Grundfrequenzverfolgung und deren Anwendung in der Spracherkennung

Diplomarbeit

von Kjell Schubert

Institut für Logik, Komplexität und Deduktