mel-Spektrum*

$$P_{l\tau} = \sum_{k=0}^{N/2-1} d_{lk} \cdot |S_{k\tau}|^2; \quad l = 1, ..., L. \tag{2.16}$$

2.3.2 Berechnung der Gesamtenergie

Die Summe der Mel-Spektrum Bandenergien ist die Gesamtenergie. Sie wird normiert, logarithmiert, zeitlich geglättet und gefiltert.

1. Die *Gesamtenergie* im Fenster τ ist die Summe aller Mel-Spektrum Koeffizienten.

$$P_{0\tau} = \sum_{l=1}^{L} P_{l\tau}; \quad L = 18 \tag{2.17}$$

2. Damit die Energie im Intervall $[\epsilon, +\infty]$ liegt, werden sehr kleine Werte auf $\epsilon = 1.0 \cdot 10^{-6}$ erhöht.

$$\tilde{P}_{0\tau} = \begin{cases} P_{0\tau} & \text{falls} \quad P_{0\tau} > \epsilon \\ \epsilon & \text{sonst} \end{cases} \tag{2.18}$$

3. Nun kann die Energie logarithmiert werden. Alle Werte sind größer als $\log \epsilon$.

$$L_{0\tau} = \log \tilde{P}_{0\tau} \tag{2.19}$$

4. Die Gesamtlautheit, die ja zum Satzende hin abnimmt, soll auf gleichem Niveau gehalten werden. Daher wird sie zeitlich geglättet.

   Durch Dilatation erhält man zunächst eine Kontur der Maxima:

$$L_{0\tau}^{(max)} = \max(L_{0,\tau-Z}, L_{0,\tau-Z+1}, ..., L_{0,\tau}) \tag{2.20}$$

   Experimentell wurde $Z = 33$ festgesetzt. Kurze Einbrüche unter diesen Maxima werden mit einem Medianfilter geglättet:

$$L_{0\tau}^{(med)} = median(L_{0,\tau-Z}^{(max)}, L_{0,\tau-Z+1}^{(max)}, ..., L_{0,\tau}^{(max)}) \tag{2.21}$$

   Die normierte Energie erhält man, indem man den aktuellen Wert von der Maximalkontur subtrahiert.

$$\hat{L}_{0\tau} = L_{0\tau}^{(med)} - L_{0\tau} \tag{2.22}$$

5. Der zeitliche Verlauf der Gesamtenergie $L_{0\tau}$ bzw. $\hat{L}_{0\tau}$ wird zusätzlich mit einem Hut-Filter ("Tiroler Hut") geglättet:

$$E_\tau = \frac{1}{4}L_{0,\tau-1} + \frac{1}{2}L_{0,\tau} + \frac{1}{4}L_{0,\tau+1} \tag{2.23}$$

$$\hat{E}_\tau = \frac{1}{4}\hat{L}_{0,\tau-1} + \frac{1}{2}\hat{L}_{0,\tau} + \frac{1}{4}\hat{L}_{0,\tau+1} \tag{2.24}$$

Als Merkmal, das die Gesamtenergie wiedergibt, können die Ergebnisse bzw. Zwischenergebnisse aus Punkt 3, 4 oder 5 verwendet werden. In der Implementierung zur Merkmalextraktion (`fex3_1`) hat man sich für die zeitlich geglättete Version $\hat{E}_\tau$ als erste Komponente des Merkmalvektors entschieden, da damit die besten Ergebnisse bei der Klassifikation erzielt werden konnten.

2.3.3 Berechnung der Mel-Cepstrum Koeffizienten

Nach Normierung und Logarithmierung der Mel-Spektrum Koeffizienten aus Abschnitt 2.3.1 werden daraus die 11 Mel-Cepstrum Koeffizienten berechnet. Diese stellen die Komponenten 2 bis 12 des 24-dimensionalen Merkmalvektors dar.

1. Zunächst werden die Bandenergien auf das Intervall $[\epsilon, 1.0]$ normiert ($\epsilon > 0$).

$$P_{max,\tau} := \max_l(P_{l\tau})$$

$$\tilde{P}_{l\tau} = \begin{cases} \frac{P_{l\tau}}{P_{max,\tau}} & \text{falls} \quad P_{l\tau} > \epsilon \\ \epsilon & \text{sonst} \end{cases} ; \quad l = 1, ..., L; L = 18. \tag{2.25}$$

   Es wird $\epsilon = 1.0 \cdot 10^{-6}$ gesetzt.

2. Die normierten Mel-Spektrum Koeffizienten werden logarithmiert und somit auf das Intervall $[\log\epsilon, 0]$ abgebildet.

$$L_{l\tau} = \log\tilde{P}_{l\tau}; \quad l = 1, ..., L; L = 18. \tag{2.26}$$

3. Das *Mel-Cepstrum* wird nun mit der Diskreten Kosinustransformation berechnet.

$$C_{k\tau} = \sum_{l=1}^{L} L_{l\tau} \cdot \cos\left(\frac{k \cdot (2l-1)\pi}{2L}\right); \quad k = 1, ..., 11; L = 18. \tag{2.27}$$

4. Frequenzverzerrungen entstehen durch spezifische Kanaleigenschaften aber auch durch die Raumakustik oder Unterschiede in der Physiologie des Vokaltraktes verschiedener Sprecher. Diese Störungen lassen sich durch die *dynamisch adaptive cepstrale Subtraktion (DACS)* eliminieren [Rie94, S.82 ff]. Von jedem Cepstrum-Koeffizienten $C_{k\tau}$ wird der Mittelwert $\bar{C}_{kn}$ subtrahiert.

$$\tilde{C}_{k\tau} = C_{k\tau} - \bar{C}_{kn}; \quad k = 1, ..., 11 \tag{2.28}$$

In die Berechnung von $\bar{C}_{kn}$ gehen alle Kurzzeitanalyse-Fenster ein, die als Sprache klassifiziert wurden, d.h. deren Energie einen Schwellwert θ überschreitet. Die Anzahl dieser für die Mittelwertbildung berücksichtigten Zeitscheiben gibt der Index $n < \tau$ an. Der Startwert für den Mittelwert $\bar{C}_{k0}$ wird dem Programm `fex3_1` als Argument übergeben. Dann wird er mit jedem Fenster τ aufgefrischt:

$$\bar{C}_{k,n+1} = \begin{cases} (1-\alpha_n) \cdot \bar{C}_{kn} + \alpha_n \cdot \tilde{C}_{k\tau} & \text{falls} \quad L_{0\tau}^{(med)} > \theta : \\ & \text{Zeitscheibe ist Sprache} \\ \bar{C}_{k,n} & \text{sonst} \end{cases} \tag{2.29}$$

$k = 1, ..., 11$

mit

$$\alpha_n = \begin{cases} 1/N_{min} & \text{falls} \quad n < N_{min} \\ 1/n & \text{falls} \quad N_{min} \leq n \leq N_{max} \\ 1/N_{max} & \text{falls} \quad n > N_{max} \end{cases}$$

Aufgrund von experimentellen Untersuchungen wird $N_{min} = 500$, $N_{max} = 2000$ und $\theta = -4.8$ empfohlen.

5. Der zeitliche Verlauf der Mel-Cepstrum Merkmale wird mit einem Hut-Filter (“Tiroler Hut”) geglättet:

$$\hat{C}_{k\tau} = \frac{1}{4}\tilde{C}_{k,\tau-1} + \frac{1}{2}\tilde{C}_{k\tau} + \frac{1}{4}\tilde{C}_{k,\tau+1} \tag{2.30}$$

DACS und Tirol-Filterung sind im Programm `fex3_1` fest implementiert, können jedoch durch minimale Änderung des Quellcodes ausgeschaltet werden..

2.3.4 Berechnung der Ableitungen

Als dynamische Merkmale werden die ersten Ableitungen der Gesamtenergie und der 11 Mel-Cepstrum Koeffizienten verwendet. Sie bilden die Komponenten 13 bis 24 des 24-dimensionalen Merkmalvektors.

Die ersten Ableitungen der Mel-Cepstrum Koeffizienten $\hat{C}_{k\tau}$ werden durch die *Regressionsgeraden* approximiert. Diese werden über 9 Kurzzeitanalyse-Fenster berechnet, also dem aktuellem mit Startzeit t_τ und den jeweils $M = 4$ vorangegangenen bzw. folgenden:

$$\hat{C}'_{k\tau} = \frac{\sum_{m=-M}^{M} m \cdot \hat{C}_{k,\tau+m}}{\sum_{m=-M}^{M} m^2}; \quad k = 1, ..., 11. \tag{2.31}$$

Es wird also ein Kontext von 90 ms berücksichtigt, was etwa der Länge eines Phonems entspricht. Die Regressionen E'_τ und $\hat{E}'_\tau$ der Gesamtenergie werden analog berechnet.

2.3.5 Zusammenstellung des Merkmalvektors

Im Programm `fex3_1` zur Merkmalberechnung werden zuerst wie oben beschrieben folgende Merkmale berechnet:

$$\tilde{\boldsymbol{c}}_\tau = (\hat{E}_\tau, E_\tau, \hat{C}_{1\tau}, ..., \hat{C}_{11\tau}, \hat{E}'_\tau, E'_\tau, \hat{C}'_{1\tau}, ..., \hat{C}'_{11\tau})^T \tag{2.32}$$

Durch Weglassen des Merkmales 2 und dessen Ableitung erhält man letztendlich diesen 24-dimensionalen Merkmalvektor:

$$\boldsymbol{c}_\tau = (\hat{E}_\tau, \hat{C}_{1\tau}, ..., \hat{C}_{11\tau}, \hat{E}'_\tau, \hat{C}'_{1\tau}, ..., \hat{C}'_{11\tau})^T \tag{2.33}$$

2.4 Experimente zur Merkmalberechnung am LME

Aus den zahlreichen Experimenten, die am Lehrstuhl für Mustererkennung (LME) der Universität Erlangen-Nürnberg unternommen wurden und zum Ziel hatten, bessere Merkmale zu finden, werden nun einige ausgewählte von S. Rieck bzw. V. Fischer ([Rie94] bzw.[Fis88]) vorgestellt. Es handelt sich hier um Ansätze, die teils in die im letzten Abschnitt beschriebene Merkmalberechnung mit eingeflossen sind, teils aber auch wieder verworfen wurden.

2.4.1 Experimente zur Verbesserung der statischen Merkmale

In [Rie94, S.121 ff] beschreibt S. Rieck u.a. verschiedene Experimente die zur Verbesserung der statischen Merkmale dienen sollten. Eine Auswahl dieser Experimente ist im folgenden zusam-

Filterform	Fensterlänge	Erkennungsrate
32 Gruppen Dreieck	10 ms	57.50 %
	16 ms	57.97 %
18 Gruppen Trapez	10 ms	56.80 %
	16 ms	57.53 %

Tabelle 2.1: Erkennungsraten für verschiedene Zeitanalysefenster und Filterbänke aus [Rie94]

mengestellt. Danach werden Versuche von V. Fischer beschrieben.

Experimentiert wird in [Rie94] u.a. mit der Mel-Filterbank. Untersuchungen, welche Fensterform hier vorzuziehen sei, geben den trapezförmigen Gewichtungen, die im vorigen Abschnitt beschrieben wurden, einen geringen Vorteil gegenüber den dreiecksförmigen.

Im nächsten Experiment wird die Größe der Filterbank und die Ausdehnung des Kurzzeitanalyse-Fensters variiert. Die Ergebnisse sind in der Tabelle 2.1 zusammengefasst. Die Erkennungsraten dort sind das Verhältnis der korrekt erkannten Merkmalvektoren zur Gesamtzahl der zu klassifizierenden Vektoren [Rie94, S.116]. Eine Filterbank mit 32 Frequenzgruppen statt der verwendeten 18 würde demnach zu verbesserten Erkennungsraten führen. Warum diese Verbesserungen in `fex3_1` nicht implementiert sind, konnte nicht in Erfahrung gebracht werden. Neuere Experimente zur Filterbankoptimierung aus [Psu01a] sind in Abschnitt 2.5 angegeben. Da eine grobere Zeitauflösung nach dem Unschärfeprinzip zu einer feineren Frequenzauflösung führt, ist das Fenster mit 16 ms Breite (wie in Abschnitt 2.3 verwendet) dem mit 10 ms vorzuziehen.

V. Fischer hat in [Fis88] Länge und Fortschaltzeit der Analysefenster variiert. Dabei hat er separat Erkennungsraten für verschiedene Lautoberklassen berechnet und analysiert.

Bei gleicher Fensterlänge und Fortschaltzeit von 12,8 ms fällt eine schlechte Erkennungsrate für Plosive auf. Erklärt wird dies u.a. damit, dass das Fenster oft schon Anteile des folgenden stimmhaften Lautes enthält.

Abhilfe sollten die kürzere Fensterlänge und Fortschaltzeit von 6,4 ms schaffen. Die Erkennungsrate für Plosive verschlechtert sich jedoch weiter, was darauf zurückgeführt wird, dass die Stichprobe für 12,8 ms lange Fenster handgelabelt war, der für die Erkennung wichtige Burst aber tatsächlich häufig nur in einem der beiden 6,4 ms langen Teilfenstern liegt.

Mit einer Fensterlänge von 25,6 ms und einer Fortschaltzeit von 12,8 ms soll eine bessere Frequenzauflösung erzielt werden und zusätzlich durch die Überlappung der Fenster solche Muster besser erkannt werden, für die die Formantübergänge wichtig sind. Die Erkennung der Plosive wird so besser, da wohl wichtige Information aus der Lautumgebung mehr ausgenutzt

werden kann. Auch wird bei doppelt so großen Fenstern der Burst öfter in der Mitte liegen und so durch das Hammingfenster verstärkt werden, während er bei kürzerer Fensterlänge häufig zwischen zwei Fenstern liegt und stark abgeschwächt wird.

In verschiedenen Experimenten zu dieser Fensterlänge werden Vokale bei männlichen Sprechern besser erkannt als bei einer Sprecherin. Dieses Phänomen wird mit den geschlechtsspezifischen Grundfrequenzen erklärt. Bei größerem Analysefenster können mehr von den längeren Perioden der männlichen Stimme analysiert werden.

In einem letzten Versuch wird die in den obigen Experimenten beste Fensterlänge von 12,8 ms und eine Fortschaltzeit von 3,2 ms gewählt. Durch stärkere Überlappung soll der Unterdrückung von Information am Fensterrand durch das Hammingfenster entgegengewirkt werden. Die Erkennungsraten verschlechtern sich aber gegenüber dem ersten Versuch, da zum einen wieder Probleme mit der im 12,8 ms handgelabelten Stichprobe entstehen und zum anderen nun das Lautstärke-Merkmal stärker gestreut ist und deshalb weniger zur Unterscheidung der Klassen beiträgt.

2.4.2 Experimente zur Verbesserung der dynamischen Merkmale

S. Rieck hat in [Rie94, S.129 ff] auch verschiedene Möglichkeiten untersucht, um die dynamischen Merkmale, die Informationen über den zeitlichen Verlauf der statischen Merkmale beinhalten, zu verbessern. Angaben zur Kontextlängen beziehen sich im folgenden immer auf ein Fenster, das die aktuelle Zeitscheibe und auf beiden Seiten symmetrisch weitere Zeitscheiben umfasst.

Einfachste Möglichkeit der Kontextberücksichtigung im Fenster τ ist die Differenz der beiden Vektoren $\boldsymbol{c}_{\tau\pm i}$. Ein Optimum in der Erkennungsrate wurde für $i = 1$ gefunden, was einem Kontext von 30 ms entspricht.

Bessere Ergebnisse werden jedoch bei der Approximation der Ableitung durch Regressionsgeraden erzielt. Begründet wird dies mit dem glatteren Verlauf dieser Merkmale. Das beste Ergebnis wird bei einem Kontext von 90 ms erzielt, wie er auch bei der Berechnung von $\boldsymbol{c}_\tau$ im letzten Abschnitt verwendet wird.

In weiteren Versuchen werden Differenzen bzw. Ableitungen zweiter Ordnung hinzugenommen, was zu weiteren Verbesserungen der Erkennungsraten führt. Ein Optimum wird erreicht, wenn die Länge des Fensters für die zweite Ableitung gleich der für die erste Ableitung ist. Allerdings werden bei den Experimenten nur die ersten sechs statischen Merkmalkomponenten abgeleitet um insgesamt nicht auf mehr als 24 Merkmale zu kommen: 12 statische, 6 Ableitungen erster Ordnung und 6 zweiter Ordnung.

Ein weiteres Experiment wird durch die Überlegung motiviert, anstelle von Differenzen oder

Regression, die ja nur eine Linearkombination der zeitlich angrenzenden Merkmale darstellen, eben diese Nachbarn selbst als Kontext-Merkmale zu verwenden und sie mit dem aktuellen Merkmalvektor aus 12 statischen Merkmalen zu konkatenieren. Um die Dimension der Merkmalvektoren konstant auf 24 zu halten, werden die Merkmale mit Linearer Diskriminanzanalyse (siehe auch Abschnitt 3.3) reduziert. Mit den so gewonnenen dynamischen Merkmalen werden die besten Erkennungsraten erzielt. In Versuchen werden zunächst auf beiden Seiten je n benachbarte Vektoren ($n = 1, 2, 3$) berücksichtigt, was einem Kontext bis zu 70 ms entspricht. Dabei verbessert sich die Erkennungsrate mit wachsendem Kontext. In weiteren Experimenten wird nur noch jeder m-te Vektor (z.B. $m = 3$) konkateniert. Beim maximalen Kontext von 370 ms wird die beste Erkennungsrate erreicht. Der Autor vermutet weitere Verbesserungen bei noch größerem Kontext, bricht aber die Versuchsreihe wegen des gestiegenen Rechenaufwands ab.

Das zweidimensionale Cepstrum liefert in [Rie94] die schlechtesten Erkennungsraten. Um die Anzahl der Merkmale auf konstant 24 zu halten werden sie in diesem Versuch mittels Hauptachsentransformation (vgl. Abschnitt 3.2) auf 24 reduziert.

2.5 Neuere Ansätze zur Merkmalberechnung

In diesem Abschnitt wird eine Auswahl neuerer Untersuchungen vorgestellt, deren Ziel es ist, die Merkmalberechnung zu verbessern bzw. ganz neue Ansätze zu finden.

In [Bat98] wird u.a. die Kosinustransformation (DCT) mit der Karhunen-Loève-Transformation (KLT, vgl. Kapitel 3) verglichen. Das Bandspektrum wird nun einmal mit Hilfe der KLT und einmal näherungsweise mit der DCT dekorreliert und reduziert. Man erhält jeweils 16 Koeffizienten; die mit der KLT berechneten liefern bessere Erkennungsraten.

In einem weiteren Experiment werden die ersten und zweiten Ableitungen hinzugezogen. Die Kosinustransformation liefert wieder 16 Koeffizienten sowie die Gesamtenergie. Zusammen mit den Ableitungen erhält man 51 Merkmale. Das KLT-Experiment ist schon mit nur 32 dekorrelierten Merkmalen besser. Dazu wird der nicht-homogene Raum aus Bandspektrum-Koeffizienten, Kurzzeitenergie und deren erste und zweite Ableitungen auf Hauptkomponenten reduziert.

Eine Optimierung der Merkmalberechnung wird auch in [Psu01a] angestrebt. Die Merkmalvektoren setzen sich hier aus statischen und jeweils genauso vielen ersten und zweiten Ableitungen zusammen. Variiert wird in zahlreichen Experimenten die Anzahl der Merkmale sowie die Anzahl der Mel-Filterbänke. Ergebnis ist, dass wesentlich weniger Filter als bisher angenommen ausreichen. Schon bei $3 \cdot 7 = 21$ Merkmalen und nur 9 Filtern werden optimale Wortakkuratheiten erreicht. Allerdings besteht die verwendete Stichprobe aus insgesamt nur 400 gelesenen Sätzen aus Zeitungen und ist kleiner als die in dieser Studienarbeit verwendete (siehe Abschnitt

4.1).

Experimente zur Optimierung der Filterbank, mit der das Modellspektrums aus Koeffizienten der Linearen Vorhersage komprimiert werden soll, werden in [Psu01b] durchgeführt.

Die KLT ist eine lineare Hauptachsentransformation. Ansätze mit nichtlinearer PCA (Principal Component Analysis) findet man in [Sch98]. Dieser Ansatz umfasst auch die Produktterme, die in Abschnitt 5.4 vorgestellt werden.

Ein ganz neuer Ansatz wird in [Her98] beschrieben. Statt der vertikalen Vektoren im Spektrogramm, die das Spektrum für bestimmte Zeitfenster wiedergeben, bilden horizontale Vektoren für verschiedene Bänder im Spektrogramm, die einen relativ langen Kontext von einer Sekunde berücksichtigen, die Grundlage für die weiteren Berechnungen. Solche zeitlichen Vektoren werden mit TRAP (specTRAl Pattern) bezeichnet.

Nach dieser Zusammenstellung grundlegender Verfahren zur Merkmalberechnung sowie neuerer Ansätze aus der Forschung werden im nächsten Kapitel Methoden zur Dekorrelation und Reduktion hochdimensionaler Merkmalvektoren vorgestellt.

Kapitel 3

Dekorrelation von Merkmalen

Die zentrale Idee in dieser Arbeit ist, eine sehr große Anzahl von Merkmalen zu berechnen, diese dann auf weniger Komponenten zu reduzieren und gleichzeitig möglichst wenig Information zu verlieren. Zur Dekorrelierung und Dimensionsreduktion der Merkmalkomponenten wird in dieser Arbeit vorwiegend die Karhunen-Loève-Transformation (KLT), eine Hauptachsentransformation (PCA: Principal Component Analysis), verwendet. Eine Alternative dazu ist unter vielen anderen die lineare Diskriminanzanalyse (LDA). Beide Methoden lassen sich auch als problemabhängige Reihenentwicklungen auffassen.

Eine andere Möglichkeit ist die Probabilistic PCA (PPCA). Hierbei werden nicht die Merkmalvektoren transformiert, sondern das Trainingsverfahren so geändert, dass die hochdimensionalen und stark korrelierten Vektoren geeignet durch die Ausgabeverteilung der HMMs repräsentiert werden.

Nach einer Zusammenfassung wichtiger Begriffe aus der Statistik, wird zunächst die KLT erklärt, ihre Bedeutung aufgezeigt sowie ihre Implementierung vorgestellt. In einem weiteren Abschnitt werden kurz LDA und PPCA erläutert.

3.1 Kovarianz und Korrelation

In diesem Abschnitt werden die Kovarianz und die Korrelation, zwei Begriffe aus der Statistik vorgestellt [Bos97, S. 133 ff].

Sei X eine Zufallsvariable mit Erwartungswert $\mu_X = E(X)$. So ist die Varianz von X

$$\begin{aligned} varX &= E[X - E(X)]^2 \\ &= E(X^2) - E(X)^2 \end{aligned} \tag{3.1}$$

Die Kovarianz zwischen zwei Zufallsvariablen X und Y ist

$$cov(X,Y) = E[(X - E(X)) \cdot (Y - E(Y))] \tag{3.2}$$
$$= E(X \cdot Y) - E(X) \cdot E(Y) \tag{3.3}$$

Es gilt offensichtlich $cov(X,Y) = cov(Y,X)$ sowie $cov(X,X) = var(X)$. Streuung σ und Korrelation ρ sind gemäß den folgenden Gleichungen definiert:

$$\sigma_X = \sqrt{varX} \tag{3.4}$$

$$\rho_{X,Y} = \frac{cov(X,Y)}{\sigma_X \sigma_Y} \tag{3.5}$$

Dabei ist zu beachten, dass die Korrelation nur für Zufallszahlen definiert ist, deren Varianzen nicht verschwinden. Ist die Korrelation Null, so heißen die beiden Zufallsvariablen unkorreliert. Sind zwei Zufallsvariablen stochastisch unabhängig, so impliziert dies Unkorreliertheit; die Umkehrung gilt i.A. nicht, d.h. unkorrelierte Zufallsvariablen können auch stochastisch abhängige sein. Die Korrelation ist auf den Wertebereich

$$-1 \leq \rho_{X,Y} \leq 1$$

eingeschränkt. Es lässt sich leicht zeigen, dass jede Zufallsvariable X mit sich selbst vollständig korreliert ist ($\rho_{X,X} = 1$). Im Allgemeinen gilt $|\rho_{X,Y}| = 1$ bei einer lineare Beziehung $Y = a \cdot X + b$. Liegen alle Elemente einer Stichprobe im zweidimensionalen Merkmalraum also auf einer steigenden Geraden, so ist die Korrelation 1, liegen sie auf einer fallenden Geraden, ist die Korrelation -1. Abbildung 3.1 veranschaulicht noch weitere Werte von ρ.

Bei n-dimensionalen Zufallsvektoren X mit Komponenten X_i ($i = 1, 2, ..., n$) betrachtet man die $n \times n$ Kovarianzmatrix $\boldsymbol{K}$. Auf ihrer Hauptdiagonalen stehen in der i-ten Zeile die Varianzen der Komponente X_i; in Zeile i und Spalte j steht $cov(X_i, X_j)$. Die Matrix ist symmetrisch und nichtnegativ definit:

$$\boldsymbol{K} = \begin{pmatrix} var(X_1) & cov(X_1,X_2) & \cdots & cov(X_1,X_n) \\ cov(X_2,X_1) & var(X_2) & \cdots & cov(X_2,X_n) \\ \vdots & \vdots & \ddots & \vdots \\ cov(X_n,X_1) & cov(X_n,X_2) & \cdots & var(X_n) \end{pmatrix} \tag{3.6}$$

Nach Gleichung 3.3 lässt sich die Kovarianzmatrix für eine Stichprobe mit N Merkmalvektoren $\boldsymbol{c}_\tau$ ($\tau = 0, 1, .., N-1$) und Mittelwert $\bar{\boldsymbol{c}} = 1/N \cdot \sum_{\tau=0}^{N-1} \boldsymbol{c}_\tau$ leicht aus dyadischen Produkten der Vektoren berechnen:

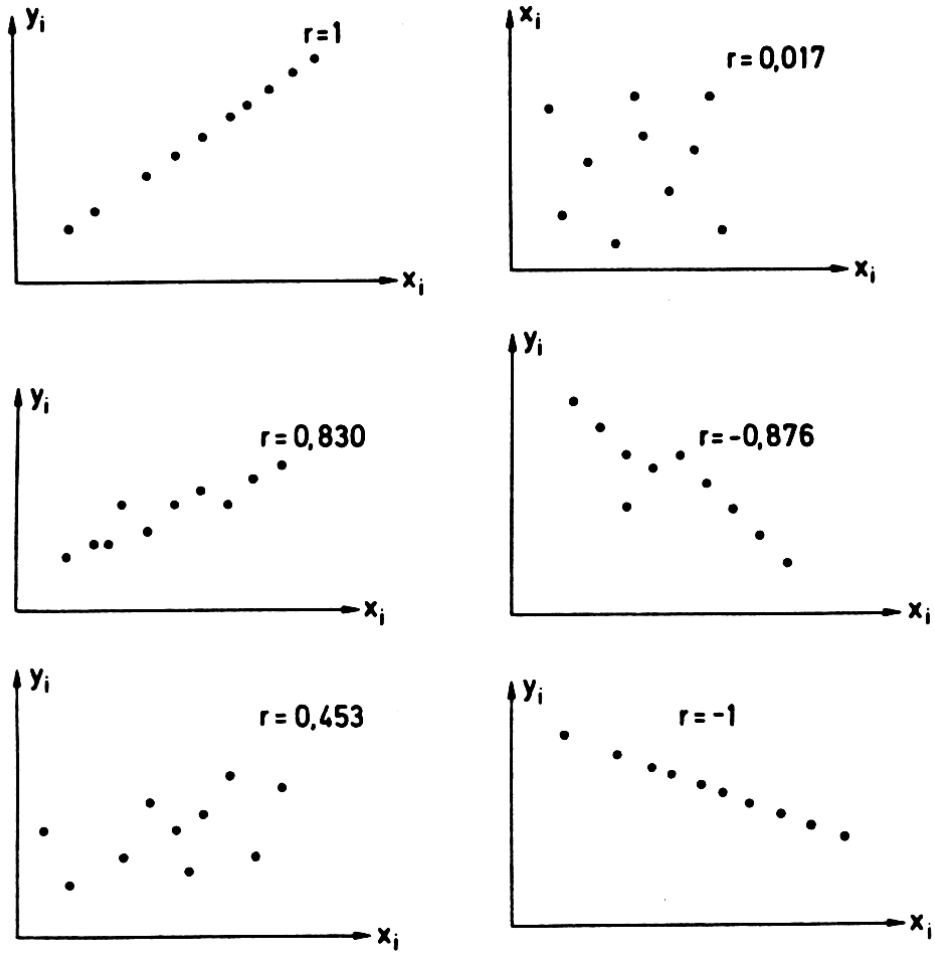

Bild 3.1: Korrelation von Punktmengen (hier mit r bezeichnet). Aus: [Bos97, S.138]

$$\boldsymbol{K} = \left(\frac{1}{N} \sum_{i=0}^{N-1} \boldsymbol{c}_i \boldsymbol{c}_i^T \right) - \bar{\boldsymbol{c}} \bar{\boldsymbol{c}}^T \tag{3.7}$$

Als effizientes Vorgehen zur Berechnung von $\boldsymbol{K}$ bietet sich an, nur die Werte einer Dreiecksmatrix zu berechnen und anschließend an der Hauptdiagonalen zu spiegeln. Die Definition der Korrelationsmatrix $\boldsymbol{R}$ ist schließlich:

$$\boldsymbol{R} = \begin{pmatrix} 1 & \rho_{X_1,X_2} & \cdots & \rho_{X_1,X_n} \\ \rho_{X_2,X_1} & 1 & \cdots & \rho_{X_2,X_n} \\ \vdots & \vdots & \ddots & \vdots \\ \rho_{X_n,X_1} & \rho_{X_n,X_2} & \cdots & 1 \end{pmatrix} \tag{3.8}$$

Auch R ist symmetrisch.

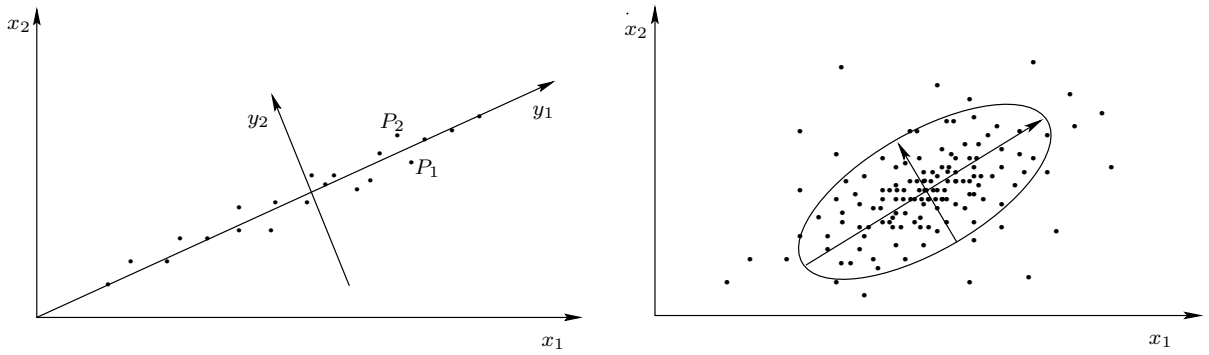

Bild 3.2: Links: Stark korrelierte Merkmale, beschrieben durch zwei verschiedene Koordinatensysteme. Rechts: Normalverteilte Merkmale

3.2 Karhunen-Loève-Transformation

Die Karhunen-Loève-Transformation (KLT) wird in der Statistik angewandt, um verschiedene Merkmale zu dekorrelieren und hochdimensionale Daten auf kleinere Dimensionen zu reduzieren. Sie ist äquivalent zur Hauptachsentransformation, falls der Mittelwert der Merkmale im Ursprung liegt [ST95, S.115]. Im Folgenden werden Grundlagen und Implementierung der KLT vorgestellt.

3.2.1 Vorgehen bei der KLT

Das zu lösende Problem ist, F-dimensionale Merkmalvektoren $\boldsymbol{c}_\tau$ auf $f < F$ Dimensionen zu reduzieren. Sei $\boldsymbol{d}_\tau$ der Fehler, der durch Projektion von $\boldsymbol{c}_\tau$ in den Unterraum entsteht, dann soll der mittlere quadratische Fehler

$$R^2 = E(||\boldsymbol{d}_\tau||^2) \tag{3.9}$$

minimiert werden [Sch77, S.249]. Zu optimieren sind hierbei der Ursprung des Koordinatensystems, das den f-dimensionalen Unterraum aufspannt und die Richtung seiner Koordinatenachsen. Die Lösung dieses Optimierungsproblems wird ausführlich in [Sch77, S.245ff] hergeleitet. Auf Einzelheiten wird hier nicht eingegangen, allein das Resultat soll erst anschaulich und dann formal beschrieben werden.

Für erste Überlegungen betrachte man Abbildung 3.2 links. Dort wird eine Menge von 2-dimensionalen Merkmalen gezeigt, die näherungsweise auf einer Geraden liegen. Die Achsen y_1 und y_2 stellen die Hauptstreuungsrichtungen der Merkmale dar und bilden eine Basis des Merkmalraumes. Projiziert man die Punkte entlang der Achse y_2 auf y_1, so erhält man ein eindimensionales Problem. Der Fehler ist so minimal. Zwar kann man nun die Punkte P_1 und P_2 nicht

mehr unterscheiden, würde man aber die Merkmale auf eine andere Achse projizieren, würde dieses Phänomen wohl öfter auftreten. Den größten Fehler würde man machen, wenn man entlang der y_1-Achse auf die y_2-Achse projiziert: Etliche im Raum sehr weit auseinanderliegende Punkte würden auf einen einzigen abgebildet. Es gingen so wesentliche Informationen verloren.

Die Vorgehensweise ist also diese: Zunächst translatiert man die Merkmalvektoren, so dass ihr Mittelwertvektor im Ursprung liegt. Dann werden die F Hauptstreuungsrichtungen der Merkmale im F-dimensionalen Merkmalraum berechnet. Diese stellen ein geeignetes problemabhängiges Koordinatensystem dar. Die Achsen sind paarweise zueinander senkrecht und bilden eine Orthogonalbasis des Merkmalraumes. Den kleinsten Fehler bei der Reduktion der Merkmale auf f Komponenten macht man nun, wenn man die $F - f$ Achsen auswählt, in deren Richtung die Merkmale am wenigsten gestreut sind, und die Punkte dann durch Parallelprojektion längs dieser Achsen in den orthogonalen f-dimensionalen Unterraum projiziert.

Es bleibt das Problem, die Hauptstreuungsrichtungen der Merkmale zu bestimmen. Sind die Merkmale normalverteilt, beschreibt man die Konturen gleicher Wahrscheinlichkeit im Merkmalraum durch F-dimensionale konzentrische Ellipsoide. Die Hauptstreuungsrichtungen zeigen dann in Richtung der Halbachsen, wie es in Abbildung 3.2 rechts verdeutlicht wird [Sch77, S.60]. In der Praxis ist die Verteilung der Merkmale aber meist unbekannt. Die Hauptstreuungsrichtungen werden dann aus einer genügend großen und repräsentativen Stichprobe, der Trainingsstichprobe, geschätzt. Sie sind die Hauptachsen der Merkmalvektoren aus der Stichprobe im Merkmalraum, also – wenn man sich zu jedem Punkt eine konstante Masse zugeordnet denkt – die Trägheitsachsen der Massenverteilung. Diese erhält man, indem man die F Eigenvektoren der $F \times F$ dimensionalen Kovarianzmatrix $\boldsymbol{K}$ (siehe Gleichung 3.6) der Trainingsstichprobe berechnet. Da die Eigenvektoren paarweise orthogonal sind, erhält man eine Orthogonalbasis des Merkmalraums.

Derjenige Eigenvektor, der dem größten Eigenwert zugehörig ist, zeigt in die Richtung mit maximaler Streuung. Sind die Eigenwerte absteigend geordnet (sie sind nichtnegativ, da die Matrix nichtnegativ definit ist), so wählt man als Unterraum denjenigen Raum, der von den f Eigenvektoren $\boldsymbol{b}_i$ mit den größten Eigenwerten λ_i ($i = 1, 2, ..., f$) aufgespannt wird. Den Ursprung legt man in den Mittelpunkt der Merkmalvektoren. Der Fehler R^2 beträgt nun

$$R^2 = \sum_{i=1}^{F} \lambda_i - \sum_{i=1}^{f} \lambda_i = Spur\boldsymbol{K} - \sum_{i=1}^{f} \lambda_i. \tag{3.10}$$

Er wird also durch Summieren derjeniger Eigenwerte, deren zugehörige Eigenvektoren weggelassen wurden, berechnet.

Sei $\boldsymbol{B}_f = (\boldsymbol{b}_1\boldsymbol{b}_2 \cdots \boldsymbol{b}_f)$ eine Matrix, deren Spalten die Eigenvektoren $\boldsymbol{b}_i$ bilden. Durch Abbildung der Merkmalvektoren $\boldsymbol{c}_\tau$ mit Mittelwert $\boldsymbol{\mu} = 1/N \cdot \sum_{\tau=0}^{N-1} \boldsymbol{c}_\tau$ in den Unterraum erhält

man nun Vektoren

$$\hat{\boldsymbol{c}}_\tau = \boldsymbol{B}_f^T(\boldsymbol{c}_\tau - \boldsymbol{\mu}); \quad \tau = 0, ..., N-1. \tag{3.11}$$

Diese Vektoren sind nun mittelwertfrei und besitzen als Kovarianzmatrix die Diagonalmatrix

$$\boldsymbol{K'} = \begin{pmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & \lambda_f \end{pmatrix} \tag{3.12}$$

mit den Eigenwerten λ_i ($i = 1, 2, ..., f$) auf ihrer Diagonale. Die Merkmalkomponenten sind jetzt also untereinander unkorreliert.

Fasst man die F normierten Eigenvektoren $\boldsymbol{b}_i$ als Orthonormalbasis des Merkmalraumes auf, so lassen sich die Merkmale $\boldsymbol{c}_\tau$ in diesem Raum als Linearkombination

$$\boldsymbol{c}_\tau = \sum_{i=1}^{F} \alpha_{i\tau} \cdot \boldsymbol{b}_i \tag{3.13}$$

schreiben, wobei die $\alpha_{i\tau}$ die Komponenten der transformierten Merkmalvektoren im problemabhängigen Merkmalraum sind. Da die Reihe aus Gleichung 3.13 bei den reduzierten Merkmalen nach $f < F$ Elementen abgebrochen wird, spricht man von unvollständiger Reihenentwicklung.

Es ist alternativ auch möglich, die Eigenvektoren und -werte nicht aus der Kovarianzmatrix $\boldsymbol{K}$ sondern aus der Korrelationsmatrix $\boldsymbol{R}$ (siehe Gleichung 3.8) zu berechnen. Unterschiede dieser beiden Vorgehensweisen werden in Abbildung 3.3 verdeutlicht. Die erste Graphik zeigt eine Stichprobe, die durch Translation so verschoben wurde, dass ihr Mittelpunkt im Ursprung liegt. Etliche Punkte dieser Stichprobe liegen auf einer Geraden, die parallel zur gekennzeichneten Richtung $\boldsymbol{v}$ verläuft. In den beiden Bildern darunter in der linken Spalte von Abbildung 3.3 wird eben diese Punktmenge mit der KLT transformiert, einmal nach Varianz-/Kovarianzanalyse und einmal nach Korrelationsanalyse.

Bei der Varianz-/Kovarianzanalyse wird $\boldsymbol{u}$ als Hauptstreuungsrichtung gefunden und die Punktwolke so gedreht (und gespiegelt), dass $\boldsymbol{u}$ parallel zur x_1-Achse ist. Bei der Korrelationsanalyse werden zunächst die Varianzen in x_1 und x_2-Richtung auf eins normiert. Die Stichprobe wird also wie in Abbildung 3.3 oben rechts gezeigt transformiert. Die Hauptstreuungsrichtung wird nun für diese normierte Stichprobe berechnet; sie wird hier mit $\boldsymbol{t}$ bezeichnet. Durch die KL-Transformation mit Korrelationsanalyse wird dann die originale, nicht normierte Stichprobe aus Abbildung 3.3 oben links so transformiert, dass $\boldsymbol{t}$ parallel zur x_1-Achse und $\boldsymbol{s}$ parallel zur x_2-Achse ist.

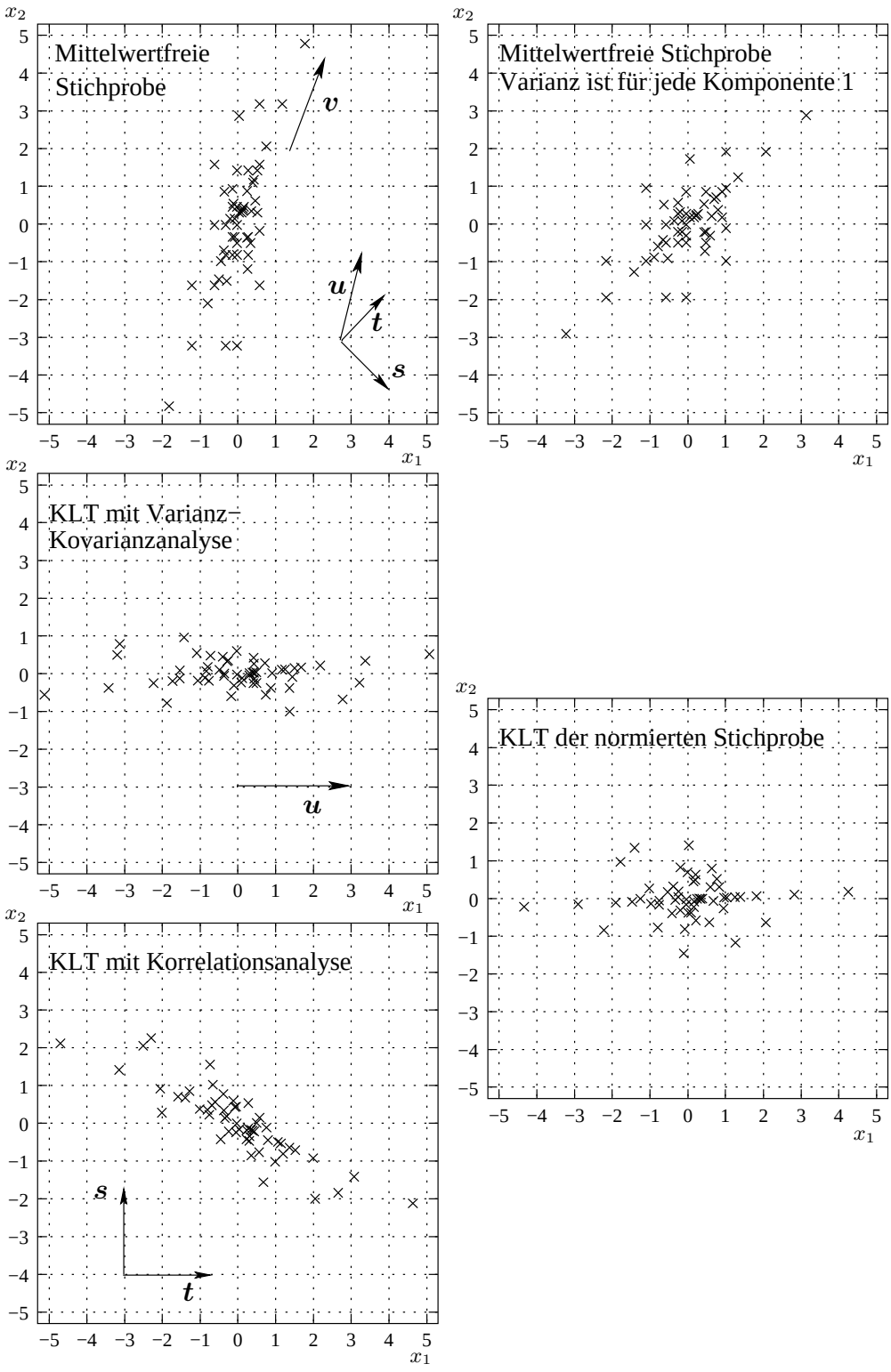

Bild 3.3: Transformation einer Stichprobe mit der KLT und verschiedenen Optionen

Die Verteilung der Punkte in $\boldsymbol{v}$-Richtung wird bei der Varianz-/Kovarianzanalyse als wichtige Information erachtet. Die Punktwolke wird so transformiert, dass nach Dimensionsreduktion im eindimensionalen Raum, der durch die x_1-Achse beschrieben wird, die Punkte in $\boldsymbol{v}$-Richtung klar trennbar sind. Grund dafür ist aber insbesondere, dass das Merkmal x_2 der ursprünglichen Stichprobe sehr viel weiter gestreut ist, als x_1, und der Winkel zwischen $\boldsymbol{v}$ und der x_2 -Achse sehr klein ist. Durch die Korrelationsanalyse sind ebenfalls die Punkte in $\boldsymbol{v}$-Richtung noch gut trennbar, es werden aber auch zu $\boldsymbol{v}$ oder zur x_2-Achse orthogonale Komponenten berücksichtigt, die nicht so stark gestreut sind.

Aus der beschriebenen Vorgehensweise bei der KLT wird verständlich, dass, wenn eine normierte Stichprobe vorliegt, deren Varianz in Richtung jeder Achse eins ist, sowohl die KLT mit Varianz-/Kovarianzanalyse als auch die nach Korrelationsanalyse das selbe Ergebnis erzeugen. In beiden Fällen wird ja nun dieselbe Hauptstreuungsrichtung gefunden, da der Normierungsschritt hinfällig wird. Dies ist in der rechten Spalte von Abbildung 3.3 verdeutlicht.

Nach der KL-Transformation mit Korrelationsanalyse beträgt der Rekonstruktionsfehler R^2 bei unvollständiger Entwicklung

$$R^2 = Spur\boldsymbol{R} - \sum_{i=1}^{f} \lambda_i = F - \sum_{i=1}^{f} \lambda_i. \tag{3.14}$$

Diese einfachere Darstellung resultiert daraus, dass sich die Spur einer Matrix aus der Summe ihrer Diagonalelemente berechnet, die bei der Korrelationsmatrix alle eins sind.

Zusammenfassend lässt sich sagen, dass durch die Karhunen-Loève-Transformation die Merkmalvektoren auf weniger Dimensionen reduziert werden, wobei der entstehende mittlere quadratische Fehler minimal bleibt und die einzelnen Komponenten dekorreliert werden.

3.2.2 Implementierung der KLT

Die Berechnung der Eigenwerte λ_i und des Mittelwerts $\boldsymbol{\mu}$ für eine Stichprobe von Merkmalvektoren, sowie der $F \times F$ Transformationsmatrix $\boldsymbol{B}_F = (\boldsymbol{b}_1 \boldsymbol{b}_2 \cdots \boldsymbol{b}_F)$ mit den Eigenvektoren $\boldsymbol{b}_i$ in den Spalten, erfolgt mit dem Programm `pca_train`. Optional kann man im wesentlichen zwischen Korrelationsanalyse und Varianz-/Kovarianzanalyse wählen.

Dem Programm `pca_test` wird eine von `pca_train` erzeugte Datei mit $\boldsymbol{\mu}$, den λ_i und den $\boldsymbol{b}_i$ übergeben. Zusätzliche Argumente sind der zu transformierende Merkmalvektor, sowie die tatsächliche Dimensionen des Vektors vor und die gewünschte Dimension nach der Transformation.

Die Korrelations- bzw Kovarianzmatrix wird nach den Formeln aus Abschnitt 3.1 berechnet, insbesondere Gleichung 3.7. Zur Berechnung der Eigenwerte wird die Matrix zunächst mit

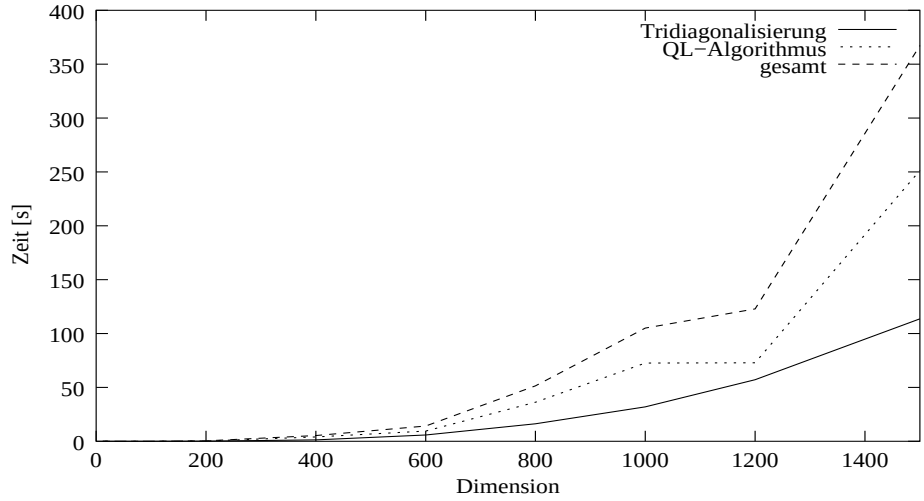

Bild 3.4: Laufzeitverhalten der Eigenwertberechnung für verschiedene Dimensionen

Householder-Matrizen auf Tridiagonalform gebracht [Pre92, S.470 - 475]. Die Eigenvektoren und -werte werden dann mit dem QL-Algorithmus berechnet [Pre92, S.476 - 481]. Der Algorithmus gilt numerisch als wesentlich stabiler, als der Von-Mises-Algorithmus der im Programm `karhunen` implementiert und in den Arbeiten aus [Rie94] verwendet wird.

Man beachte, dass diese Berechnungen mit `pca_train` nur einmalig zum Trainieren des Erkenners erforderlich sind. Für 24-dimensionale Matrizen fallen die Berechnungszeiten gar nicht ins Gewicht. Angaben zum Laufzeitverhalten der Eigenwertberechnungen für $n \times n$-dimensionale Matrizen ($n \in \{12, 24, 48, 100, 200, 400, 600, 800, 1000, 1200, 1500\}$ findet man in Abbildung 3.4. Die Komplexität des Tridiagonalisierungs-Algorithmus ist $O(\frac{4}{3}n^3)$, die des QL-Algorithmus $O(3n^3)$ [Pre92, S.474 und S.480].

Nach den Berechnungen wird die Gültigkeit der Eigenwertgleichung

$$\boldsymbol{M}\boldsymbol{b}_i = \lambda_i \boldsymbol{b}_i \tag{3.15}$$

überprüft. $\boldsymbol{M}$ ist hierbei die $F \times F$ Kovarianzmatrix $\boldsymbol{K}$ bzw. die Korrelationsmatrix $\boldsymbol{R}$, λ_i und $\boldsymbol{b}_i$ ($i = 1, 2, ..., F$) sind die Eigenwerte bzw -vektoren. Die Abweichung von der Gleichheit wird für alle F Eigenvektoren gemittelt

$$s_1 = \frac{1}{F} \cdot \sum_{i=1}^{F} ||\boldsymbol{M}\boldsymbol{b}_i - \lambda_i \boldsymbol{b}_i||_1, \tag{3.16}$$

wobei $||.||_1$ die Betragssummennorm bezeichnet, welche die Komponenten eines Vektors addiert. s_1 sollte im Idealfall Null sein. Ferner wird die paarweise Orthogonalität der Eigenvektoren geprüft. Wieder wird die Summe durch die Anzahl der Summanden geteilt, um einen Mittelwert

	s_1	s_2	m_1	m_2
Korrelationsanalyse	$11.20 \cdot 10^{-07}$	$6.82 \cdot 10^{-08}$	$20.15 \cdot 10^{-07}$	$5.63 \cdot 10^{-07}$
Varianz-/Kovarianzanalyse	$12.48 \cdot 10^{-07}$	$5.08 \cdot 10^{-08}$	$38.21 \cdot 10^{-07}$	$2.96 \cdot 10^{-07}$

Tabelle 3.1: Die numerischen Fehler s_1 , s_2, m_1 und m_2

zu erhalten.

$$s_2 = \frac{2}{F(F-1)} \sum_{i=1}^{F} \sum_{j=i+1}^{F} \frac{|\boldsymbol{b}_i^T \boldsymbol{b}_j|}{||\boldsymbol{b}_i|| \cdot ||\boldsymbol{b}_j||} \tag{3.17}$$

Auch s_2 sollte möglichst verschwinden. Zusätzlich werden auch die maximal auftretenden Abweichungen m_1 und m_2 berechnet:

$$m_1 = \max_i ||\boldsymbol{M}\boldsymbol{b}_i - \lambda_i \boldsymbol{b}_i||_1, \tag{3.18}$$

$$m_2 = \max_{i<j} \frac{|\boldsymbol{b}_i^T \boldsymbol{b}_j|}{||\boldsymbol{b}_i|| \cdot ||\boldsymbol{b}_j||} \tag{3.19}$$

Ein Beispiel, für die Größenordnung dieser Fehlerwerte gibt Tabelle 3.1, für welche die Standardtrainingsstichprobe, die im nächsten Abschnitt vorgestellt wird, analysiert wurde. Das Verhalten dieser Fehler für große Matrizen findet man im Anhang A.

In den Experimenten zur Optimierung der Merkmalberechnung in Kapitel 5 wird ausschließlich die Korrelationsanalyse verwendet. So können bessere Ergebnisse erzielt werden, wie ein Vergleichsexperiment in Kapitel 5 Tabelle 5.1 zeigen wird. Entscheidend ist jedoch, dass die Versuche in dieser Arbeit nur nach strikt einheitlichem Vorgehen vergleichbar sind.

3.3 Alternative Vorgehensweisen

Alternativen zur Karhunen-Loève-Transformation (Principal Component Analysis, PCA) sind unter vielen anderen die lineare Diskriminanzanalyse (LDA) oder die Probabilistic PCA (PPCA). Da diese Verfahren in der vorliegenden Arbeit keine, bzw. nur eine untergeordnete Rolle spielen, werden sie hier nur knapp vorgestellt.

Abbildung 3.5 verdeutlicht, dass die KL-Transformation zur Klassentrennung nicht immer geeignet ist: Die Ellipse wird so gedreht, dass ihre große Hauptachse auf der y_1-Achse des Koordinatensystems liegt. Dann aber ist y_2 das optimale Merkmal um die drei Punktwolken unterscheiden zu können, die KLT mit Dimensionsreduktion auf ein Merkmal wählt aber y_1 aus; die drei Klassen werden zusammengeworfen.

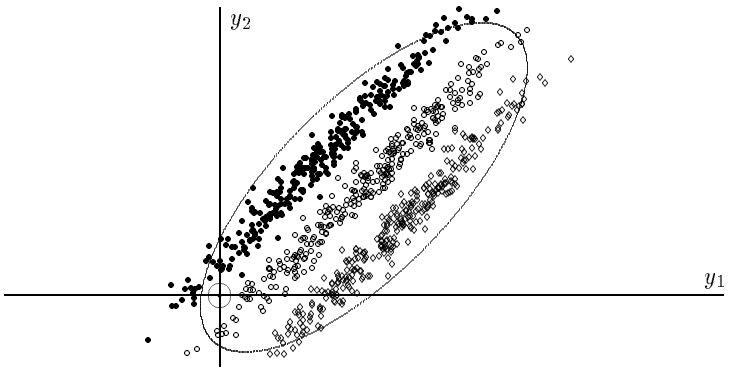

Bild 3.5: Das Adidas Problem, aus [ST95, S.116]

Bei der LDA wird nun versucht, getrennte Gebiete als solche zu erhalten. Dazu werden eine Intraklassen-Streuungsmatrix und eine Interklassen-Streuungsmatrix eingeführt [Nie83, S.110], die zu berechnen allerdings voraussetzt, dass die Klassenzugehörigkeit der Stichprobe bekannt ist. In [ST95, S.116ff] wird die Berechnung der LDA- Transformationsmatrix erläutert, die den Intraklassenabstand konstant hält, jedoch — um die Klassen besser trennen zu können — ihre Klassenzentren auseinander zieht. Die LDA wird in dieser Arbeit nicht verwendet, da zum Zeitpunkt der Anfertigung keine numerisch stabile Version vorlag.

Bei PCA und LDA werden zuerst die hochdimensionalen Merkmalvektoren auf weniger Dimensionen reduziert und danach mit diesen kleineren Merkmalvektoren ein Erkenner trainiert. Bei der PPCA [Tip99] dagegen trainiert man direkt mit den hochdimensionalen und stark korrelierten Vektoren, jedoch wird eine geeignete Ausgabe-Dichte für das Sprachmodell verwendet, die mit diesen Vektoren umgehen kann.

Sei wieder F die Dimension der hochdimensionalen Merkmalvektoren, deren f Hauptkomponenten betrachtet werden sollen ($f < F$). Die PPCA-Dichten sind spezielle Gaußsche Wahrscheinlichkeitsdichten, die mit $F \times F$ Kovarianzmatrizen

$$\boldsymbol{\Sigma} = \boldsymbol{W}\boldsymbol{W}^T + \sigma^2 \boldsymbol{I} \tag{3.20}$$

modelliert werden. Dabei ist $\boldsymbol{W}$ eine $F \times f$ Transformations-Matrix, $\boldsymbol{I}$ die $F \times F$ Einheitsmatrix und σ^2 die durch die Dimensionsreduktion "verlorene" Varianz, gemittelt über den "verlorenen" Dimensionen:

$$\sigma^2 = \frac{1}{F-f} \sum_{i=f+1}^{F} \lambda_i. \tag{3.21}$$

Die λ_i sind die Eigenwerte der Kovarianzmatrix der Stichprobe. Die $F \times f$ Matrix $\boldsymbol{W}$ ist schließlich

$$\boldsymbol{W} = \boldsymbol{U}(\boldsymbol{\Lambda} - \sigma^2 \boldsymbol{I})^{1/2} \boldsymbol{R}. \tag{3.22}$$

$\boldsymbol{U}$ ist eine $F \times f$ Matrix mit den f Eigenvektoren der Kovarianzmatrix der Stichprobe, die zu den größten Eigenwerten $\lambda_1 ... \lambda_f$ gehören. Jene findet man auf der Diagonalen der $f \times f$ Matrix $\boldsymbol{\Lambda}$. $\boldsymbol{R}$ ist eine beliebige orthogonale $f \times f$ Rotations-Matrix.

Sind die Ausgabedichten der HMMs eine Mischung aus PPCA-Dichten, so wird erwartet, dass der Merkmalraum besser repräsentiert werden kann, als durch eine Mischung aus Gauß-Dichten nach Merkmalreduktion durch die Standard-PCA [Ste01].

Bisher wurden Methoden zur Merkmalberechnung vorgestellt sowie Verfahren, hochdimensionale Merkmalvektoren zu reduzieren. Bevor die Experimente dieser Arbeit beschrieben und Ergebnisse präsentiert werden, folgt nun noch ein Kapitel zur Beschreibung des Aufbaus der Versuche.

Kapitel 4

Beschreibung des Versuchsaufbaus

Nachdem in den vorangegangenen Kapiteln die theoretischen Grundlagen für die Berechnung und statistische Analyse der Merkmale behandelt worden sind, wird in diesem Kapitel die Vorgehensweise bei der Durchführung der Experimente, welche im nächsten Kapitel erläutert werden, beschrieben. In knapper Form kann der Ablauf so zusammengefasst werden: Man benötigt eine Stichprobe, aus der in unterschiedlicher Weise Merkmale berechnet werden. Mit diesen trainiert man einen Erkenner, der einer zeitlichen Folge von Merkmalvektoren eine Folge von Wörtern zuordnet, die am wahrscheinlichsten gesprochen wurde. Anschließend testet man diesen Erkenner und bewertet die Testergebnisse geeignet, um eine Aussage darüber treffen zu können, wie gut die jeweilige Vorgehensweise bei der Merkmalberechnung ist. Auch soll dem Leser eine Vorstellung über die zeitliche Dauer dieser Berechnungen gegeben werden.

4.1 Die Stichprobe

Zur Durchführung der Experimente in dieser Arbeit wurde auf eine feste Menge von Sprachaufnahmen zurückgegriffen, die EVAR-Stichprobe. Es handelt sich hierbei um aufgezeichnete Anfragen verschiedener Personen in kontinuierlicher, frei gesprochener Sprache an das Zugfahrplan-Auskunft-System am Lehrstuhl für Mustererkennung (LME) der Universität Erlangen-Nürnberg. EVAR bedeutet “Erkennen, Verstehen, Antworten, Rückfragen” und ist in [Gal98] beschrieben.

Die gesamte Stichprobe ist aus Mikrofon- und Telefonaufnahmen gemischt. Bessere Erkennungsraten werden aber erzielt, nachdem die Mikrofonaufnahmen bandpaßgefiltert worden sind, so dass sie nun keine nennenswerten Frequenzanteile über 4000 Hz mehr enthalten.

Aus den 20678 Äußerungen wurde zufällig eine Teilmenge von insgesamt 7438 ausgewählt, um das Training und Testen zu beschleunigen. Diese wurde in drei disjunkte Stichproben zer-

legt, eine Trainingsstichprobe bestehend aus 4999, eine Validierungsstichprobe aus 441, und eine Teststichprobe aus 1998 Äußerungen[1]. Insgesamt entspricht dies etwa acht Stunden gesprochener Sprache; fast alle vorkommenden Wörter sind Deutsch.

Mit diesen Stichproben werden nun Erkenner für Sprache in Telefonqualität trainiert und getestet. Die besonders erfolgreichen Experimente werden zusätzlich mit großen Stichproben durchgeführt. Die Trainingsstichprobe beinhaltet dann 15721 , die Validierungsstichprobe wieder 441, und die Teststichprobe 3908 Äußerungen.

4.2 Bewertung

Bevor im nächsten Abschnitt die Vorgehensweise zum Testen eines Erkenners erläutert wird, wird nun zunächst ein Gütemaß für die Bewertung der erkannten Wortfolge eingeführt [Rie94, S.115f]. Dazu wird jene Wortkette mit der für die Testdaten vorliegenden Verschriftung verglichen. Man unterscheidet drei mögliche Abweichungen der erkannten Wortfolge von der tatsächlich gesprochenen:

- Ein nicht gesprochenes Wort wurde fälschlicherweise eingefügt. w_{ins} bezeichnet die Anzahl solcher Einfügungen (engl.: *insertions*).
- Ein gesprochenes Wort wurde ausgelassen. Die Anzahl solcher Auslöschungen (engl.: *deletions*) wird mit w_{del} bezeichnet.
- Ein gesprochenes Wort wurde falsch erkannt, d.h. durch ein falsches substituiert. w_{subs} ist die Anzahl der Substitutionen.

Es können also zwei verschieden lange Wortketten zum Vergleich vorliegen. Sei nun w_{ges} die Gesamtzahl der gesprochenen Wörter. Dann definiert man die Wortakkuratheit

$$WA = \left(1 - \frac{w_{subs} + w_{ins} + w_{del}}{w_{ges}}\right) \cdot 100\% \tag{4.1}$$

Der Levensthein-Abstand [Lev66] gibt die minimale Anzahl von Einfügungen, Auslöschungen und Substitutionen an, die benötigt werden, um die gesprochene Wortkette auf die erkannte abzubilden. Man beachte, dass WA in sehr schlechten Fällen auch negativ werden kann. Das Komplement zur Wortakkuratheit ist die Wortfehlerrate. Bei der Wortkorrektheit bleiben die Einfügungen unberücksichtigt:

[1] Es hat keinerlei Bedeutung, dass die Größe der Stichproben nicht etwa 5000 bzw. 2000 beträgt (Durch einen Fehler zu Beginn der Experimente sind drei Dateien verloren gegangen).

$$WC = \left(1 - \frac{w_{subs} + w_{del}}{w_{ges}}\right) \cdot 100\% \tag{4.2}$$

Findet die Bewertung auf der Ebene der korrekt erkannten Merkmalvektoren statt, spricht man von der Gesamterkennungsrate (Anteil der korrekt erkannten Vektoren). Ihr Komplement ist die Fehlerrate. Beide spielen in dieser Arbeit jedoch keine Rolle.

4.3 Training und Test

Die Experimente, die in dieser Arbeit durchgeführt werden, unterscheiden sich in der Merkmalberechnung. Nachdem Merkmale berechnet worden sind, wird ein Spracherkenner trainiert und anschließend getestet. Die Testergebnisse sind dann natürlich in den einzelnen Versuchen unterschiedlich, aber vergleichbar.

4.3.1 Training des Erkenners mit ISADORA

Zunächst wird mit den aus der Trainings- und Validierungsstichprobe berechneten Merkmalvektoren ein Erkenner trainiert. Dies geschieht mit dem auf HMMs basierenden Mustererkennungssystem ISADORA, das am Lehrstuhl für Mustererkennung (LME) der Universität Erlangen-Nürnberg entwickelt wurde. Eine Beschreibung dieses Systems findet man in [ST95, S.271ff.]. Eine schematische Darstellung zeigt Abbildung 4.1. Für die EVAR-Stichprobe liegt eine Verschriftung aller Sprachdateien vor, sowie die phonetische Transkription der 2640 Wörter des verwendeten Wortschatzes. Alle Laute und Nonverbalien (Geräusche wie z.B. Husten) sind wiederum in ihre subphonemischen Untereinheiten zerteilt.

Der Erkenner wird mit den berechneten Merkmalvektoren mit Hilfe dieses Systems so lange trainiert, bis die Wortakkuratheit, die auf der Validierungsstichprobe erzielt wird, konvergiert. Die Konvergenz ist in Abbildung 4.2 verdeutlicht: Es werden abwechselnd einmal das Codebuch und einmal die HMM-Parameter mit dem Baum-Welch-Algorithmus neu geschätzt. Jeder dieser Berechnungsabschnitte umfasst bis zu zehn Iterationen. Bei jeder Iteration wird die Wortakkuratheit auf der Validierungsstichprobe berechnet. Nimmt die Wortakkuratheit ab, so wird bereits vor der zehnten Iteration abgebrochen. Nach dem Abbruch bzw nach spätestens zehn Iterationen erfolgt eine Neuinitialisierung, woraus die Einbrüche resultieren.

In Abbildung 4.2 erkennt man 20 Einbrüche und 20 relative Maxima. Im Beispiel werden also zehn Codebücher und HMM-Netzwerke berechnet - mehr als genug, da schon mit dem sechsten eine Wortakkuratheit von 55.8 % erreicht wird, die sich nur noch auf 56,4 % verbessert. Eine Konvergenz der relativen Maxima ist also schon früher erkennbar, da es sich hierbei aber

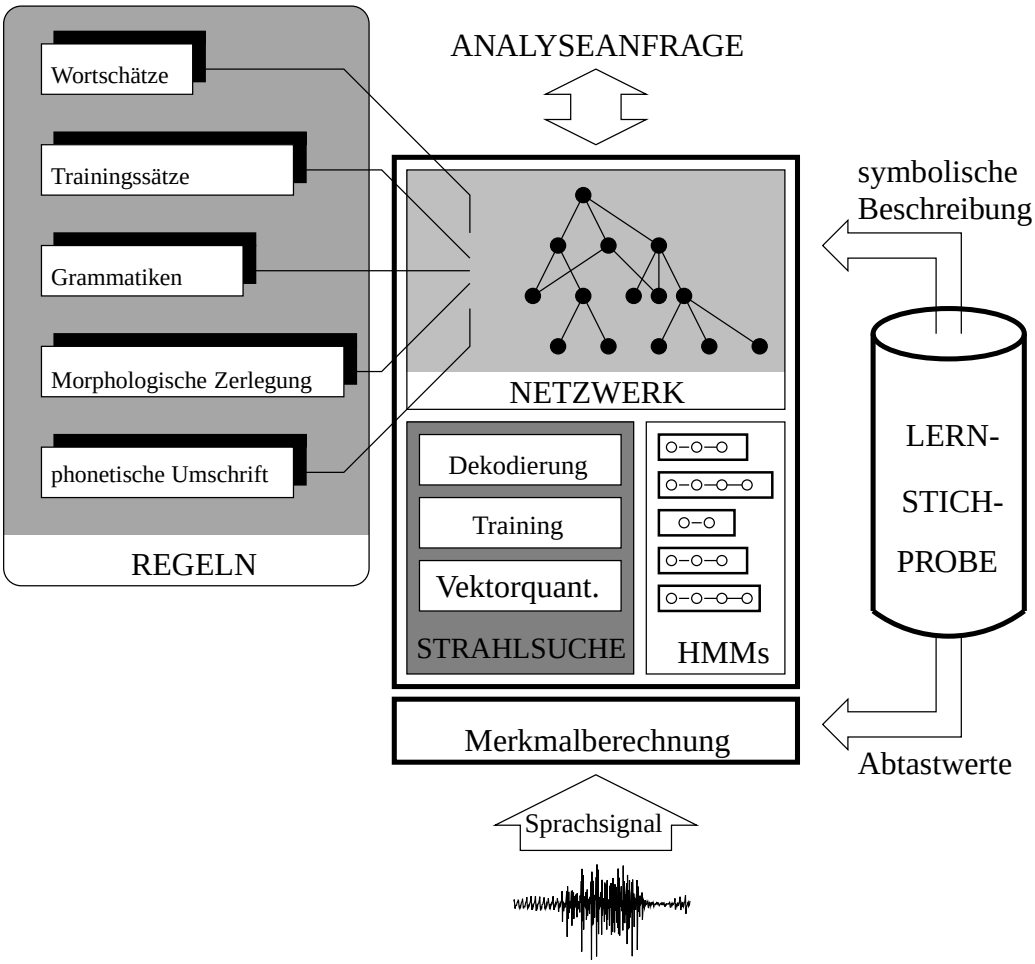

Bild 4.1: Architektur des ISADORA-Systems, aus [ST95, S.279]

um den Erkenner mit der ursprünglichen Merkmalberechnungen handelt, mit dem alle anderen Experimente verglichen werden sollen, war hier ein äußerst gründliches Training notwendig.

Bei den Untersuchungen in dieser Arbeit liegt teilweise eine deutlich langsamere Konvergenz vor. Folglich werden im Durchschnitt pro Experiment zehn bis elf Codebücher und HMM-Netze berechnet. In jedem Fall wird sorgfältig abgeschätzt, ob die Wortakkuratheit gegen einen viel zu geringen Wert konvergiert, und das Training frühzeitig abgebrochen werden kann, oder ob die Hoffnung besteht, das Vergleichssystem mit den ursprünglichen Berechnungen zu “übertreffen”. Wohl wegen der zu kleinen Validierungsstichprobe ist ein solches Abschätzen aber nur sehr grob möglich. In unterschiedlichen Experimenten kann man auch bei um etwa fünf Prozentpunkte unterschiedlichen Wortakkuratheiten beim Training nicht im Voraus sagen, welcher Erkenner auf der *Teststichprobe* bessere Ergebnisse liefert. Mehr Informationen zum Testen findet man im nachfolgenden Unterabschnitt.

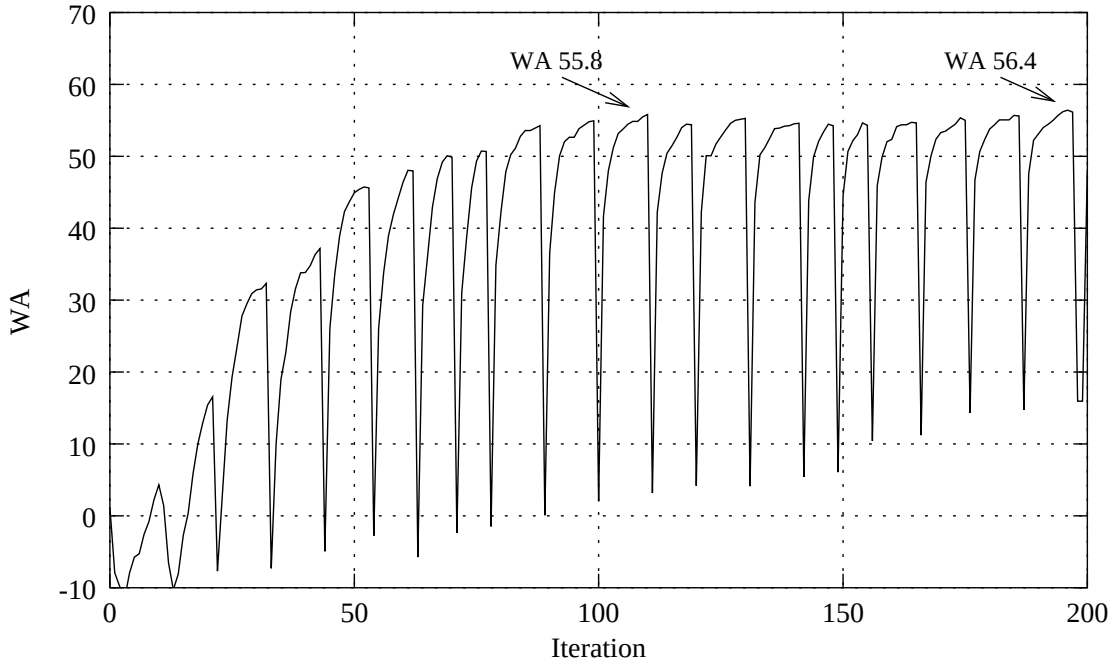

Bild 4.2: Konvergenz der Wortakkuratheit beim Training

Das Training kann dadurch beschleunigt werden, dass man in ausgewählten Runden das Neuschätzen des Codebuches überspringt. Auch erwies es sich als sinnvoll, grundsätzlich erst nach zehn Iterationen neu zu initialisieren und nicht schon dann, wenn sich die Wortakkuratheit erstmals wieder verschlechtert. So kann das Evaluieren in vielen Iterationen übersprungen werden.

4.3.2 Test mit LRBEAM-Erkenner

Nach dem Training eines Spracherkenners wird dieser mit einer zu den Trainings- und Validierungsdaten disjunkten Teststichprobe getestet. Dies geschieht mit dem LRBEAM-Erkenner (Programmname `lr_beam`). Ausgegeben werden u. a. die erkannten Sätze in Textform, sowie die berechnete Wortakkuratheit. Letztere ist beim Test stets wesentlich höher als jene Wortakkuratheit, die beim Training mit der Validierungsstichprobe erzielt wird, da nun ein Sprachmodell verwendet wird. Im Beispiel aus Abbildung 4.2 beträgt die Wortakkuratheit beim Test 69,89 %, die Wortkorrektheit 75.26%.

Mit verschiedenen im Laufe des Trainings berechneten Codebüchern und HMM-Netzen können unterschiedliche Erkenner zusammengebaut werden, die auf derselben Testmenge unterschiedlich gut (gemessen in Wortakkuratheit) sind, sich ab einem gewissen Konvergenzstadium aber nur noch um Bruchteile von einem Prozent unterscheiden. Im Beispieltraining aus

Abbildung 4.2 ist das letzte berechnete Codebuch sowohl auf der Validierungs- als auch auf der Teststichprobe das beste, was nicht immer der Fall ist. In den Experimenten werden einheitlich jeweils Erkenner aus denjenigen Codebüchern und HMM-Netzen berechnet, die auf der Validierungsstichprobe die besten Ergebnisse erzielt haben. Diese werden getestet und die resultierenden Ergebnisse in den Tabellen des folgenden Kapitels aufgelistet.

Die Vorgehensweise ist also äquivalent zu einem in verschiedenen Experimenten unterschiedlich langen Training – was wegen unterschiedlich schneller Konvergenz durchaus angebracht ist – und Abbruch an der Stelle, an der ein möglichst optimales Ergebnis auf der Validierungsstichprobe erzielt wird. Unter der Annahme, dass die im Vergleichsexperiment aus Abbildung 4.2 erzielte Wortakkuratheit von 69,89 % optimal ist, d.h in Abbildung 4.2 auch wirklich Konvergenz vorliegt, lässt sich folgendes sagen: Wird diese Wortakkuratheit von einem anderen Experiment übertroffen, das sich nur in der Merkmalberechnung unterscheidet, wurden in diesem Experiment wohl geeignetere Merkmale für wenigstens diese spezielle Stichprobe gefunden.

4.4 Laufzeiten

In diesem Abschnitt wird kurz auf die Laufzeiten der Berechnungen eingegangen. Gearbeitet wurde an Intel Pentium-III Rechnern mit 600 bzw. 700 MHz Prozessoren. Das gesamte Training eines Erkenners dauert etwa 6-7 Tage, das Testen samt Bewerten etwa 3-4 Stunden. Auch bei gleichzeitiger Benutzung mehrerer Rechner war also die Gesamtzahl der durchführbaren Experimente sehr eingeschränkt.

Bisher wurden theoretische Grundlagen dargelegt, sowie der allgemeine Aufbau der Experimente beschrieben. Im nächsten Kapitel werden nun die einzelnen Versuche motiviert und beschrieben sowie ihre Ergebnisse diskutiert.

Kapitel 5

Optimierung der Merkmalberechnung

In diesem Kapitel werden die Experimente vorgestellt, die unternommen wurden, um die Merkmale für die Spracherkennung zu verbessern. Grundidee in allen Versuchen ist es, zunächst wesentlich mehr als die bisher verwendeten 24 Merkmale zu berechnen. Diese $F > 24$ Merkmale werden dann mittels KL-Transformation auf $f < F$ Merkmale reduziert. Man erhält so die f besten Merkmale, die aus Linearkombination der F Merkmale hervorgehen. Im Großteil der Experimente ist $f = 24$, um vergleichen zu können, ob die neuen Merkmale auch wirklich qualitativ besser sind, als die alten.

In jedem Experiment wird eigens ein Erkenner trainiert, der anschließend mit der Teststichprobe getestet wird. Verglichen werden die Wortakkuratheiten, die auf der Teststichprobe erzielt werden. Für die bisherigen, in Abschnitt 2.3 beschriebenen Merkmale wurde eine Wortakkuratheit (WA) von 69.89 % erzielt.

Im ersten Abschnitt werden diese bisherigen Merkmale mit der KLT verändert. In den darauf folgenden beiden Abschnitten wird getrennt versucht, die dynamischen bzw. statischen Merkmale dadurch zu verbessern, dass man sie für verschiedene Zeitauflösungen berechnet. Zuletzt werden Versuche mit Produkttermen durchgeführt. Abschließend sind die wichtigsten Ergebnisse nochmals zusammengestellt.

5.1 Veränderung der Merkmale mit KLT

Im Folgenden werden Experimente diskutiert, in denen versucht wird, die Karhunen-Loève-Transformation direkt auf die im Abschnitt 2.3 beschriebenen Merkmale anzuwenden, ohne zusätzliche oder andere Merkmale zu berechnen.

Zunächst werden hier die 24 Merkmale durch vollständige Entwicklung der KLT in einen anderen 24-dimensionalen Merkmalraum transformiert. Dabei wird die KLT einmal auf der Basis

Merkmale	**# Merkmale** KLT: input → output	**WA** in %
Bisherige Merkmale		69.89
KLT mit Korrelationsanalyse	$24 \rightarrow 24$	71.27
KLT mit Varianz-/Kovarianzanalyse	$24 \rightarrow 24$	70.25

Tabelle 5.1: Vollständige Entwicklung der bisherigen Merkmale mit KLT

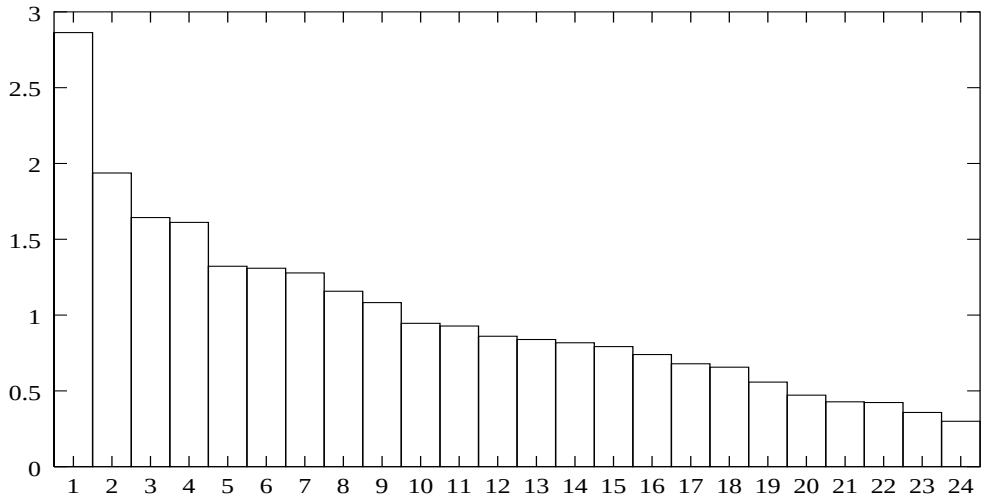

Bild 5.1: Eigenwerte der Korrelationsmatrix der Trainingsstichprobe

einer Korrelationsanalyse und einmal mit einer Varianz-/Kovarianz-Analyse verwendet. In beiden Fällen sind Verbesserungen der Wortakkuratheiten zu verzeichnen. Auch erkennt man, dass die Korrelationsanalyse bessere Ergebnisse als die Kovarianzanalyse liefert (Tabelle 5.1).

Abbildung 5.1 zeigt die sortierten Eigenwerte der Korrelationsmatrix, die den Anteil der Gesamtstreuung in Richtung der jeweiligen Hauptachsen wiedergeben. Nun wird untersucht, inwieweit sich die Wortakkuratheit bei der unvollständigen Entwicklung der 24 Merkmale aus Abschnitt 2.3 verändert. Die Merkmale werden längs der Hauptachsen mit den kleinsten zugehörigen Eigenwerten in den orthogonalen Unterraum projiziert. Ergebnisse sind in Tabelle 5.2 aufgelistet.

Aus der Tatsache, dass die Wortakkuratheit mit sinkender Anzahl von Merkmalen abnimmt, lässt sich folgern, dass jene weitgehend unkorreliert sind. Schon beim Verzicht auf die beiden Hauptachsen mit geringster Streuung gehen wichtige Information verloren. Graphisch ist der Vergleich der Experimente aus Tabelle 5.2 in Abbildung 5.2 veranschaulicht.

Merkmale	**# Merkmale** KLT: input → output	**WA** in %
KLT mit Korrelationsanalyse	$24 \rightarrow 22$	69.35
	$24 \rightarrow 20$	68.61
	$24 \rightarrow 18$	66.02
	$24 \rightarrow 16$	66.27
	$24 \rightarrow 14$	63.42
	$24 \rightarrow 12$	61.99
	$24 \rightarrow 10$	62.94
	$24 \rightarrow 8$	61.01
	$24 \rightarrow 6$	55.90

Tabelle 5.2: Unvollständige Entwicklung der bisherigen Merkmale mit KLT

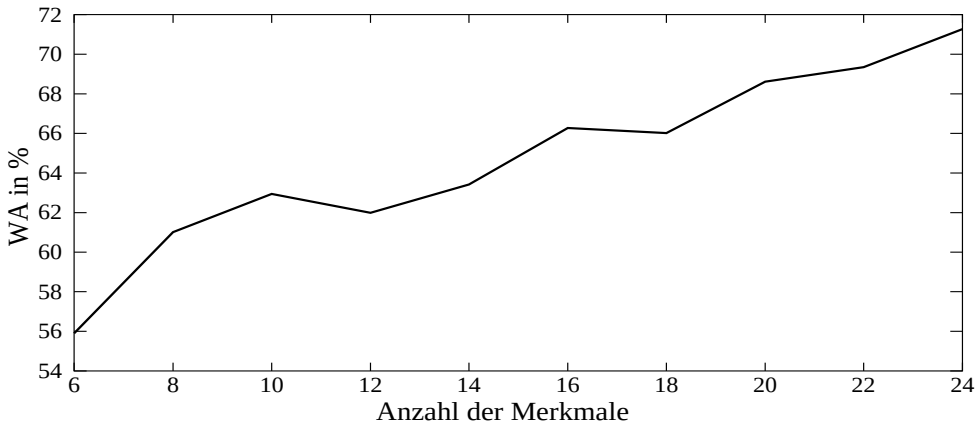

Bild 5.2: Wortakkuratheiten für verschiedene Anzahl von Merkmalen nach unvollständiger Entwicklung mit KLT

Ableitungen über diese Fensterbreite	**# Nachbarn** pro Richtung ($= M$)	**# Merkmale** KLT: input $\rightarrow$ output	**WA** in %
30 ms	1	$12 \rightarrow 12$	72.08
50 ms	2	$12 \rightarrow 12$	**72.48**
70 ms	3	$12 \rightarrow 12$	71.37
90 ms	4	$12 \rightarrow 12$	69.19

Tabelle 5.3: Vollständige Entwicklung der 12 Ableitungen mit KLT

5.2 Dynamische Merkmale für verschiedene Kontextfenster

In diesem Abschnitt werden die dynamischen Merkmale optimiert, also diejenigen, die den Kontext berücksichtigen. Der Kontext erstreckt sich jeweils symmetrisch zum aktuellen Kurzzeitanalysefenster auf der Zeitachse in beiden Richtungen über endlich viele angrenzende Fenster. Die Kontextmerkmale werden aus solchen verschieden breiten Kontextfenstern durch Ableitung bzw. Konkatenation der statischen Merkmale berechnet, wie in den beiden folgenden Abschnitten beschrieben wird.

5.2.1 Kontext durch Ableitungen

Die Kontextmerkmale werden in den Untersuchungen dieses Abschnitts durch Ableitung der statischen Merkmale berechnet. Approximiert wird die Ableitung mit Hilfe der Regressionsgeraden. Diese berechnet sich aus $2M + 1$ Kurzzeitanalysefenstern, dem aktuellen und den M angrenzenden auf jeder Seite. Es fließen also statische Merkmale aus M zeitlich benachbarten Merkmalvektoren in jeder Richtung in die Berechnung der Regression ein.

Die Fortschaltzeit bei der Berechnung der Kurzzeitanalysefenster beträgt $T_f = 10$ ms. Bisher ist $M = 4$, das Kontextfenster umfasst also 90 ms des Zeitsignals (vgl. Abschnitt 2.3). Diese Breite liefert nach [Rie94] optimale Erkennungsraten (vgl. Abschnitt 2.4.2), optimiert wird dort aber auf einer Stichprobe mit gelesener Sprache. Da für jedes der 12 statischen Merkmale die Regression berechnet wird, erhält man 12 dynamische Merkmale.

Optimale Fensterbreite

Alle Experimente in dieser Arbeit werden mit einer Stichprobe mit frei gesprochener Sprache durchgeführt. Vermutlich ist das optimale Kontextfenster also kleiner. Untersucht wird nun, für welches M die Wortakkuratheit optimal wird, wenn man die 12 Ableitungen zusätzlich dekorreliert, indem man sie mit der KL-Transformation vollständig entwickelt. In vier verschiedenen

Experimenten mit festem $M \in \{1, 2, 3, 4\}$ werden also 24 dimensionale Merkmalvektoren berechnet und deren 12 dynamische Merkmale transformiert, die 12 statischen aber im Gegensatz zu den Versuchen aus Tabelle 5.1 konstant gelassen. Tabelle 5.3 zeigt, dass sich jetzt für $M = 2$ eine optimale Wortakkuratheit von 72,48 % ergibt.

Motivation des "Multi-Resolution"-Ansatzes

In jedem der folgenden Experimente in diesem Abschnitt werden nun gleichzeitig für z verschiedene M und damit z verschiedene Zeitauflösungen Ableitungen berechnet ($z \in \{1, 2, 3, ...\}$). Der so entstehende hochdimensionale Merkmalvektor hat $F = 12 \cdot z + 12$ Komponenten, es gilt also $F > 24$. Mit der KL-Transformation wird daraus wieder ein 24-dimensionaler Vektor gewonnen. Man versucht so die Vorteile verschiedener Ableitungen zu kombinieren.

Da die Merkmale, die aus der KL-Transformation hervorgehen, ja nur eine Linearkombination der F zu entwickelnden Koeffizienten sind, müssen also, um letztendlich bessere Merkmale zu erhalten, schon die Merkmale, die durch Ableitungen mit $M \neq 4$ Nachbarn entstehen, zusätzliche wichtige Information enthalten. Dies soll zunächst anhand der Abbildungen 5.3 und 5.4 motiviert werden. Beide Abbildungen zeigen die Gesamtenergie im Wort "Bahnhof" sowie Ableitungen über Kontextfenster von 50 ms bzw. 170 ms Breite. Im Voraus kann man nicht entscheiden, welche der Ableitungen besser ist. Sofort fällt aber auf, dass in jeder der Abbildungen beide Ableitungen sehr stark korreliert sind, die KL-Transformation also gerechtfertigt ist. Die Regression für größere M ist eine Glättung der Regression für kleineres M, sie enthält also weniger detaillierte Information über die Steigung der Gesamtenergie. Bei zu großem M wird die Ableitung konstant Null sein und somit gar keine Informationen mehr über den Zeitverlauf des Merkmals wiedergeben.

Betrachtet man Abbildung 5.3, so erkennt man aber, dass auch die gröber berechnete Regression über 170 ms breite Kontextfenster wesentliche Informationen enthält. Sie hat nur noch genau drei relative Maxima in den energiereichen Vokalen und dem Frikativ /f/. Die Vokale beginnen bei maximaler Steigung der Energie (den relativen Maxima der Ableitung) und enden mit maximalem Gefälle. Die Energieschwankungen im Bereich des /n/ und des /h/ werden vollständig eliminiert.

Abbildung 5.4 zeigt dieselben Merkmale für das Wort "Bahnhof", welches jedoch diesmal von einem anderen Sprecher gesprochen und stark verschliffen artikuliert ist. Vergleicht man die Beobachtungen aus Abbildung 5.3 mit dieser Graphik, so erkennt man, dass die beiden Ableitungen über 50 ms Kontextfenster sehr verschieden aussehen, die für 170 ms Kontextfenster jedoch ähnlich sind. Bei größerem Fenster werden also wohl allgemeinere Kontextinformationen, die für beide Sprachsignale gelten, extrahiert und speziellere Informationen vernachlässigt.

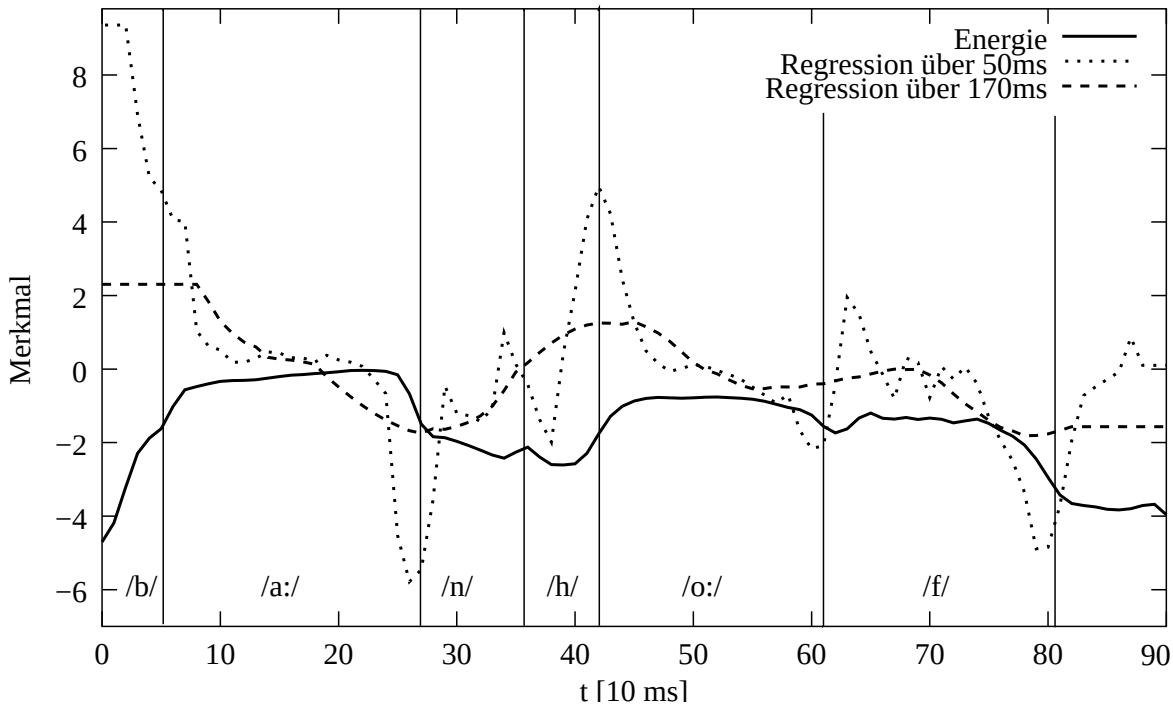

Bild 5.3: Gesamtenergie und Ableitungen des Wortes “Bahnhof” (I)

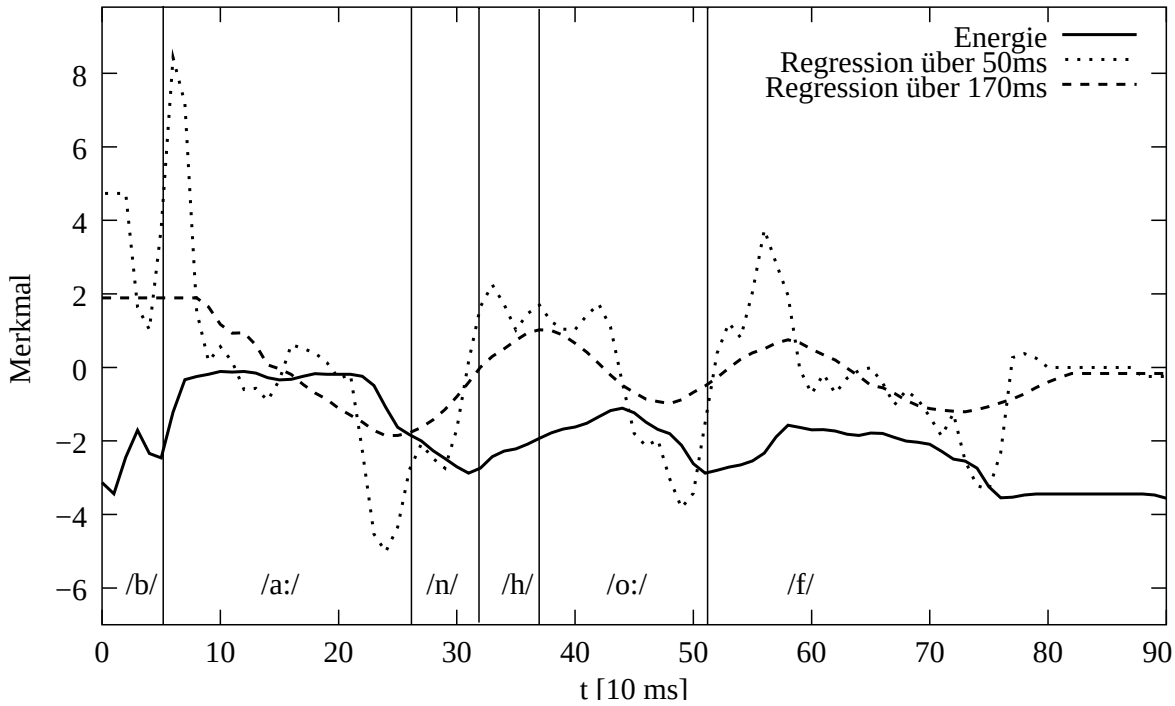

Bild 5.4: Gesamtenergie und Ableitungen des Wortes “Bahnhof” (II)

Ableitungen über diese Fensterbreiten	**# Nachbarn** pro Richtung ($= M$)	**# Merkmale** KLT: input $\rightarrow$ output	**WA** in %
90 ms wie bisher	4		69.89
50 ms	2	$12 \rightarrow 12$	72.48
30, 50 ms	1, 2	$24 \rightarrow 12$	73.25
30, 50, 70 ms	1, 2, 3	$36 \rightarrow 12$	**73.78**
30, 50, 70, 90 ms	1, 2, 3, 4	$48 \rightarrow 12$	73.43
30, 50, 70, 90, 110 ms	1, 2, ..., 5	$60 \rightarrow 12$	73.26
30, 50, 70,... ,130 ms	1, 2, ..., 6	$72 \rightarrow 12$	73.13
30, 50, 70,... ,170 ms	1, 2, ..., 8	$96 \rightarrow 12$	71.81
30, 50, 70,... ,210 ms	1, 2, ..., 10	$120 \rightarrow 12$	72.02
30, 50, 70,... ,250 ms	1, 2, ..., 12	$144 \rightarrow 12$	70.95

Tabelle 5.4: Ableitung über verschiedene Kontextfenster; KLT nur mit den dynamischen Merkmalen

Sehr schön ist auch zu sehen, dass die Regression über 50 ms in Abbildung 5.4 drei Maxima im Bereich zwischen $t = 30$ ms und $t = 45$ ms aufweist, während die Ableitung über 170 ms hingegen genau die wichtige Stelle betont, nämlich die zu Beginn des Lautes /o:/, welche auch in Abbildung 5.3 von beiden Regressionen sehr stark hervorgehoben wird.

KLT nur mit den dynamischen Merkmalen

In Experimenten werden jeweils gleichzeitig für eine unterschiedliche Anzahl z von Auflösungen Ableitungen berechnet. Es ergibt sich ein $F = 12 \cdot z + 12$ dimensionaler Merkmalvektor aus $12z$ dynamischen uns 12 statischen Merkmalen. Nun werden die $12z$ Ableitungen mit der KLT auf 12 Merkmale reduziert und die statischen Merkmale, also die Gesamtenergie und die 11 Mel-Cepstrum Merkmale, unverändert gelassen, da diese Merkmale ja bereits relativ unkorreliert sind (die Kosinustransformation lässt sich als Hauptachsentransformation interpretieren) und sehr gute Ergebnisse in der Spracherkennung liefern. Die Korrelation zwischen den statischen und den dynamischen Merkmalen wird also ignoriert. Die Mel-Cepstrum Merkmale werden so auch davor geschützt, eliminiert zu werden.

Für z.B. $M = 1, 2, ..., 8$, also Kontextfenstern von 30 ms bis 170 ms Breite, erhält man 96 Ableitungsmerkmale. Nur diese 96 Merkmale werden mit der KL-Transformation auf eben 12 Merkmale reduziert. Zusammen mit den unveränderten 12 statischen Merkmalen erhält man wieder 24-dimensionale Merkmalvektoren. In diesem Beispiel kann die Erkennungsrate auf 71.81 % verbessert werden (Tabelle 5.4).

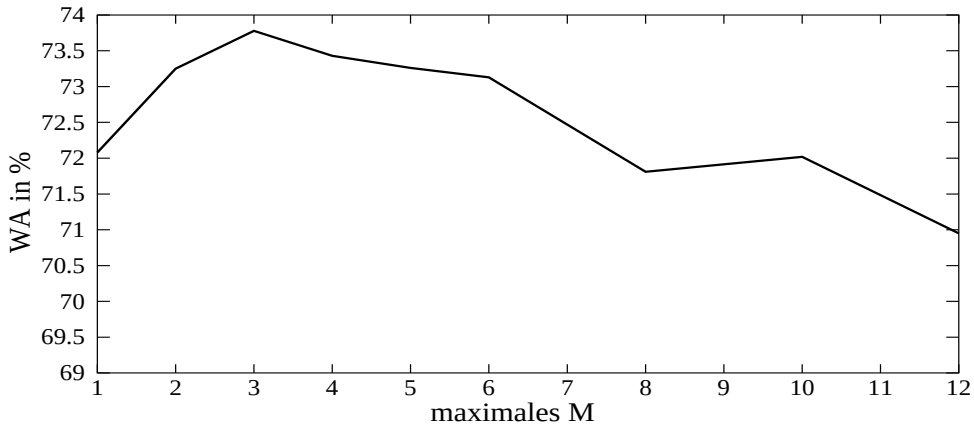

Bild 5.5: Graphische Darstellung der Wortakkuratheiten aus Tabelle 5.4

Um die Wortakkuratheit zu optimieren, werden nun die jeweils maximalen Werte von M verändert. Diese Experimente sind in Tabelle 5.4 beschrieben und in Abbildung 5.5 illustriert. Ein Optimum wurde für $M = 1, 2, 3$ gefunden, also bei Kontextfenstern von 30 bis 70 ms. Die Wortakkuratheit beträgt hier 73,78 %, was eine wesentliche Verbesserung der mit den bisherigen Merkmalen erreichten Wortakkuratheit von 69,89 % bedeutet. Durch Hinzunahme zusätzlicher Ableitungen selektiert die KL-Transformation die für die Erkennung wichtigen Merkmale nicht mehr robust genug; die Wortakkuratheit wird schlechter.

Das optimale Ergebnis erscheint sehr plausibel, wenn man Tabelle 5.3 betrachtet, in der als optimale Fensterbreite für die Berechnung von dekorrelierten Ableitungsmerkmalen 50 ms, also $M = 2$, gefunden wurde. So wurden 72.48 % WA erreicht. Wenn nun zusätzlich noch die Regressionen für $M = 1$ und $M = 3$ in die Berechnungen mit einfließen, verbessert sich die Wortakkuratheit weiter um über ein Prozent. Da aber a priori nur bekannt war, dass für nicht KL-transformierte Ableitungen Fenster von 90 ms Breite optimal sind, war es bei der Suche nach der maximalen Wortakkuratheit keineswegs naheliegend, dass eben diese Fensterbreite ganz unberücksichtigt bleiben kann.

Abbildung 5.6 zeigt das erste dynamische Merkmale im Wort “Bahnhof”, das nach KL-Transformation der 12 Ableitungen für $M = 2$ entsteht und im Vergleich gestrichelt das erste dynamische Merkmal, das durch KL-Transformation der 36 Ableitungsmerkmale aus den drei optimalen Zeitauflösungen ($M = 1, 2, 3$) hervorgeht. Im Vergleich dazu ist das Spektrogramm aus Bild 2.4 abgebildet. Man erkennt, dass durch den “Multi-Resolution”-Ansatz die Übergänge zwischen den Lauten deutlicher werden.

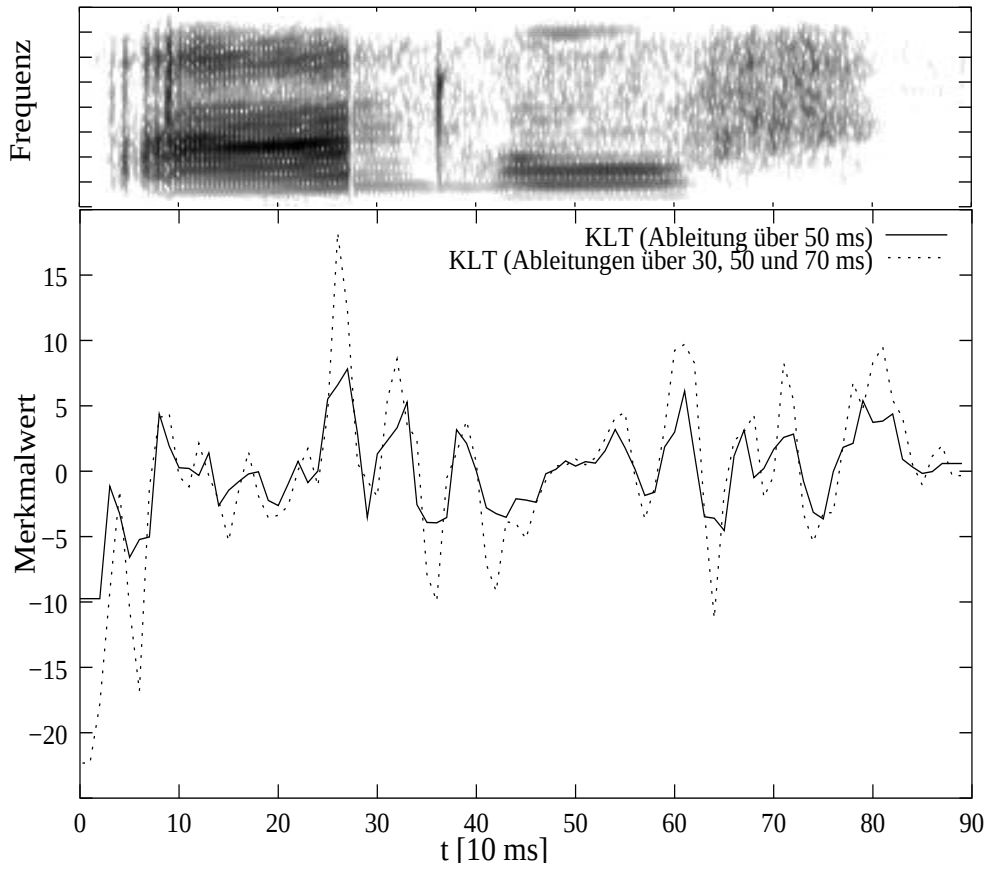

Bild 5.6: Das erste dynamische Merkmal nach KLT der Ableitungen einer bzw. dreier Zeitauflösungen im Vergleich

Ableitungen über diese Fensterbreiten	**# Nachbarn** pro Richtung ($= M$)	**# Merkmale** KLT: input $\rightarrow$ output	**WA** in %
30, 50, 70 ms	1, 2, 3	$48 \rightarrow 24$	**72.68**
30, 50, 70, ... ,170 ms	1, 2, ..., 8	$108 \rightarrow 24$	69.96
50, 90, 130, 170, 210 ms	2, 4, ..., 10	$72 \rightarrow 24$	69.36

Tabelle 5.5: Ableitung über verschiedene Kontextfenster; KLT mit allen (auch den statischen) Merkmalen

Ableitungen über diese Fensterbreiten	**# Nachbarn** pro Richtung ($= M$)	**# Merkmale**	**WA** in %
30, 50 ms	1, 2	36	72.92
30, 50, 70 ms	1, 2, 3	48	**73.99**
30, 50, 70, 90 ms	1, 2, 3, 4	60	72.85
30, 50, 70, 90, 110 ms	1, 2, 3, 4, 5	72	72.16

Tabelle 5.6: Ableitung über verschiedene Kontextfenster; PPCA mit allen (auch den statischen) Merkmalen

KL-Transformation aller Merkmale

In weiteren Experimenten (Tabelle 5.5) wird versucht, alle $F = 12z + 12$ Merkmale, nämlich die $12z$ Ableitungen *und* die 12 statischen Merkmale, mit der KL-Transformation auf 24 Merkmale zu reduzieren. Für z.B. $M = 1, 2, ..., 8$, also Kontextfenstern von 30 bis 170 ms, erhält man wieder 96 Ableitungsmerkmale. Mit der KL-Transformation werden nun 108 Merkmale (96 Ableitungs- + 12 statische Merkmale) auf 24 Merkmale reduziert. Die Wortakkuratheit von 69,96 % liegt aber unter der, die im vergleichbaren oben beschriebenen Experiment erzielt wurde (71,81 %; Tabelle 5.4). Ähnliches zeigt sich auch bei anderen Kombinationen von Fensterbreiten, weshalb diese Vorgehensweise nicht weiter verfolgt wird. Insbesondere bei der gleichzeitigen Berechnung der Regression für 30 ms, 50 ms und 70 ms — die Kombination, die sich oben als optimal erwiesen hat und 73,78 % Wortakkuratheit geliefert hat — werden hier nur 72,68 % erreicht. Immerhin ist dieses Ergebnis aber deutlich besser als jenes, was mit den ursprünglichen Merkmalen aus Abschnitt 2.3 erzielt wird (69,89 %) und auch besser als die 71.27 % WA aus Tabelle 5.1, die nach KL-Transformation aller dieser unveränderter Merkmale erzielt wurden.

Probabilistic PCA

Ähnlich wie in den Experimenten aus Tabelle 5.5 werden bei abschließenden Untersuchungen mit der “Probabilistic Principal Component Analysis” (PPCA) die statischen und dynamischen Merkmale nicht getrennt behandelt. Statt einer Reduktion der Merkmaldimension wird hier ein Erkenner mit den hochdimensionalen Merkmalvektoren trainiert. Diese hochdimensionalen und stark korrelierten Merkmalvektoren werden aber durch spezielle Ausgabeverteilungen der HMMs repräsentiert (vgl. Abschnitt 3.3). Tabelle 5.6 zeigt die Ergebnisse dieser Experimente, die ähnlich verlaufen wie die in Abbildung 5.5 gezeigten Wortakkuratheiten. Vermutet wurde, dass zwar bessere Ergebnisse erzielt werden als in den Experimenten in denen *alle* Merkmale mit der KLT transformiert wurden (Tabelle 5.5) aber schlechtere als jene Ergebnisse, die aus

der Trennung von statischen und dynamischen Merkmalen resultieren (Tabelle 5.4). Jedoch werden bei der optimalen Kombination aus Regressionen für 30 ms, 50 ms und 70 ms die obigen Experimente sogar alle übertroffen: Es wird eine Wortakkuratheit von 73.99% erreicht.

Ein Abschlußexperiment mit der großen Stichprobe

Abschließend wird nun eines der erfolgreichen Experimente aus diesem Abschnitt noch mit der großen Stichprobe, die in Abschnitt 4.1 beschrieben ist, durchgeführt. Bei getrennter Behandlung statischer und dynamischer Merkmale wie in Tabelle 5.4 werden im Experiment mit Ableitungen über 30 ms, 50 ms und 70 ms 74.61 % WA erzielt, in einem Vergleichsexperiment mit unveränderten Merkmalen nur 71.97 % WA. Dies bestätigt die bisherigen Erfolge mit dem "Multi-Resolution"-Ansatz.

Der Vorteil des Berechnens von Ableitungen in verschiedenen Zeitauflösungen lässt sich folgendermaßen zusammenfassen: Zum einen wird die optimale Auflösung durch zusätzliche Informationen ergänzt. Zum anderen kann eben diese optimale Auflösung für unterschiedliche Äußerungen verschieden sein; im Merkmalvektor, der aus unterschiedlichen Auflösungen resultiert, wird die wichtige Information aber wahrscheinlicher enthalten sein.

5.2.2 Kontext durch Konkatenation

Da die Regression ja nur eine Linearkombination von zeitlich benachbarten Merkmalvektoren ist, ist es eine naheliegende Idee, einfach diese Nachbarn mit dem aktuellen Merkmalvektor zu konkatenieren und aus diesem hochdimensionalen Vektor dann die wichtige Information mit der KL-Transformation zu extrahieren.

Prinzipiell ist dies dieselbe Vorgehensweise, wie sie auch S. Rieck in [Rie94, S.129 ff] beschrieben hat (siehe Abschnitt 2.4.2). In seinen Experimenten hat er aber die LDA verwendet, die in seiner Implementierung auf einem numerisch instabilen Algorithmus zur Eigenwertberechnung basiert. Im folgenden wird jedoch der in Kapitel 3 beschriebene Algorithmus zur KL-Transformation verwendet.

Zur einfacheren Beschreibung der unterschiedlichen Experimente werden nun diese Bezeichnungen eingeführt (vgl. Abbildung 5.7): Der aktuelle Merkmalvektor setzt sich wie in Abschnitt 2.3 beschrieben aus 12 statischen Merkmalen zusammen, die im Vektor $\mathbf{s}_0$ zusammengefasst sind, und 12 Ableitungen, die mit $\mathbf{a}_0$ bezeichnet werden. Die Merkmale des benachbarten vorangegangenen Vektors spalten sich in $\mathbf{s}_{-1}$ und $\mathbf{a}_{-1}$ auf, die des nachfolgenden in $\mathbf{s}_1$ und $\mathbf{a}_1$. Berücksichtigt man nun einen Kontext von je M Merkmalvektoren in jede Zeitrichtung, so liegen die Vektoren $\mathbf{s}_\tau$ und $\mathbf{a}_\tau$ für $\tau = 0, \pm1, \pm2, ..., \pm M$ im aktuellen Kontextfenster.

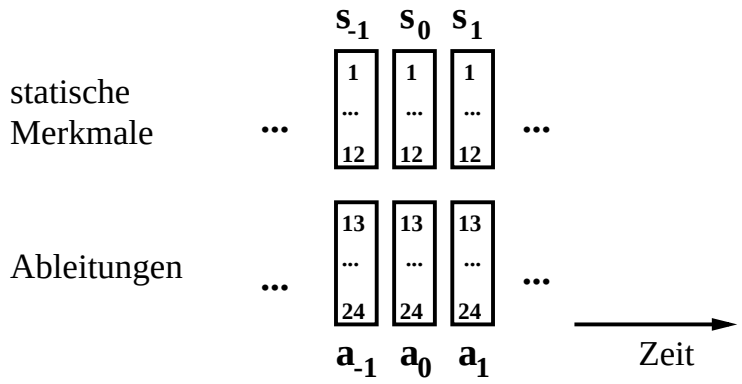

Bild 5.7: Eine zeitliche Folge von Merkmalvektoren

Merkmale	**# Nachbarn** $2 \times M$	**# Merkmale** KLT: input → output	**WA** in %
$\mathbf{s}_0, \mathbf{a}_0$ (bisherige Merkmale)	0		69.89
$KL(\mathbf{s}_0, \mathbf{a}_0)$	0	24 → 24	71.27
$KL(\mathbf{s}_0, \mathbf{s}_{\pm1}, \mathbf{s}_{\pm2}, \mathbf{a}_0, \mathbf{a}_{\pm1}, \mathbf{a}_{\pm2})$	2×2	120 → 24	70.74
$KL(\mathbf{s}_0, \mathbf{s}_{\pm1}, \mathbf{s}_{\pm2}, \mathbf{s}_{\pm3}, \mathbf{a}_0, \mathbf{a}_{\pm1}, \mathbf{a}_{\pm2}, \mathbf{a}_{\pm3})$	2×3	168 → 24	69.53
$KL(\mathbf{s}_0, \mathbf{s}_{\pm1}, \ldots, \mathbf{s}_{\pm6}, \mathbf{a}_0, \mathbf{a}_{\pm1}, \ldots, \mathbf{a}_{\pm6})$	2×6	312 → 24	68.32
$\mathbf{s}_0 + KL(\mathbf{s}_{\pm1}, \mathbf{a}_0, \mathbf{a}_{\pm1})$	2×1	60 → 12	**71.16**
$\mathbf{s}_0 + KL(\mathbf{s}_{\pm1}, \mathbf{s}_{\pm2}, \mathbf{a}_0, \mathbf{a}_{\pm1}, \mathbf{a}_{\pm2})$	2×2	108 → 12	71.12
$\mathbf{s}_0 + KL(\mathbf{s}_{\pm1}, \mathbf{s}_{\pm2}, \mathbf{s}_{\pm3}, \mathbf{a}_0, \mathbf{a}_{\pm1}, \mathbf{a}_{\pm2}, \mathbf{a}_{\pm3})$	2×3	156 → 12	70.77
$\mathbf{s}_0 + KL(\mathbf{s}_{\pm1}, \ldots, \mathbf{s}_{\pm4}, \mathbf{a}_0, \mathbf{a}_{\pm1}, \ldots, \mathbf{a}_{\pm4})$	2×4	204 → 12	68.76
$\mathbf{s}_0 + KL(\mathbf{s}_{\pm1}, \mathbf{s}_{\pm2})$	2×2	48 → 12	63.01
$\mathbf{s}_0 + KL(\mathbf{s}_{\pm1}, \mathbf{s}_{\pm2}, \mathbf{s}_{\pm3})$	2×3	72 → 12	62.78
$\mathbf{s}_0 + KL(\mathbf{s}_{\pm1}, \mathbf{s}_{\pm2}, \mathbf{s}_{\pm3}, \mathbf{s}_{\pm4})$	2×4	96 → 12	63.31
$\mathbf{s}_0 + KL(\mathbf{s}_{\pm1}, \mathbf{s}_{\pm2}, \ldots, \mathbf{s}_{\pm8})$	2×8	192 → 12	65.40

Tabelle 5.7: Kontext durch Konkatenation

Im ersten Teil von Tabelle 5.7 sind Experimente aufgelistet, in denen erst alle $\mathbf{s}_\tau$ und $\mathbf{a}_\tau$ konkateniert werden und danach die KL-Transformation angewendet wird. Für $M = 2$ Nachbarn auf jeder Seite kann die Wortakkuratheit auf 70,74 % gegenüber den 69,89 % mit den in Kapitel 2.3 beschriebenen Merkmalen verbessert werden. Für größere Kontexte nimmt die Wortakkuratheit ab.

Ähnlich wie im letzten Abschnitt, kann die Wortakkuratheit wieder verbessert werden, wenn man die aktuellen statischen Merkmale $\mathbf{s}_0$ von der KL-Transformation ausklammert (Tabelle 5.7, 2. Teil). Für z.B. $M = 2$ Nachbarn auf jeder Seite kann so eine Wortakkuratheit von 71,12 % erzielt werden.

Das Phänomen, dass die Wortakkuratheit für größere Kontexte wieder abnimmt, resultiert wohl aus der kombinierten Anwendung von Konkatenation und Ableitungen. Die Ableitungen, die für ein Zeitfenster berechnet werden, das etliche Millisekunden entfernt ist, verschlechtern wohl die Erkennungsraten für das aktuelle Fenster. In den im letzten Teil der Tabelle 5.7 aufgelisteten Experimenten werden deshalb die $\mathbf{a}_\tau$ gar nicht mehr berücksichtigt. Die KL-Transformation wird nur auf die statischen Merkmale der Nachbarn angewandt, die aktuellen statischen Merkmale bleiben wieder ausgeklammert. Nun kann man eine Verbesserung der Wortakkuratheit mit wachsendem Kontext, wie in [Rie94] beschrieben, erkennen, jedoch ist diese sehr gering und liegt bei $M = 8$ erst bei 65,40 %.

Zusammenfassend lässt sich sagen, dass bei keinem der Experimente in diesem Abschnitt die Ergebnisse aus dem letzten Abschnitt übertroffen werden konnten, nicht einmal aber die Wortakkuratheit von 71.27 %, die durch KL-Transformation nur der bisherigen Merkmale erzielt wurde. Grund dafür ist wohl, dass die KLT aus Vektoren hoher Dimension die zur Erkennung wichtige Information nicht mehr extrahieren kann.

5.3 Statische Merkmale bei verschiedenen Zeitauflösungen

In diesem Abschnitt wird versucht, die zwölf statischen Merkmale zu optimieren, indem man sie für verschiedene Zeitauflösungen berechnet. Die zwölf dynamischen Merkmale bleiben konstant, d.h. es werden die Ableitungen der bisherigen unveränderten statischen Merkmale in einfacher Auflösung beibehalten, und die Ergebnisse aus dem letzten Abschnitt wieder verworfen, um die Experimente besser vergleichen zu können. In den anderen Unterabschnitten wird erst die Kosinustransformation durch die Karhunen-Loève-Transformation ersetzt und danach sogar direkt das Spektrum mit KLT transformiert.

5.3.1 Das Spektrum für verschiedene Zeitauflösungen

Wie in Abschnitt 2.3 ausführlich beschrieben, werden bei der Kurzzeitanalyse 16 ms breite Fenster des Signals betrachtet, was 256 Abtastwerten entspricht. Die Fortschaltzeit beträgt 10 ms; die Fenster überlappen also. In diesem Abschnitt wird die Fensterbreite nun variiert, das Spektrum also für verschiedene Zeitauflösungen berechnet. Aus jedem dieser Spektren wird genauso wie in Abschnitt 2.3 das Mel-Cepstrum berechnet, samt anschließender dynamisch adaptiver cepstraler Subtraktion (DACS), zeitlicher Energie-Glättung und Tirol-Filterung. Jede Auflösung liefert also elf Mel-Cepstrum Koeffizienten und die Energie. Bei Berechnung des Cepstrums aus z verschiedenen Zeitauflösungen erhält man folglich $12 \cdot z$ statische Merkmale sowie die 12 Ableitungen. Da sich die getrennte Behandlung beider Merkmalgruppen im letzten Abschnitt bewährt hat, werden auch hier nur die $12 \cdot z$ statischen Merkmale mit der KLT auf 12 Merkmale reduziert und mit den unveränderten dynamischen Merkmalen zu einem 24-dimensionalen Merkmalvektor konkateniert.

Motivation

Nimmt man statt der 16 ms Analysefenster 32 ms, so wird die Zeitauflösung verschlechtert und nach dem Unschärfeprinzip die Frequenzauflösung verbessert. 16 ms entsprechen 256 und 32 ms entsprechen 512 Abtastwerten. Bei Zeitbereichsfenstern der Länge 128 hingegen verschlechtert sich die Auflösung im Spektrum. Die verschiedenen Frequenzauflösungen im logarithmierten Spektrum zeigt die linke Spalte aus Abbildung 5.8.

Durch paralleles Betrachten mehrerer Zeitauflösungen erwartet man, zusätzliche Informationen zu erhalten. Da aber das Spektrum mit der Mel-Filterbank aus Abbildung 2.5, Abschnitt 2.1 wieder zusammengefasst wird, geht diese zusätzliche Information wohl wieder verloren. Dies erzwingt als weiteres Vorgehen, eine optimale Mel-Filterbank zu finden, was jedoch über den Rahmen dieser Arbeit hinausgehen würde. Als Anregung dienen die Experimente von S. Rieck [Rie94] (siehe Abschnitt 2.4, Tabelle 2.1), in denen tatsächlich auch bei den bisher verwendeten Merkmalen in einfacher Zeitauflösung durch eine Filterbank mit 32 statt 18 Frequenzgruppen bessere Erkennungsraten erzielt werden konnten.

Abbildung 5.9 zeigt, dass die Mel-Cepstra für 256, 512 und 128 Abtastwerte, d.h. 16, 32 und 8 ms Zeitfenster, sehr stark korreliert sind. Es fällt auf, dass beim 8 ms Fenster die scharfe Trennung der Laute /n/ und /h/ eliminiert wird. Dennoch wird experimentell überprüft, ob die wenige zusätzliche Information eine Verbesserung der Erkennungsraten verursacht.

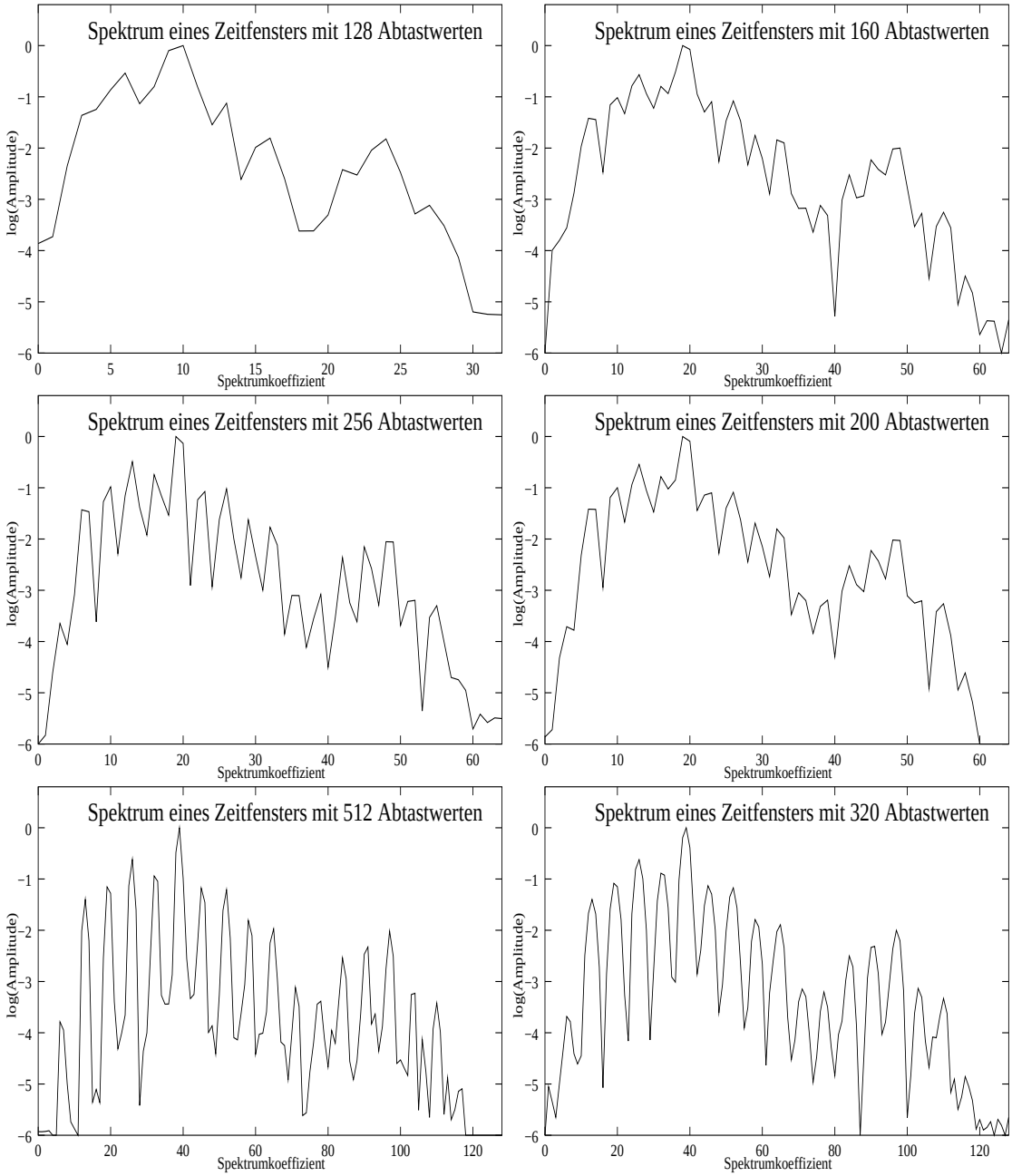

Bild 5.8: Die logarithmierten Spektren von unterschiedlich breiten Zeitfenstern des Vokals /a/. Da es sich um Telefonsprache handelt, sind nur die Koeffizienten aus dem Bereich 0-4 kHz abgebildet.

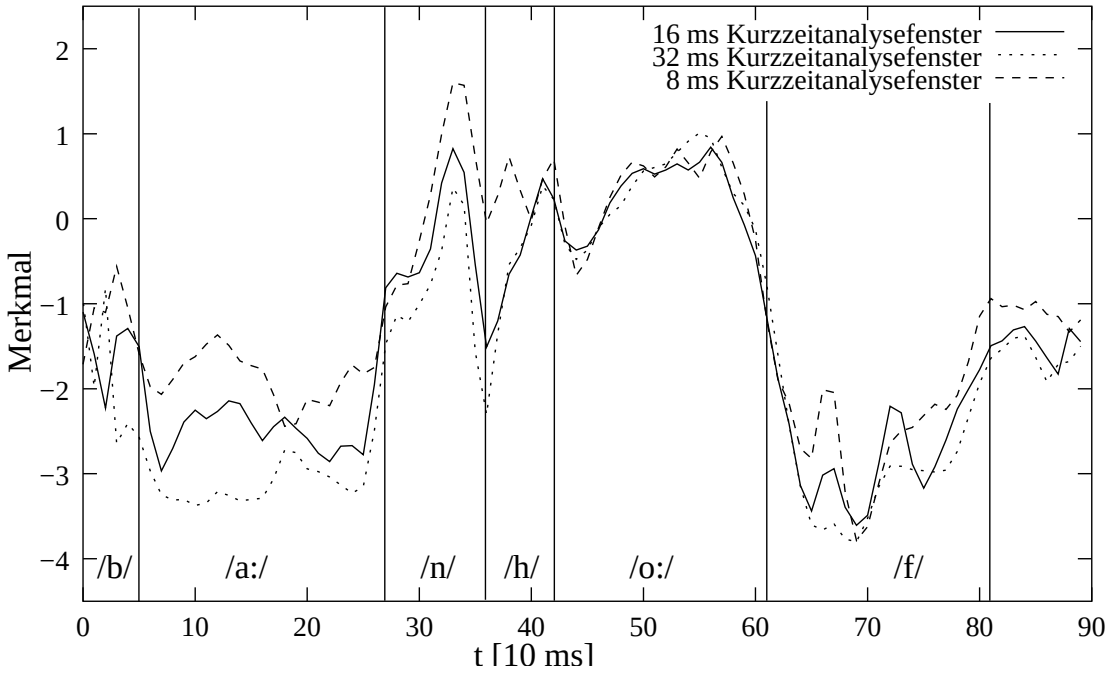

Bild 5.9: Cepstrum-Koeffizient 3 für verschiedene Zeitauflösungen

Fensterbreiten zur Kurzzeitanalyse in Frames (und ms)					**# Merkmale**	**WA**
32 (2)	64 (4)	128 (8)	256 (16)	512 (32)	KLT: input → output	in %
			X		—	69.89
			X		12 → 12	70.41
X			X		24 → 12	68.90
		X	X		24 → 12	69.11
	X	X	X		36 → 12	70.13
X	X	X	X		48 → 12	69.77
		X	X	X	36 → 12	68.11
			X	X	24 → 12	70.18

Tabelle 5.8: Berechnung der MFCCs über verschiedene Fensterbreiten und anschließend KLT

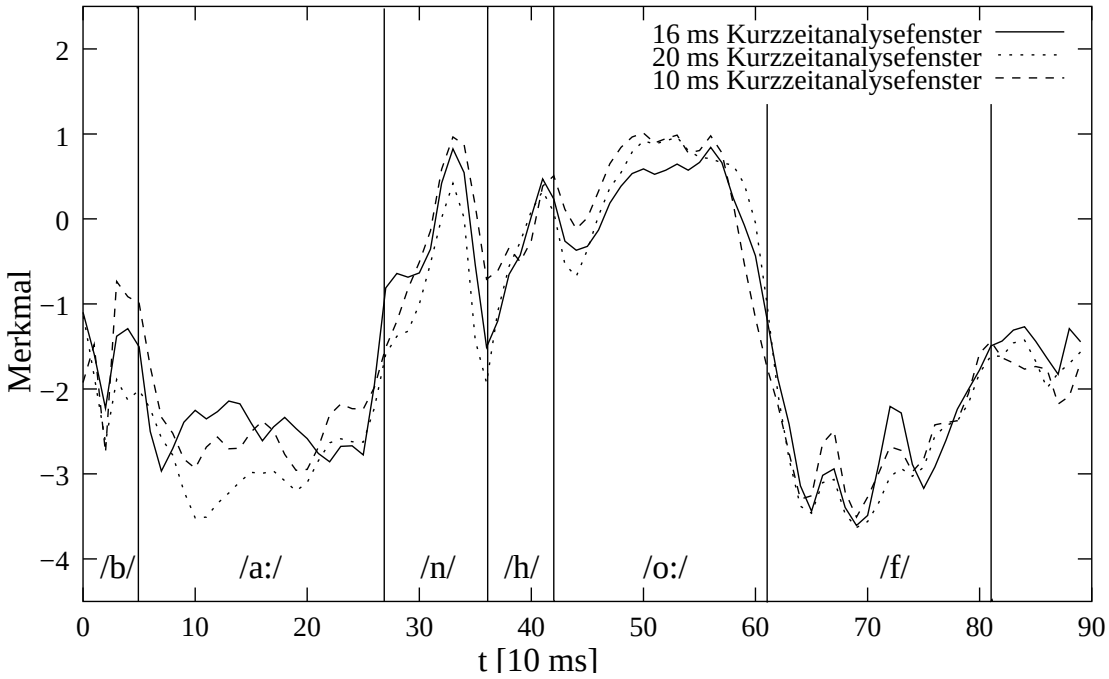

Bild 5.10: Cepstrum-Koeffizient 3 für verschiedene Zeitfenster, deren Breiten keine Zweierpotenzen von Abtastwerten sind

Experimente und Ergebnisse

Bei Kombination der Merkmale aus verschiedenen Auflösungen wird nach Berechnung der Hauptkomponenten der Merkmale aus 16 und 32 ms breiten Fenstern eine Wortakkuratheit von 70.18 % erreicht. Es liegt zwar eine geringe Verbesserung gegenüber den 69.89 % WA vor, die mit den ursprünglichen Merkmalen (Abschnitt 2.3) erzielt wurden. Transformiert man aber zum Vergleich in einem Experiment nur jene statischen Merkmale aus 16 ms Analysefenstern mit der KLT und konkateniert sie anschließend mit den unveränderten Ableitungen, so werden 70.41 % WA erzielt (Tabelle 5.8, oberer Teil). Der “Multi-Resolution”–Ansatz bringt hier also keine Vorteile. Auch andere Kombinationen mit Kurzzeitanalysefenstern, deren Breite eine Zweierpotenz von Abtastwerten ist, wie es von der FFT gefordert wird, bringen keine Erfolge. Alle durchgeführten Experimente sind in Tabelle 5.8 zusammengestellt.

Um weitere Experimente ausführen zu können, werden nun auch die Spektren aus Zeitfenstern, deren Breite in Abtastwerten keine Zweierpotenz ist, berechnet. Dazu wird das jeweilige Fenster wie gehabt mit einem Hamming-Fenster gefiltert und anschließend links und rechts mit Nullen ergänzt, so dass eine Zweierpotenz von Werten vorliegt. Nun ist der Ausschnitt aus dem Zeitsignal zwar etwas verfälscht, aber für die FFT geeignet. Beispiele für solche Spektren sind in Abbildung 5.8 rechte Spalte illustriert. Die Spektren sind untereinander sehr stark korreliert,

Fensterbreiten zur Kurzzeitanalyse in Frames (oben) und ms (unten)												**WA**
32	64	80	100	128	160	200	256	300	320	360	512	in %
2	4	5	6.25	8	10	12.5	16	18.75	20	22.5	32	
				X			X					69.11
X	X			X			X					69.77
	X			X			X					70.13
		X	X	X	X	X	X					69.93
			X	X		X	X					69.96
	X		X	X		X	X					69.74
	X	X		X	X		X					69.03
				X			X				X	68.11
			X	X		X	X		X			69.29
		X			X		X		X			**71.79**
	X	X		X	X		X		X			69.39
							X	X				69.18
							X		X			68.75
							X			X		69.56
							X				X	70.18

Tabelle 5.9: Berechnung der MFCCs über verschiedene Fensterbreiten und anschließend KLT. Die dynamischen Merkmale bleiben unverändert.

insbesondere ein Spektrum aus n Abtastwerten, mit $2^{k-1} < n < 2^k$ für ein $k \in \mathbb{N}$, mit dem Spektrum aus 2^k Abtastwerten. Die resultierenden Cepstra für 256, 320 und 160 Abtastwerte, d.h. 16, 20 und 10 ms, sind in Abbildung 5.10 zusammengestellt.

Nun werden verschiedene Cepstra aus verschiedenen Zeitauflösungen kombiniert. Leider gibt es dabei eine riesige Anzahl an Möglichkeiten; eine bestimmte Suchrichtung ist durch Vergleichen der Teilergebnisse nicht ersichtlich. Zu Beachten ist lediglich, dass bei zu großer Anzahl von Merkmalen die KLT die für die Erkennung wichtigen Merkmale nicht mehr robust genug herausfiltert. Auch scheinen Fenster kleiner als 64 ms die Erkennungsraten zu verschlechtern. Tabelle 5.9 zeigt nun die verschiedenen Experimente samt der erzielten Wortakkuratheiten im Vergleich. Durch Hinzunahme von ausschließlich kleineren bzw. ausschließlich größeren Fenstern wird die Wortakkuratheit nicht verbessert; bei kleineren und größeren Fenstern gemischt werden in einem Fall 71.79 % WA erreicht. In diesem Fall werden Spektren aus Fenstern der Länge 5, 10, 16 und 20 ms berechnet. Schon nach Hinzunahme weniger benachbarter Fensterlängen kann dieses Ergebnis nicht mehr erzielt werden.

Kombiniert man dieses Experiment mit dem erfolgreichen Experiment aus Abschnitt 5.2.1 und berechnet zusätzlich die Hauptkomponenten der Ableitungen über 30 ms, 50 ms und 70 ms, können jedoch nur 71.88 % WA erreicht werden. Die Erfolge beider Experimente addieren sich also nicht, sondern schwächen sich gegenseitig.

Abschließend wird in einem gesonderten Experiment mit der großen Stichprobe, die in Abschnitt 4.1 beschrieben ist, der Erfolg des Versuchs mit Spektren aus Fenstern der Länge 5, 10, 16 und 20 ms überprüft. Es werden 72.16 % WA erzielt, in einem Vergleichsexperiment mit unveränderten Merkmalen fast genauso viel (71.97 % WA). Der Erfolg kann also nicht bestätigt werden.

5.3.2 Ersetzen der Kosinustransformation durch KLT

Es folgt nun ein Versuch, die Ergebnisse aus dem letzten Abschnitt zu verbessern. Dazu wird die Idee herangezogen, anstelle der Diskreten Kosinustransformation (DCT) zur Berechnung des Cepstrums, die Karhunen-Loève-Transformation (KLT) zu verwenden, da beide ja eine Dekorrelierung der Merkmale bewirken [Bat98]. Im obigen Abschnitt wurden für jede Zeitauflösung die zwölf Cepstrum-Koeffizienten mit der DCT berechnet und anschließend alle Cepstra zusammen mit der KLT transformiert und reduziert. Eine Verbesserung verspricht man sich nun, wenn man gleich die logarithmierten Mel-Spektren aus verschiedenen Auflösung gemeinsam mit Hilfe der KLT auf 12 Koeffizienten reduziert.

Dazu wird zunächst versucht, im bisherigen Berechnungsablauf für eine Zeitauflösung (Abschnitt 2.3), die DCT durch die KLT zu ersetzen. In einem ersten Experiment, in dem diese neu erzeugten statischen Merkmalen mit den ursprünglichen 12 Ableitungen konkateniert werden, wird eine Wortakkuratheit von 65.17 % erzielt, beim zusätzlichen Berechnen der 12 Ableitungen dieser neuen Merkmale gar nur 61.39 %.

Es ist an dieser Stelle notwendig alle Berechnungsschritte aus Abschnitt 2.3, die der Mel-Cepstrum-Berechnung folgen, für dieses neue Cepstrum zu optimieren. Dazu gehört, dass die Gesamtenergie während des gesprochenen Satzes auf konstantem Niveau gehalten wird sowie die dynamisch adaptive cepstrale Subtraktion (DACS) der übrigen Cepstrum-Koeffizienten. Die erste durch DCT berechnete Cepstrum-Komponente ist die Gesamtenergie, von den durch KLT erzeugten Koeffizienten ist Nummer 8 mit dieser am stärksten korreliert; der Korrelationskoeffizient beträgt -0.91 (Abbildung 5.11). Allerdings sind auch andere Komponenten mit der Gesamtenergie korreliert, Koeffizient 3 mit -0.87, Koeffizient 10 mit Korrelation 0.70 oder Koeffizient 1 mit -0.6, so dass ein äquivalentes Vorgehen zu dem aus Abschnitt 2.3 kaum möglich ist. Auffallend ist, dass in einem Experiment, in dem DACS sowie die zeitliche Energieglättung ganz weggelassen werden, sogar **65.95 % WA** erzielt werden. In [Bat98] ist die KLT der reinen DCT

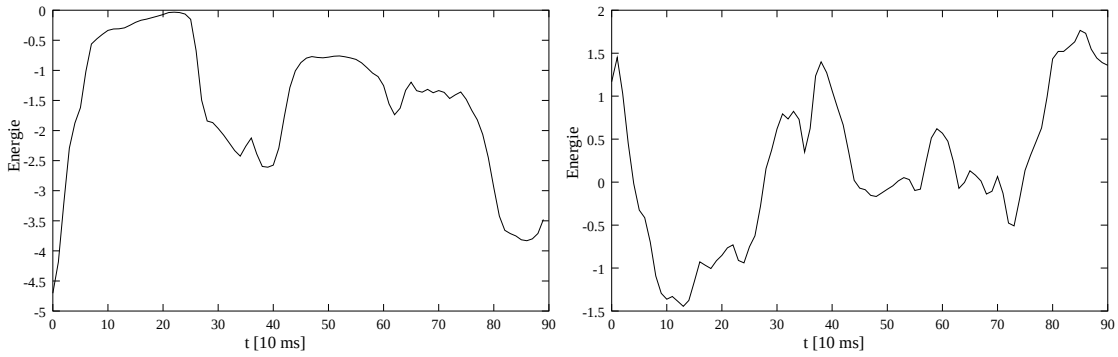

Bild 5.11: Links Das Merkmal “Energie” nach DCT, rechts ein entsprechendes Merkmal nach KLT. Die Korrelation beträgt -0.91

überlegen. Dort werden 16 Mel-Spektrum Koeffizienten berechnet, die auf je 16 dekorrelierte Merkmale transformiert werden.

Experimente für mehrere Zeitauflösungen nach diesem Ansatz wurden exemplarisch nur für die Kombination aus 16 ms und 32 ms breiten Kurzzeitanalysefenstern durchgeführt. Aus diesen Fenstern werden je 18 logarithmierte Mel-Spektrum Koeffizienten und die Energie berechnet, und alle 38 Koeffizienten mit der KLT auf 12 Merkmale reduziert; DACS und zeitlichen Energieglättung werden weggelassen. Als dynamische Merkmale werden die Ableitungen der ursprünglichen statischen Merkmale verwendet. So werden **66.62 % WA** erzielt, durch den “Multi-Resolution”-Ansatz wird das Ergebnis hier also verbessert.

5.3.3 Verzicht auf die Filterbank

Anstelle die Mel-Filterbank zu optimieren, was über den Rahmen dieser Arbeit hinausgehen würde, wird nun der Schritt gewagt, sämtliche ausgeklügelten Schritte zur Merkmalberechnung aus Abschnitt 2.3, insbesondere die Berechnung von Bandspektrum und Mel-Cepstrum, ganz zu überspringen. Es wird jetzt der Karhunen-Loève-Transformation überlassen, allein aus dem logarithmierten Betragsquadratspektrum die wichtige Information herauszufiltern. In einem Experiment mit Kurzzeitanalysefenstern der Breite 256 werden 128 Spektrum-Koeffizienten auf 12 reduziert. In einem “Multi-Resolution”- Experiment werden 384 Spektrum-Koeffizienten, die aus 256 und 512 Abtastwerten großen Fenstern resultieren, ebenfalls auf 12 Merkmale vermindert. Die Wortakkuratheit liegt bei 61.57 % bzw. 58.04 %. Hier zeigt sich wie auch schon in den obigen Abschnitten, dass die KLT aus zu großen Merkmaldimensionen nicht mehr genügend robust die für die Erkennung wichtige Information extrahieren kann.

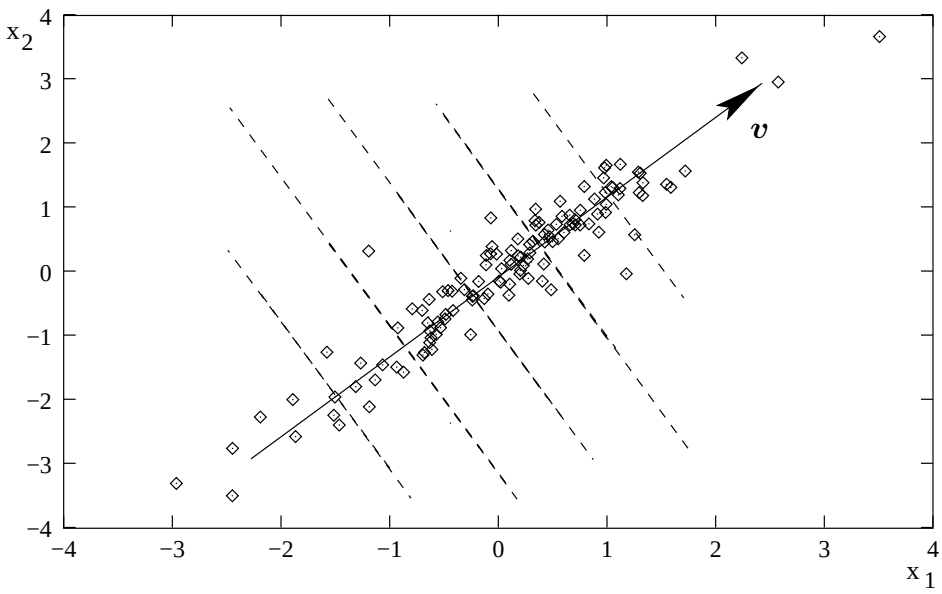

Bild 5.12: Konturlinien konstanter Projektion bei linearer PCA

5.4 Produktterme

In den Versuchen in diesem Kapitel wird die Dimension der Merkmalvektoren zuerst erhöht, indem als zusätzliche Komponenten Produkte der bisherigen Merkmalkomponenten hinzugenommen werden. Anschließend werden die hochdimensionalen Vektoren mit Karhunen-Loève-Transformation (KLT) wieder reduziert. Das gesamte Vorgehen wird in der Literatur auch als nichtlineare Hauptachsentransformation (nonlinear PCA = nonlinear Principal Component Analysis) bezeichnet. Zunächst wird nun das Vorgehen motiviert und theoretische Grundlagen aus der Literatur zitiert. Danach werden die Experimente beschrieben.

5.4.1 Motivation

Idee der Produktterme ist es, neue Merkmale in Betracht zu ziehen, die man durch Multiplikation von Komponenten der ursprünglichen Merkmalvektoren erhält. Die Merkmalvektoren werden also in einen neuen höher dimensionalen Merkmalraum transformiert. Man hofft nun, in diesem Raum Ballungsgebiete von Merkmalvektoren schärfer trennen zu können. Der ursprüngliche Merkmalraum sei wieder f-dimensional, der hochdimensionale Raum F-dimensional mit $F > f$.

Ähnliche Vorgehensweisen findet man in der Literatur: Im folgenden werden in Anlehnung an die Arbeit [Sch98] lineare und nichtlineare PCA beschrieben. Abbildung 5.12 zeigt einen zweidimensionalen Raum mit einer Punktwolke. Die Hauptkomponente mit größtem zugehöri-

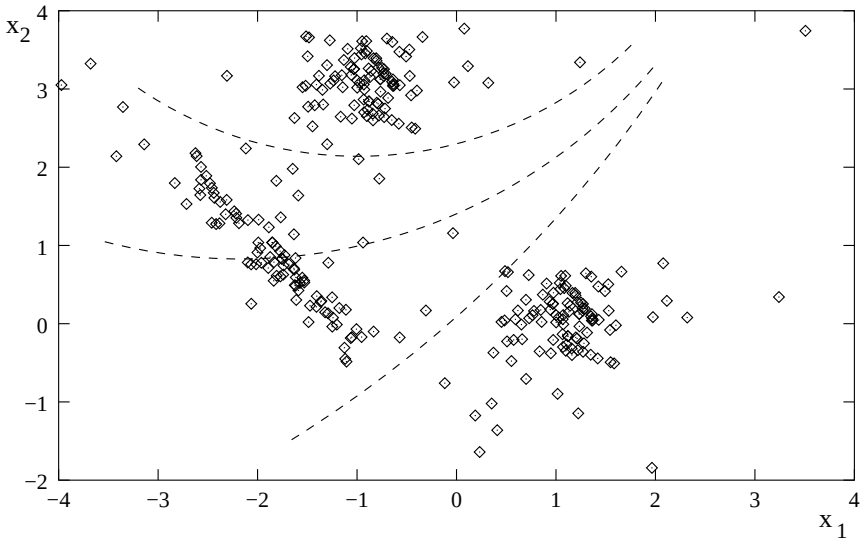

Bild 5.13: Konturlinien konstanter Projektion bei nichtlinearer PCA

gen Eigenwert heißt $\boldsymbol{v}$. Wählt man nun dieses Merkmal $\boldsymbol{v}$ zur Beschreibung der Punkte, wird der zweidimensionale Raum – nach Diskretisierung der $\boldsymbol{v}$-Achse – in feine Streifen zerlegt. Die Konturlinien konstanter Projektion auf die $\boldsymbol{v}$-Achse sind in Abbildung 5.12 verdeutlicht. Möglicherweise ist es aber vorteilhaft, den Raum in anderer Weise zu unterteilen, wie es beispielsweise Abbildung 5.13 zeigt. Ein Lösungsweg ist die nichtlineare PCA: Durch eine nichtlineare Abbildung

$$\Phi : \mathbb{R}^f \rightarrow \mathbb{R}^F \tag{5.1}$$

werden zunächst die Merkmalvektoren durch Hinzunahme von Produkten höherer Ordnung in einen höherdimensionalen Raum abgebildet und dort mit linearer PCA analysiert. Die Konturlinien konstanter Projektion sind im $\mathbb{R}^F$ wieder Geraden, durch Rücktransformation in den ursprünglichen Raum werden sie aber nichtlinear verzerrt. Da die Eigenvektoren im ursprünglichen Raum nicht existieren, sind sie nicht eingezeichnet. Durch eine lineare PCA im F-dimensionalen Raum werden also die Merkmale im ursprünglichen Raum auf nichtlineare Hauptkomponenten hin untersucht.

In [Sch98] wird die Kernel-PCA definiert. Dazu wird von Kernel-Funktionen

$$k(\boldsymbol{x}, \boldsymbol{y}) = \Phi(\boldsymbol{x})^T \Phi(\boldsymbol{y}) \tag{5.2}$$

Gebrauch gemacht, die in ähnlicher Weise auch bei den “Support Vector Maschinen” [Bur98] verwendet werden. Anstelle zwei f-dimensionale Merkmalvektoren $\boldsymbol{x}$ und $\boldsymbol{y}$ erst in einen hochdimensionalen Raum zu transformieren und dann ihr Skalarprodukt zu bilden, wird dazu äquiva-

lent eine Kernel-Funktion k mit den ursprünglichen Vektoren berechnet. Da sich die lineare PCA allein durch Skalarprodukte der Merkmale ausdrücken lässt, braucht man die Merkmalvektoren generell nicht mit Φ zu transformieren.

Durch Bilden von Produkten höherer Ordnung gelangt man sehr schnell in extrem hochdimensionale Räume und an die Grenzen der Rechnerkapazitäten. So werden z.B. in [Sch98] Produkte der Ordnung 5 von 256-dimensionalen Merkmalvektoren berechnet. Man erhält dort 10^{10}-dimensionale Vektoren und wird dadurch gezwungen, die Kernel-PCA zu verwenden.

Wird durch Φ der Vektor $\boldsymbol{x}$ mit den Komponenten x_i ($i = 1, 2, ..., f$) in einen Vektor $\boldsymbol{\xi}$ mit Komponenten ξ_j transformiert, wobei die ξ_j Produktterme der Ordnung p aus den Komponenten x_i sind, so lautet die entsprechende Kernel-Funktion im $\mathbb{R}^f$

$$k(\boldsymbol{x}, \boldsymbol{y}) = (\boldsymbol{x}^T \boldsymbol{y})^p. \tag{5.3}$$

Vorteile der nichtlinearen PCA, die durch die Kernel PCA realisiert werden kann, sind eine bessere Trennbarkeit der Merkmalklassen durch nichtlineare Trennfunktionen, aber auch die Möglichkeit, mehr als f (= Dimension der ursprünglichen Merkmalvektoren) Hauptkomponenten zu entwickeln.

Die Experimente in [Sch98] werden mit einer Stichprobe handgeschriebener Ziffern durchgeführt. Die Grauwerte der 256 Pixel der gerasterten Bilder dienen als Merkmale. In Experimenten, in denen mit nichtlinearer PCA wieder 256 Merkmale erzeugt werden, können bereits durch Ordnung 2 niedrigere Fehlerraten erzielt werden, als bei linearer PCA; die Ergebnisse für höhere Ordnungen sind ähnlich. Wenn sogar bis zu 2048 Merkmale berechnet werden, verbessern sich die Ergebnisse weiter. Eine minimale Fehlerrate erhält man hier mit Produkttermen der Ordnung 5.

5.4.2 Experimente und Ergebnisse

In den Experimenten in diesem Abschnitt wird nun untersucht ob auch in der Spracherkennung Verbesserungen durch Produktterme erzielt werden können. Dazu werden wieder die 24-dimensionalen Merkmalvektoren aus Abschnitt 2.3 herangezogen und Produkte der Ordnung 2 gebildet, also je zwei Merkmalkomponenten des ursprünglichen Vektors multipliziert. Auch in diesem Abschnitt werden die 12 statischen und die 12 dynamischen Merkmale getrennt behandelt. Die Abbildung Φ aus dem letzten Abschnitt transformiert also hier $f = 12$–dimensionale Vektoren in einen F–dimensionalen Raum ($F \in \{78, 90, 144\}$). F variiert in den nachfolgend beschriebenen Experimenten, ist aber noch so klein, dass die Transformation in den hochdimensionalen Raum explizit ausgeführt werden kann. Von der Kernel-PCA wird hier also kein Gebrauch gemacht.

Komponenten des $\mathbb{R}^{24}$	**PCA der** a_i s_i unverändert	**PCA der** s_i a_i unverändert	**PCA der (** a_i *und* s_i **)**
12 statische und 12 dynamische Merkmale (aus 90 ms Kontextfenstern)	69.19 (vgl. Tabelle 5.3)	70.41 (vgl. Tabelle 5.8)	71.27 (vgl. Tabelle 5.1)

Tabelle 5.10: Wortakkuratheit in Prozent nach *linearer* PCA. Die s_i sind die statischen Merkmale, die a_i sind die Ableitungen.

Diejenigen Komponenten der transformierten Merkmalvektoren, die Produkte der ursprünglichen Merkmale sind, haben sehr große Werte und sind sehr weit gestreut. Deshalb werden sie so normiert, dass die Varianz in Richtung jeder Achse eins ist.

Die Wortakkuratheit, die mit den unveränderten Merkmalen erreicht wird, beträgt 69.89 %. Schon durch Dekorrelierung aller 24 Merkmale mit linearer PCA (Karhunen-Loève-Transformation) kann diese Wortakkuratheit auf 71.27 % verbessert werden. Werden nur die 12 dynamischen Merkmale mit der linearen PCA transformiert, ergibt sich zusammen mit den unveränderten statischen Merkmalen eine Wortakkuratheit von 69.19 %; werden nur die 12 statischen Merkmale dekorreliert, erhält man 70.41 % WA (siehe Tabelle 5.10). Hier schneidet die lineare PCA der Ableitungen besonders schlecht ab, da die Ableitungen nicht über den optimalen Kontext aus Tabelle 5.3 berechnet werden, sondern über ein Kontextfenster von 90 ms, wie es bisher in `fex3_1` (vgl. Abschnitt 2.3) implementiert ist. In den folgenden Experimenten werden jedoch konsequent die Berechnungsverfahren aus `fex3_1` übernommen.

Wie die erste Zeile in Tabelle 5.11 zeigt, können diese Ergebnisse zunächst aber bei weitem nicht erreicht werden. Nach getrennter nichtlinearer PCA der statischen und der dynamischen Merkmale werden nur 52.55 % Wortakkuratheit erzielt. Es gilt hier $F = 90$: Die 12 ursprünglichen Merkmale werden beibehalten, zusätzlich 12 Quadrate gebildet und $\begin{pmatrix} 12 \\ 2 \end{pmatrix}$ Produkte $x_i x_j$ mit ($i \neq j$).

In gesonderten Experimenten werden nun einmal die statischen Merkmale unverändert gelassen, und die dynamischen nach nichtlinearen Hauptkomponenten entwickelt und einmal umgekehrt. Es fällt auf, dass sich die Verwendung von Produkttermen der statischen Merkmale negativ auf die Wortakkuratheit auswirkt (es werden nur 48.47 % erreicht). Mit den Produkten der dynamischen Merkmale werden jedoch relativ gute Erkennungsraten erzielt (61.58 %).

Projiziert man den $\mathbb{R}^{90}$ in verschiedene Koordinatenebenen, so kann man folgendes beobachten: Für jede Komponente x_i des ursprünglichen Merkmalvektors gilt, dass im $\mathbb{R}^{90}$ die Ebene, die von den Achsen x_i und $(x_i)^2$ aufgespannt wird, nicht zur Unterscheidung der Klassen beiträgt, da

Komponenten des $\mathbb{R}^F$ $x \in \{a, s\}$	**PCA der** a_i s_i unverändert	**PCA der** s_i a_i unverändert	**PCA der** a_i **und PCA der** s_i
unveränderte x_i, Produkte $x_i x_j (i \neq j)$ und Quadrate $x_i x_i$	61.58	48.47	52.55
unveränderte x_i und Produkte $x_i x_j (i \neq j)$ (keine Quadrate)	62.57	—	—
Translation der x_i um Δ_i: alle $x_i + \Delta_i$ und Produkte $(x_i + \Delta_i)(x_j + \Delta_j)$ $(i \neq j)$	72.01	70.48	72.76
Translation der x_i um Δ_i: Quadrate $(x_i + \Delta_i)^2$ und alle Produkte $(x_i + \Delta_i)(x_j + \Delta_j)$ $(i \neq j)$	72.55	70.76	71.62

Tabelle 5.11: Wortakkuratheit in Prozent nach *nichtlinearer* PCA. Die s_i sind die statischen Merkmale, die a_i sind die Ableitungen.

alle Merkmale dort auf einer Parabel liegen. Deshalb werden nun die 12 Quadrate weggelassen und die PCA im $\mathbb{R}^{78}$ ausgeführt. Eine geringe Verbesserung der Wortakkuratheit zeigt die zweite Zeile aus Tabelle 5.11.

Nun wird die Ursache für die Unterschiede zwischen den Ergebnissen nach Produktbildung der statischen und der dynamischen Merkmale gesucht. Es fällt auf, dass die Ableitungen zum Großteil im Intervall $[-10; 10]$ liegen, die Cepstrum-Merkmale aber in kleineren Intervallen, manche Komponenten sogar größtenteils innerhalb $[-1; 1]$. Wie Abbildung 5.14 zeigt, werden aber durch Produkte von Merkmalkomponenten, deren Betrag kleiner eins ist, getrennte Punktwolken zusammengeworfen. Daher erscheint es sinnvoll, in den nächsten Experimenten die Merkmale zunächst in den Raum $[1; \infty)^f$ zu translatieren (Abbildung 5.15). Dazu wird das Minimum jeder Merkmalkomponente aus der Trainingsstichprobe ermittelt. Die Minima der Ableitung sind im Vektor $\boldsymbol{a_{min}}$ zusammengefasst, die der statischen Merkmale in $\boldsymbol{s_{min}}$, mit

$$\begin{aligned} \boldsymbol{a_{min}} &= (-9.8, -6.9, -7.4, -6.5, -7.5, -8.1, -9.7, -8.5, -10.2, -7.9, -7.1, -7.2)^T \\ \boldsymbol{s_{min}} &= (-7.5, -4.5, -4.7, -4.9, -5.2, -5.8, -6.1, -9.4, -8.4, -8.1, -7.8, -6.7)^T. \end{aligned} \quad (5.4)$$

Das folgende Vorgehen wird nun für die statischen Merkmale und die Ableitungen separat

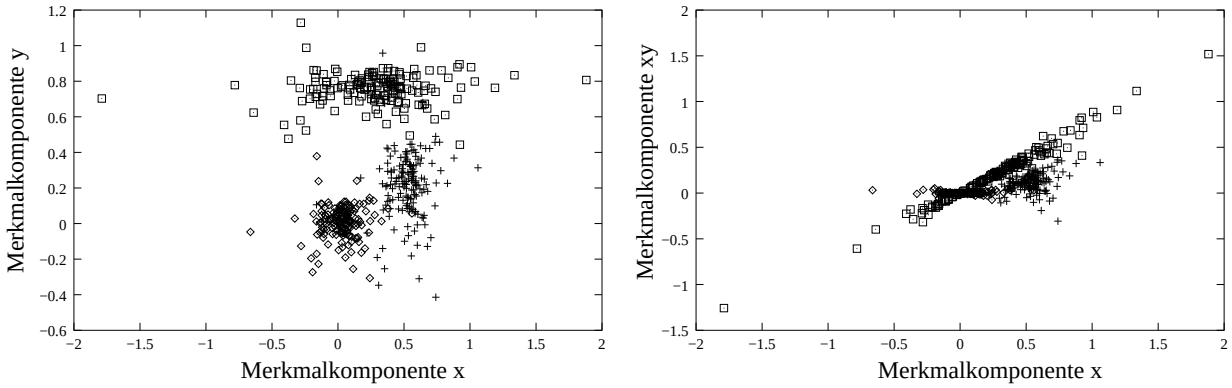

Bild 5.14: Merkmalkomponenten x und y, deren Betrag größtenteils kleiner eins ist (links); Produktterm xy (rechts)

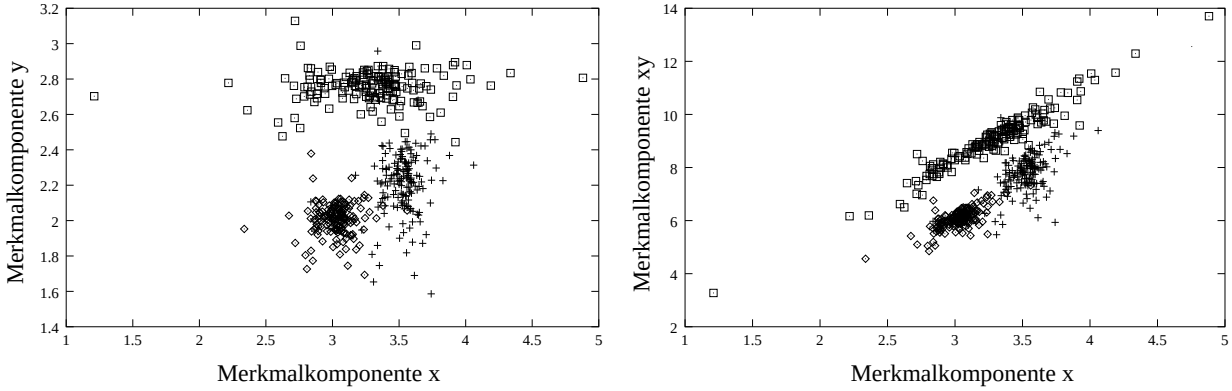

Bild 5.15: Merkmalkomponenten x und y, die größer eins sind (links); Produktterm xy (rechts)

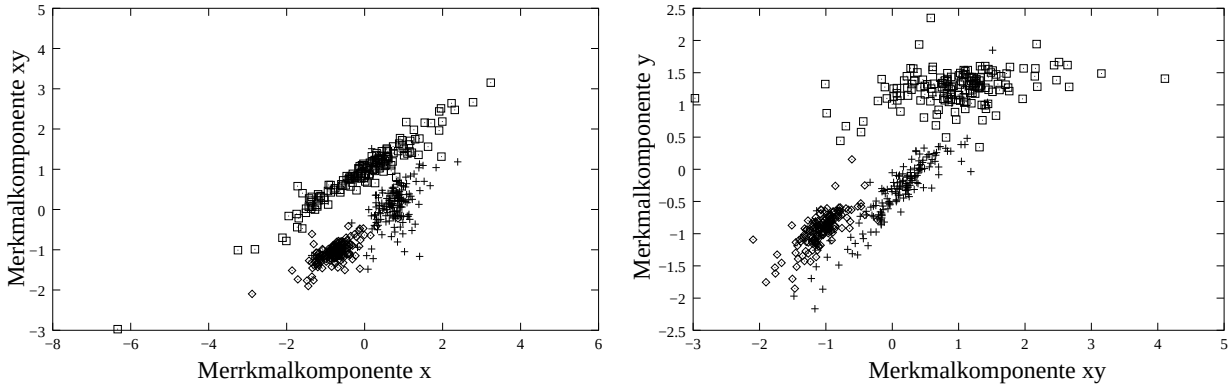

Bild 5.16: Zwei Ansichten des F-dimensionalen Raumes nach Normierung der Varianzen in jeder Richtung und nach Rücktransformation des Mittelwertes in den Ursprung

durchgeführt:

- Normierung der Merkmale im $\mathbb{R}^{12}$, so dass ihr Mittelwert Null und die Varianz in jeder Richtung eins ist.
- Translation der statischen Merkmale um $-\boldsymbol{s_{min}}+1$ sowie der Ableitungen um $-\boldsymbol{a_{min}}+1$.
- Transformation in den $\mathbb{R}^{78}$ durch Hinzunahme von $\begin{pmatrix} 12 \\ 2 \end{pmatrix}$ Produkten der ursprünglichen Merkmalkomponenten.
- Translation im $\mathbb{R}^{78}$, so dass der Mittelwert wieder Null ist.
- Normierung der Merkmale im $\mathbb{R}^{78}$, so dass die Varianz in jeder Richtung eins ist. Die resultierenden Merkmale sind in Abbildung 5.16 illustriert.
- Lineare PCA (Karhunen-Loève-Transformation) mit Korrelationsanalyse im $\mathbb{R}^{78}$. Man erhält 12 dekorrelierte Merkmale.

In den drei verschiedenen Variationen dieses Experimentes werden nun gute Ergebnisse erzielt. Tabelle 5.1, dritte Zeile, zeigt Wortakkuratheiten bis zu 72.76 %.

In weiteren Experimenten wird das Vorgehen so verändert, dass die PCA nun im $\mathbb{R}^{144}$ durchgeführt wird. Dazu werden alle möglichen $12 \cdot 12$ Produkte gebildet: Quadrate x_i^2 anstelle der ursprünglichen Merkmale und doppeltes Hinzunehmen der Produkte $x_i x_j = x_j x_i$ ($i \neq j$). Das Skalarprodukt zweier solcher transformierten Vektoren im $\mathbb{R}^{144}$ entspricht nämlich genau der Kernelfunktion

$$k(\boldsymbol{x}, \boldsymbol{y}) = (\boldsymbol{x}^T \boldsymbol{y})^2 \tag{5.5}$$

der 12-dimensionalen Vektoren $\boldsymbol{x}$ und $\boldsymbol{y}$ (vgl. Gleichung 5.3). Die Ergebnisse sind in Tabelle 5.1, letzte Zeile, zusammengefasst. Sie entsprechen etwa den vorherigen Resultaten. Nach nichtlinearer PCA der statischen Merkmale werden 70.76 % WA erreicht, bei den dynamischen 72.55 % WA. Der Erfolg nach Transformation beider Merkmalgruppen addiert sich jedoch bei dieser Variante nicht.

In einem abschließenden Experiment wird untersucht, ob mit der nichtlinearen PCA der gesamte nicht-homogene Merkmalraum aus statischen und dynamischen Merkmalen *zusammen* nach Hauptkomponenten erfolgreich analysiert werden kann. Mit 71.70 % WA wird auch hier ein besseres Ergebnis als mit linearer PCA erreicht (71.27 & WA).

Vergleicht man die Ergebnisse, die durch nichtlineare PCA erzielt werden, mit denen, die durch lineare PCA erzielt werden, so fällt eine deutliche Verbesserung in denjenigen Experimenten auf, in denen die Ableitungen transformiert werden und die statischen Merkmale konstant gelassen werden (72.55 % gegenüber 69.19 %). Bei den übrigen Experimenten ist der Unterschied zwischen linearer und nichtlinearer PCA mit Produkttermen der Ordnung 2 nur gering.

5.5 Zusammenstellung der Experimente

An dieser Stelle werden die wichtigsten Experimente aus diesem Kapitel zusammengestellt und die Erfolge aufgezeigt. Die ausgewählten Experimente sind in Tabelle 5.12 aufgelistet. Als Grundlage zum Vergleich dient ein Experiment mit den bisherigen unveränderten Merkmalen, die mit `fex3_1` erzeugt werden und Ausgangsbasis der Arbeit bildeten. Wie in Abschnitt 2.3 beschrieben, werden 12 statische und 12 dynamische Merkmale berechnet. Die Wortakkuratheit (WA) beträgt 69.89 %; nach Dekorrelierung der Merkmale mit KLT werden gar 71.27 % WA erreicht. Werden nur die statischen Merkmale dekorreliert und die Ableitungen unverändert gelassen, ergeben sich 70.41 % WA.

Nun werden Experimente durchgeführt, bei denen die statischen Merkmale unverändert bleiben und nur die dynamischen variieren. Zuerst werden die 12 Ableitungen über verschieden breite Kontextfenster berechnet und mit der KLT dekorreliert. Ein Optimum wird bei 50 ms Kontextfenstern erreicht, während in `fex3_1` hingegen 90 ms Kontext betrachtet werden. Die erzielte Wortakkuratheit beträgt 72.48 %. Als nächstes werden die Ableitungen parallel für verschiedene Zeitauflösungen berechnet. Mit der KLT werden hier 36 auf 12 Merkmale reduziert. Ein Optimum von 73.78 % WA wird für Fenster der Breite 30 ms, 50 ms und 70 ms erzielt.

Transformiert man mit der KLT statische und dynamische Merkmale gemeinsam, also hier 48 (12 statische Merkmale plus 36 Ableitungen) auf 24 Merkmale, sind die Erkennungsraten schlechter (72.68 % WA). In Experimenten mit der PPCA erhält man bis zu 73.99 % WA.

Ein anderer Weg der Kontextberücksichtigung ist die Konkatenation des aktuellen Merkmalvektors mit seinen zeitlichen Nachbarn. In einem Versuch werden statische und dynamische Merkmale der beiden Nachbarn sowie die Ableitungen des aktuellen Merkmalvektors (es ergeben sich 60 Merkmale) mit der KLT auf 12 Komponenten reduziert; die aktuellen statischen Merkmale bleiben unverändert. Die Wortakkuratheit ist dann 71.16 %.

In weiteren Experimenten werden die statischen Merkmale optimiert. Kurzzeitenergie und Mel-Cepstrum-Koeffizienten werden nun nicht nur aus 16 ms breiten Fenstern des Zeitsignals berechnet, sondern parallel dazu auch für andere Fensterbreiten. Die statischen Merkmale aus verschiedenen Zeitauflösungen werden mit der KLT auf 12 Merkmale reduziert; die Ableitungen

Merkmale	**Transformation**	**WA** in %
Bisherige Merkmale aus `fex3_1`	a_i, s_i	69.89
Bisherige Merkmale aus `fex3_1`	KLT(a_i, s_i)	71.27
Bisherige Merkmale aus `fex3_1`	a_i, KLT(s_i)	70.41
Regressionsgerade über 50 ms	KLT(a_i), s_i	72.48
Dynamische Merkmale für mehrere Zeitauflösungen		
Regression über 30 ms, 50 ms, 70 ms	KLT(a_i), s_i	73.78
Regression über 30 ms, 50 ms, 70 ms	KLT(a_i, s_i)	72.68
Regression über 30 ms, 50 ms, 70 ms	PPCA(a_i, s_i)	**73.99**
Konkatenation $a_i^{\pm 1}$, $s_i^{\pm 1}$ sind die zeitlichen Nachbarn	KLT($a_i, a_i^{\pm 1}, s_i^{\pm 1}$), s_i	71.16
Statische Merkmale für mehrere Zeitauflösungen		
Kurzzeitanalysefenster 16 ms, 32 ms	a_i, KLT(s_i)	70.18
Kurzzeitanalysefenster 4 ms, 8 ms, 16 ms	a_i, KLT(s_i)	70.13
Kurzzeitanalysefenster 5 ms, 10 ms, 16 ms, 20 ms	a_i, KLT(s_i)	71.79
KLT statt Kosinustransformation Kurzzeitanalysefenster 16 ms	a_i, KLT(s_i)	65.95
Kurzzeitanalysefenster 16 ms, 32 ms	a_i, KLT(s_i)	66.62
Keine Filterbank; KLT des logarithmierten Spektrum Kurzzeitanalysefenster 16 ms	a_i, KLT(s_i)	61.57
Kurzzeitanalysefenster 16 ms, 32 ms	a_i, KLT(s_i)	58.04
Produktterme		
Transformation in den $\mathbb{R}^{(12\cdot 12)}$	KLT(a_i), s_i	72.55
Transformation in den $\mathbb{R}^{(12\cdot 12)}$	a_i, KLT(s_i)	70.76
Transformation jeweils in den $\mathbb{R}^{78}$	KLT(a_i), KLT(s_i)	72.76
Transformation in den $\mathbb{R}^{(24\cdot 24)}$	KLT(a_i, s_i)	71.70

Tabelle 5.12: Zusammenstellung ausgewählter Experimente. Die s_i sind die statischen Merkmale, die a_i sind die Ableitungen.

bleiben unverändert. Nach Berechnung der Merkmale aus 16 ms und 32 ms breiten Analysefenstern beträgt die Wortakkuratheit 70.18 %. Auch mit kleineren Fenstern der Breite 4, 8 und 16 ms werden 70.13 % erreicht. In einem Experiment mit 5 ms, 10 ms, 16 ms und 20 ms Analysefenstern kann die Wortakkuratheit auf 71.79 % verbessert werden. Schon nach geringfügiger Veränderung dieser Kombination wird ein so gutes Ergebnis aber nicht mehr erzielt. Auch ein Versuch mit der großen Stichprobe kann eine so deutliche Verbesserung nicht bestätigen.

In einer Abänderung dieses Versuchs wird in der Mel-Cepstrum-Berechnung die Kosinustransformation durch die KLT ersetzt. Im Experiment mit verschiedenen Fensterbreiten werden nun mit Hilfe der KLT die unterschiedlichen logarithmierten Mel-Spektren gemeinsam auf 12 statische Merkmale reduziert. Die 12 Ableitungen bleiben wieder unverändert. Bei Kurzzeitanalysefenstern von 16 ms Breite werden 65.95 % WA erzielt, bei 16 ms und 32 ms breiten Fenstern 66.62 %. Die Berechnung für mehrere Zeitauflösungen bringt also Vorteile, jedoch werden insgesamt schlechtere Ergebnisse als mit der Kosinustransformation erzielt.

In anderen Untersuchungen wird kein Bandspektrum berechnet sondern das logarithmierte Spektrum selbst mit der KLT reduziert. Bei 16 ms breiten Kurzzeitanalysefenstern beträgt die Wortakkuratheit 61.57 %, bei Fenstern der Breiten 16 ms und 32 ms nur 58.04 %.

Die Idee der Produktterme in den letzten Experimenten ist es, durch Hinzunahme von Produkten der ursprünglichen f Merkmalkomponenten die Merkmalvektoren verschiedener Klassen im Raum besser trennen zu können. Die hochdimensionalen Vektoren (Dimension $F > f$) werden dann wieder mit KLT reduziert. Diesen Ansatz nennt man auch nichtlineare PCA. In unterschiedlichen Untersuchungen gilt $F = f \cdot f$ bzw. $F = \begin{pmatrix} f \\ 2 \end{pmatrix} + f$. Durch Produktbildung der Ableitungen und anschließender KLT werden bei konstant gehaltenen statischen Merkmalen bis zu 72.55 % WA erreicht, durch Produkte der statischen Merkmale bei konstant gehaltenen dynamischen bis zu 70.76 % WA. Nach separater Transformation der beiden Merkmalgruppen ist die Erkennungsrate 72.76 % WA, nach nichtlinearer PCA beider Gruppen gemeinsam 71.70 % WA.

Zusammenfassend lässt sich sagen, dass die erfolgreichste Vorgehensweise in dieser Arbeit diejenige ist, in der die Ableitungen über verschiedene Zeitauflösungen berechnet werden. So kann die ursprüngliche Erkennungsrate von 69.89 % WA auf bis zu 73.99 % WA verbessert werden. Dies entspricht einer Verringerung der Fehlerrate um 13.6 %. Auch auf der großen Stichprobe kann die Fehlerrate um 9.4 % vermindert werden. Ein zusätzliches Experiment mit der PPCA auf der großen Stichprobe könnte diese Verbesserung bestimmt noch erhöhen.

Im nächsten Kapitel werden weitere mögliche Experimente zur Verbesserung der Merkmalberechnung in einem Ausblick dargestellt. Eine Zusammenfassung dieser Arbeit folgt danach.

Kapitel 6

Ausblick

In diesem Kapitel wird ein kurzer Überblick darüber gegeben, welche weiteren Untersuchungen unternommen werden können, um bessere Merkmale für die automatische Spracherkennung zu finden. Zahlreiche andere Experimente aus dem weiten Themenbereich “Optimierung der Merkmalberechnung” sind denkbar; hier folgt eine Auswahl solcher Untersuchungen, die inhaltlich direkt an diese vorliegende Studienarbeit anknüpfen.

Zweierlei Ansätze, die Merkmalberechnung zu verbessern, werden in dieser Arbeit verfolgt. Einmal werden Merkmale parallel für verschiedene Zeitauflösungen berechnet und der entstehende hochdimensionale Merkmalvektor mit der Karhunen-Loève-Transformation (KLT) wieder auf die Merkmaldimension 24, die bisher in der automatischen Spracherkennung am LME verwendet wird, reduziert. Der andere Ansatz ist, Produkte der bisherigen 24 Merkmalkomponenten zu berechnen, wiederum einen hochdimensionalen Merkmalvektor zu erhalten und auch diesen mit der KLT zu reduzieren.

Bei der Berechnung der Merkmale in verschiedenen Zeitauflösungen werden die Experimente für die statischen und die dynamischen Merkmale getrennt durchgeführt. Als dynamische Merkmale werden Ableitungen erster Ordnung verwendet. Interessant wäre es nun zu untersuchen, ob weitere Verbesserungen erzielt werden können, wenn man zusätzlich Ableitungen zweiter Ordnung mit hinzuzieht. Durch Experimente muss herausgefunden werden, für welche Zeitauflösungen die zweiten Ableitungen optimale Ergebnisse liefern und ob sie gemeinsam mit den ersten Ableitungen mit Hilfe der KLT transformiert werden sollen, oder getrennt. Im Falle einer getrennten Behandlung, müsste man untersuchen, wie viele dynamische Merkmale man für die KL-transformierten Ableitungen erster Ordnung reservieren muss und wieviele für die Ableitungen zweiter Ordnung.

Nachdem die Berechnung der statischen Merkmale über verschiedene Zeitauflösungen keine

Erfolge gebracht hat, wird in dieser Arbeit eine Optimierung der Mel-Filterbank gefordert. Man hofft, durch Verwendung einer Filterbank mit mehr Filtern die Feinheiten, die durch verschiedene Zeitauflösungen entstehen, im Spektrum robuster ausfiltern zu können. Vermutlich wird man allein mit einer anderen Mel-Filterbank bessere Merkmale erhalten und darauf aufbauend evtl. zusätzliche Verbesserungen durch den "Multi-Resolution"- Ansatz erzielen.

Auch beim Ansatz, die Merkmale auf nichtlineare Hauptkomponenten hin zu untersuchen, indem man Produktterme der Merkmale bildet, ist weitere Forschungsarbeit notwendig: Durch Produkte der Ordnung drei und höher sind Verbesserungen zu erwarten. Ferner muss nach einer optimalen Anzahl von Merkmalen nach der Karhunen-Loève-Transformation gesucht werden.

In dieser Arbeit wurden Produkte der Komponenten i und j der Merkmalvektoren $\boldsymbol{c_\tau}$ eines Kurzzeitanalysefensters τ berechnet, also $c_{i,\tau} \cdot c_{j,\tau}$. Interessant wäre es nun, Produkte aus benachbarten Komponenten zeitlich benachbarter Vektoren zu bilden, also beispielsweise $c_{i-1,\tau-1} \cdot c_{i,\tau}$ oder $c_{i+1,\tau-1} \cdot c_{i,\tau}$. Da nämlich durch das Cepstrum eine Quefrenzanalyse des Spektrums erfolgt, würde durch solche Produkte berücksichtigt, dass sich Merkmale zwischen zeitlich nahen Analysefenstern in benachbarte Quefrenzbereiche verschieben können.

Die Untersuchungen, Experimente für mehrere Zeitauflösungen parallel durchzuführen, können auch auf den neuen Ansatz der TRAPs ([Her98], siehe Abschnitt 2.5) ausgedehnt werden. In weiteren Experimenten kann untersucht werden, welche zusätzliche Verbesserung der Erkennungsraten durch das Ersetzen der KLT durch die Linearen Diskriminanzanalyse erreicht werden kann. Auch der Einsatz der PPCA, mit der das optimale Ergebnis in dieser Arbeit erzielt wurde, sollte noch auf andere erfolgreichen Untersuchungen ausgedehnt werden.

Auch die bisher verwendete Merkmaldimension 24 sollte in weiteren Versuchen variiert werden. In dieser Arbeit wird sie zum Zwecke der Vergleichbarkeit der Ergebnisse konstant gehalten. Bei einer Optimierung müssten aber folgende Aspekte berücksichtigt werden:

- Man muss abwägen, wieviel Verbesserung es bringt, die Dimension zu erhöhen, und wieviel aufwendiger dadurch die nachfolgenden Berechnungen bei der Klassifikation werden. Insbesondere muss eine obere Schranke gesetzt werden, um in Echtzeit klassifizieren zu können.

- Die optimale und vertretbare Merkmaldimension wird für verschiedene Methoden der Merkmalberechnung unterschiedlich sein.

- Es muss untersucht werden, welcher Anteil der Merkmalkomponenten für statische Merkmale reserviert werden soll und welcher für dynamische Merkmale. Gegebenfalls muss

eine weitere Unterteilung für Ableitungen erster und zweiter Ordnung unternommen werden. Die Anzahl der statischen und der dynamischen Merkmale muss nämlich nicht mehr identisch sein, wenn im Berechnungsablauf zusätzlich die KL-Transformation eingebaut ist.

Die Merkmalberechnung kann also wohl durch viel Forschungsarbeit noch weiter verbessert werden; im nachfolgenden Kapitel sind die Untersuchungen aus dieser Arbeit abschließend zusammengefasst.

Kapitel 7

Zusammenfassung

In der vorliegenden Studienarbeit wird die Merkmalberechnung optimiert. Dabei werden zunächst höherdimensionale Merkmalvektoren dadurch gebildet, dass man die Merkmale für verschiedene Zeitauflösungen berechnet oder Produkte der bisherigen Merkmalkomponenten hinzunimmt. Diese Vektoren werden dann wieder auf die ursprüngliche Dimension reduziert, mit dem Ziel, dann bessere Merkmale zu erhalten.

Zunächst werden aus der Literatur bekannte Verfahren zur Merkmalberechnung vorgestellt. Bei der Kurzzeitanalyse wird das Signal durch Multiplikation mit einer Fensterfunktion in kleine Zeitabschnitte zerlegt. Aus diesen Signalfenstern werden Zeitbereichsmerkmale, wie z.B. die Kurzzeitenergie, sowie spektrale Merkmale berechnet. Durch Filterung mit verschiedenen Bandpass-Filtern, die auf der Mel-Skala äquidistant liegen, erhält man die Mel-Spektrum-Koeffizienten. Durch das Mel-Spektrum wird die menschliche Tonhöhenempfindung modelliert, durch zusätzliches Logarithmieren auch die Lautheitsempfindung des Menschen. Das Frequenzverhalten des Spektrums (man spricht hier von der Quefrenz) wird durch das Cepstrum beschrieben. Durch Anwenden der Kosinustransformation auf das Mel-Spektrum erhält man das Mel-Cepstrum. Die Mel-Spektrum-Koeffizienten werden dadurch dekorreliert. Alternativ zum Cepstralanalyse wird unter anderem auch die lineare Vorhersage verwendet.

Alle bisher beschriebenen Merkmale geben nur Informationen über das jeweilige Kurzzeitanalysefenster. Man spricht von statischen Merkmalen. Dynamische Merkmale beschreiben das Verhalten der statischen Merkmale über größere Zeitbereiche. Man erhält sie durch Ableiten jener Merkmale oder durch das 2D-Cepstrum.

Am LME werden die Merkmale mit dem Programm `fex3_1` erzeugt. Aus dem Sprachsignal werden alle 10 ms überlappende Kurzzeitanalysefenster der Breite 16 ms betrachtet. Daraus werden 12 statische Merkmale gebildet: die Energie und elf Mel-Cepstrum-Koeffizienten. Zusammen mit den zwölf Ableitungen dieser Merkmale erhält man 24-dimensionale Merkmalvektoren.

Die Ableitungen werden durch Regressionsgeraden über 90 ms breiten Fenstern approximiert.

Untersuchungen zur Verbesserung der Merkmalberechnung wurden am LME von S. Rieck [Rie94] und V. Fischer [Fis88] durchgeführt. Eine Auswahl anderer Veröffentlichungen gibt Abschnitt 2.5.

In den Experimenten werden F-dimensionale Merkmalvektoren berechnet und danach auf $f < F$ Dimensionen reduziert. Dazu wird eine problemabhängige Reihenentwicklung, die Karhunen-Loève-Transformation (KLT), benutzt. Dabei werden die f Hauptstreuungsrichtungen einer mittelwertfreien Stichprobe von Merkmalvektoren berechnet. Diese entsprechen denjenigen Eigenvektoren der Korrelationsmatrix, welche die f größten Eigenwerte haben. Durch Orthogonalprojektion der hochdimensionalen Vektoren in den Unterraum, der von jenen f Eigenvektoren aufgespannt wird, bleibt der mittlere quadratische Fehler minimal. Für die Korrelationsanalyse hat man sich entschieden, da so bessere Ergebnisse als bei der Varianz-/Kovarianzanalyse erzielt werden. Zur Berechnung der Eigenwerte und -vektoren wird die Matrix mit Householder-Matrizen auf Tridiagonalform gebracht. Danach werden die Eigenvektoren mit dem QL-Algorithmus berechnet.

Einige Experimente werden zusätzlich mit der PPCA (Probabilistic Principal Component Analysis) durchgeführt. Dabei wird der Erkenner direkt mit den F-dimensionalen Vektoren trainiert. Es wird jedoch eine spezielle Ausgabe-Dichte für das Sprachmodell verwendet, die mit diesen hochdimensionalen stark korrelierten Vektoren umgehen kann.

Die durchgeführten Experimente sind folgendermaßen aufgebaut: Zunächst werden aus einer Stichprobe Merkmale in unterschiedlicher Weise berechnet. Als Stichprobe wird die EVAR-Stichprobe herangezogen, aus der 7438 Äußerungen zufällig ausgewählt wurden. Alle Sprachsignale wurden auf Telefonqualität gefiltert. Die Trainingsstichprobe umfasst 4999 Äußerungen, die Validierungsstichprobe 441 und die Teststichprobe 1998. Ausgewählte besonders erfolgreiche Untersuchungen werden zusätzlich auf einer größeren Stichprobe von 20678 Äußerungen durchgeführt. Der Erkenner wird mit Hilfe des Systems ISADORA trainiert und anschließend getestet. Dazu wird der LRBEAM-Erkenner verwendet. Verglichen werden nun die erzielten Erkennungsraten, gemessen in Wortakkuratheit.

Als Grundlage zum Vergleich dient ein Experiment mit den bisherigen unveränderten Merkmalen, die mit `fex3_1` erzeugt werden. Hier wird eine Wortakkuratheit (WA) von 69.89 % erzielt. Nach Dekorrelierung der Merkmale mit KLT werden gar 71.27 % WA erreicht. Es werden nun drei Blöcke von Experimenten durchgeführt: Einmal werden die Ableitungen über mehrere Zeitauflösungen berechnet und die statischen Merkmale unverändert gelassen. Danach werden

umgekehrt die statischen Merkmale für verschiedene Zeitfenster berechnet. Im dritten Block werden Produkte der ursprünglichen Merkmalkomponenten gebildet.

Die besten Ergebnisse werden im ersten Block erzielt. Zunächst wird untersucht, wie breite Kontextfenster zur Berechnung der Regression der statischen Merkmale optimale Erkennungsraten liefern. Die Ableitungen werden danach mit der KLT dekorreliert. Es zeigt sich, dass optimale Erkennungsraten für 50 ms Kontextfenster erzielt werden und nicht, wie in `fex3_1`, für 90 ms. Werden die Ableitungen parallel aus verschieden breiten Kontextfenstern berechnet und anschließend diese hochdimensionalen Vektoren wieder auf 12 Komponenten reduziert, so kann die Wortakkuratheit weiter verbessert werden. Ein Optimum wird für Fenster der Breite 30 ms, 50 ms und 70 ms erzielt. Auf diese Weise wird die optimale Auflösung durch zusätzliche Informationen ergänzt. In verschiedenen Untersuchungen werden bis zu 73.99 % WA erreicht. Dies entspricht einer Verringerung der ursprünglichen Fehlerrate um 13.6 %. Ein Experiment mit der großen Stichprobe bestätigt den Erfolg.

Ein anderer Weg der Kontextberücksichtigung ist die Konkatenation des aktuellen Merkmalvektors mit seinen zeitlichen Nachbarn. In einem Experiment wird eine Wortakkuratheit von 71.16 % erzielt. Dort werden statische und dynamische Merkmale der beiden Nachbarn sowie die Ableitungen des aktuellen Merkmalvektors mit der KLT auf 12 dynamische Merkmale reduziert.

In weiteren Experimenten werden die statischen Merkmale optimiert. Kurzzeitenergie und Mel-Cepstrum-Koeffizienten werden nun nicht nur aus 16 ms breiten Fenstern des Zeitsignals berechnet, sondern parallel dazu auch für andere Fensterbreiten. Durch größere Fenster wird die Zeitauflösung verschlechtert und nach dem Unschärfeprinzip die Frequenzauflösung verbessert, bei kleineren Fenstern umgekehrt. In Experimenten werden also die aus verschiedenen Fenstern gewonnenen statischen Merkmale mit der KLT auf 12 Merkmale reduziert. Nur in einem Experiment mit Fenstern der Breite 5 ms, 10 ms, 16 ms und 20 ms kann die Wortakkuratheit auf 71.79 % verbessert werden. Schon nach geringfügiger Veränderung dieser Kombination wird ein so gutes Ergebnis nicht mehr erzielt. Auch ein Versuch mit der großen Stichprobe kann eine so deutliche Verbesserung nicht bestätigen. Die Ergebnisse dieses Ansatzes werden dadurch begründet, dass mit der unveränderten Mel-Filterbank die Vorteile, die durch bessere Frequenzauflösung erzielt werden, wohl wieder eliminiert werden.

In einer Abänderung dieses Versuchs wird in der Mel-Cepstrum-Berechnung die Kosinustransformation durch die KLT ersetzt. Im Experiment werden nun die logarithmierten Mel-Spektren aus verschiedenen Analysefenstern mit der KLT auf 12 statische Merkmale reduziert. Es wird gezeigt, dass die Berechnungen für mehrere Zeitauflösungen Vorteile bringen, jedoch werden insgesamt schlechtere Ergebnisse erzielt. Dies liegt wohl daran, dass die der Cepstrum-

berechnung folgende zeitliche Energieglättung und die dynamisch adaptive cepstrale Subtraktion (DACS) weggelassen wurden, da sie in der bisherigen Form nicht auf das durch die KLT entstandene Cepstrum anwendbar sind. Insbesondere gibt es kein reines Energie-Merkmal mehr.

Da die oben geforderte Filterbankoptimierung über den Rahmen dieser Arbeit hinausgehen würde, wird stattdessen das logarithmierte Spektrum selbst mit der KLT reduziert. In einem Experiment mit 16 ms breiten Kurzzeitanalysefenstern werden so immerhin 61.57 % WA erreicht. Der Ansatz mit mehreren Zeitauflösungen schneidet schlechter ab. Begründet wird dies damit, dass die KLT in diesem Fall aus bis zu 384 Koeffizienten nicht mehr robust genug die für die Spracherkennung wichtige Information herausfiltern kann.

Im dritten und letzten Block werden Versuche mit nichtlinearer PCA (Principal Component Analysis) durchgeführt. Die Idee ist, durch Hinzunahme von Produkten der ursprünglichen Merkmalkomponenten die Merkmalvektoren verschiedener Klassen im Raum besser trennen zu können. Die hochdimensionalen Vektoren werden dann wieder mit KLT reduziert. Während Gebiete konstanter Projektion in den Unterraum bei der linearen PCA (= KLT) stets durch Hyperebenen getrennt sind, werden so auch gekrümmte Trennkurven möglich. In verschiedenen Experimenten ergeben sich so Wortakkuratheiten von bis zu 72.76 % WA. Durch Produktbildung der Ableitungen und anschließender KLT werden bei konstant gehaltenen statischen Merkmalen 72.55 % WA erreicht, in einem Vergleichsexperiment mit linearer PCA dagegen nur 69.19 % WA. In anderen Experimenten ist allerdings der Unterschied zwischen linearer und nichtlinearer PCA nur unbedeutend.

Weitere Experimente zur Merkmalberechnung sind in großer Anzahl möglich: Durch Produkte der Ordnung drei und höher sind Verbesserungen zu erwarten. Auch sollte die Mel-Filterbank optimiert werden. Weitere Vorteile verspricht man sich davon, zusätzlich die zweiten Ableitungen für verschiedene Zeitauflösungen zu berücksichtigen. Experimente mit der LDA statt der KLT könnten weitere Verbesserungen bringen. Zu guter letzt kann die Gesamtzahl der verwendeten Merkmale für die verbesserte Merkmalberechnung optimiert werden.

S. Rieck und V. Fischer haben am LME durch verschiedene Ansätze versucht, die Merkmalberechnung zu optimieren. Dank schnellerer Rechner ist es heutzutage leichter, diese vielseitigen Optimierungsansätze zu Ende zu führen. Die Verringerung der Fehlerrate um 13.6 % in dieser Studienarbeit sollte dazu motivieren.

Anhang A

Verhalten der Eigenwertberechnung für große Matrizen

In Abschnitt 3.2.2 wurde die Implementierung des Programme `pca_train` zur Berechnung der Eigenwerte und Eigenvektoren von Matrizen erläutert. Im folgenden ist das numerische Verhalten, also die Fehler s_1, s_2, m_1 und m_2 aus den Gleichungen 3.16 bis 3.19 für $n \times n$-dimensionale Matrizen ($n \in \{12, 24, 48, 100, 200, 400, 600, 800, 1000, 1200, 1500\}$, illustriert.

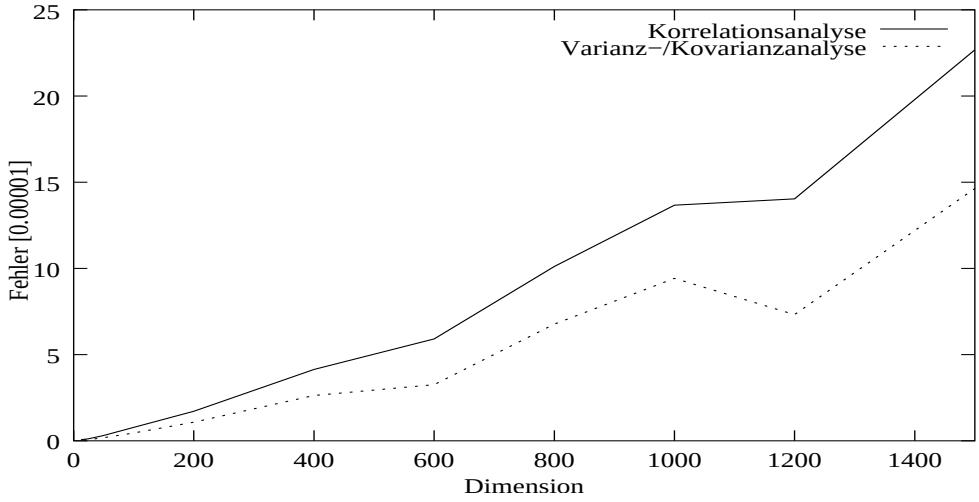

Bild A.1: Der Fehler $s_1 \cdot 10^5$ für verschiedene Dimensionen

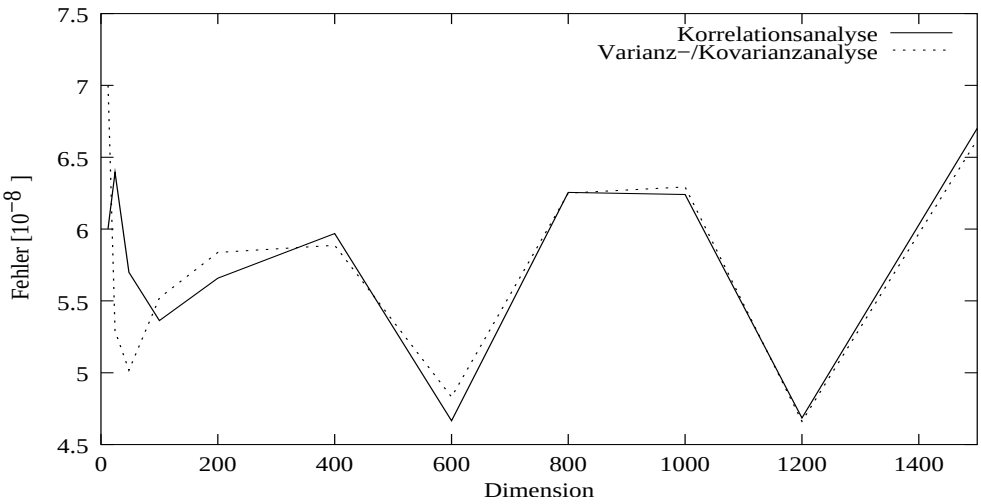

Bild A.2: Der Fehler $s_2 \cdot 10^5$ für verschiedene Dimensionen

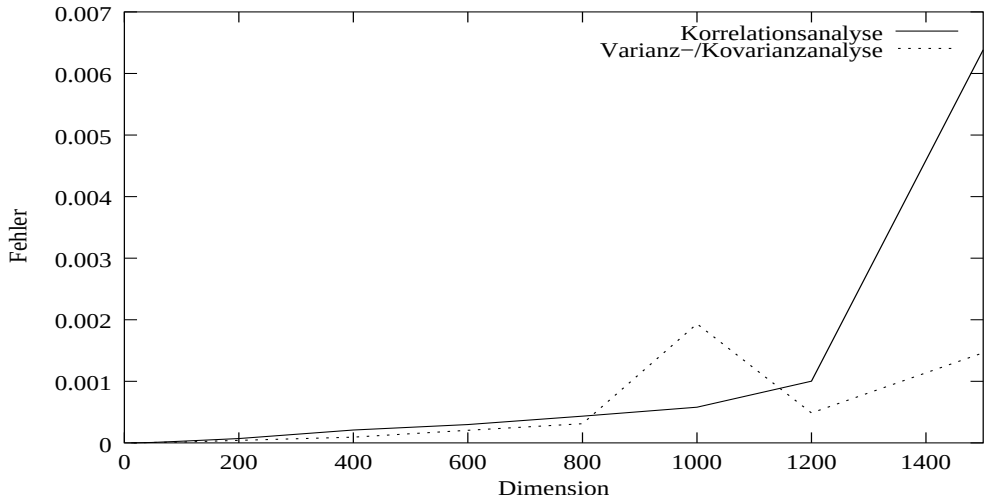

Bild A.3: Der Fehler m_1 für verschiedene Dimensionen

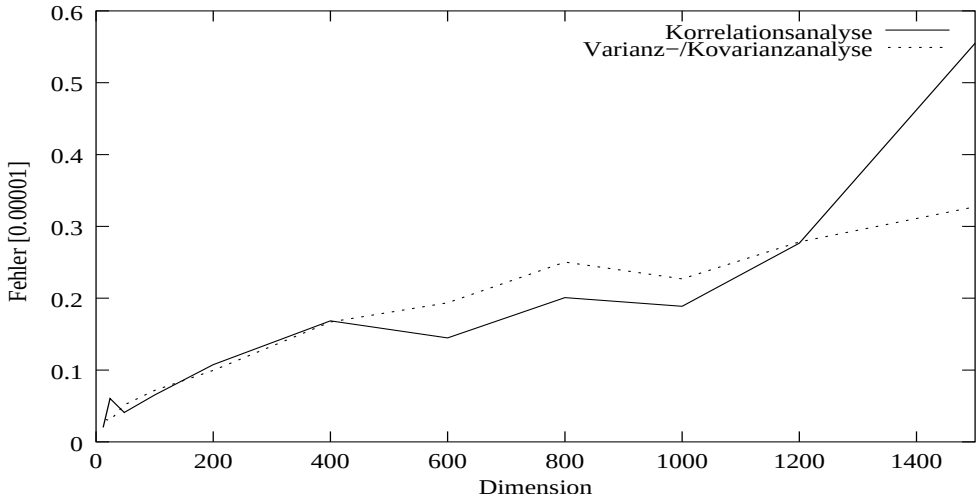

Bild A.4: Der Fehler $m_2 \cdot 10^5$ für verschiedene Dimensionen

Verzeichnis der Bilder

Verzeichnis der Tabellen

Literaturverzeichnis

[Bat98] Batlle, E.; Nadeu, C.; Fonollosa, J. A. R.: *Feature Decorrelation Methods in Speech Recognition. A Comparative Study*, in *Proc. Int. Conf. on Spoken Language Processing (ICSLP)*, Bd. 3, Sydney, Australia, 1998, S. 951 – 954.

[Bos97] Bosch, K.: *Elementare Einführung in die angewandte Statistik*, Vieweg, Braunschweig, 1997.

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

[Bur01] Burget, L.; Hermansky, H.: *Data Driven Design of Filter Bank for Speech Recognition*, in Matoušek et al. [Mat01], S. 299 – 304.

[Fis88] Fischer, V.: *Variable Länge und Fortschaltzeit der Analysefenster für die automatische Spracherkennung*, Studienarbeit, Lehrstuhl für Mustererkennung (Informatik 5), Universität Erlangen–Nürnberg, 1988.

[Gal98] Gallwitz, F.; Aretoulaki, M.; Boros, M.; Haas, J.; Harbeck, S.; Huber, R.; Niemann, H.; Nöth, E.: *The Erlangen Spoken Dialogue System EVAR: A State–of–the–Art Information Retrieval System*, in *Proceedings of 1998 International Symposium on Spoken Dialogue (ISSD 98)*, Sydney, Australia, 1998, S. 19–26.

[Her98] Hermansky, H.; Sharma, S.: *TRAPS – Classifiers of Temporal Patterns*, in *Proc. Int. Conf. on Spoken Language Processing (ICSLP)*, Sydney, Australia, 1998.

[Lev66] Levenshtein, V.: *Binary Codes Capable of Correcting Deletions Insertions and Reversals, Cybernetics and Control Theory*, Bd. 10, 1966, S. 707–710.

[Mat01] Matoušek, V.; Mautner, P.; Mouček, R.; Taušer, K. (Hrsg.): *Proc. 4th International Conference on Text, Speech and Dialogue (TSD 2001)*, Bd. 2166 von *Lecture Notes for Artificial Intelligence*, Springer–Verlag, Berlin, September 2001.

[Nie83] Niemann, H.: *Klassifikation von Mustern*, Springer–Verlag, Berlin, 1983.

[Nie90] Niemann, H.: *Pattern Analysis and Understanding*, Bd. 4 von *Springer Series in Information Sciences*, Springer, Heidelberg, 1990.

[Nöt01] Nöth, E.; Boros, M.; Fischer, J.; Gallwitz, F.; Haas, J.; Huber, R.; Niemann, H.; Stemmer, G.; Warnke, V.: *Research Issues for the Next Generation Spoken Dialogue Systems Revisited*, in Matoušek et al. [Mat01], S. 341 – 348.

[Opp83] Oppenheim, A.; Willsky, A.; Young, I. T.: *Signals and Systems*, Prentice–Hall, Englewood Cliffs, NJ, 1983.

[Pre92] Press, W.; Teukolsky, S.; Vetterling, W.; Flannery, B.: *Numerical Recipes in C*, Cambridge University Press, Cambridge, 1992.

[Psu01a] Psutka, J.; Müller, L.; Psutka, J. V.: *Comparison of MFCC and PLP Parameterizations in the Speaker Indpendent Continuous Speech Recognition Task*, in *Proc. European Conf. on Speech Communication and Technology*, Aalborg, 2001, S. 1813 – 1816.

[Psu01b] Psutka, J.; Müller, L.; Psutka, J. V.: *The Influence of a Filter Shape in Telephone–Based Recognition Module Using PLP Parameterization*, in Matoušek et al. [Mat01], S. 222 – 228.

[Rie94] Rieck, S.: *Parametrisierung und Klassifikation gesprochener Sprache*, Dissertation, Technische Fakultät der Universität Erlangen–Nürnberg, 1994.

[Sch77] Schürmann, J.: *Polynomklassifikatoren für die Zeichenerkennung*, Oldenbourg, München, 1977.

[Sch98] Schölkopf, B.; Smola, A.; Müller, K.-R.: *Nonlinear Component Analysis as a Kernel Eigenvalue Problem*, *Neural Computation*, Bd. 10, 1998, S. 1299–1319.

[ST95] Schukat-Talamazzini, E. G.: *Automatische Spracherkennung – Grundlagen, statistische Modelle und effiziente Algorithmen*, Vieweg, Braunschweig, 1995.

[Ste01] Stemmer, G.; Hacker, C.; Nöth, E.; Niemann, H.: *Multiple Time Resolutions for Derivatives of Mel-Frequency Cepstral Coefficients*, in *Proceedings of the Automatic Speech Recognition and Understanding Workshop (ASRU'01)*, 2001.

[Tip99] Tipping, M. E.; Bishop, C. M.: *Mixtures of Probabilistic Principal Component Analyzers*, *Neural Computation*, Bd. 11, Nr. 2, 1999, S. 443–482.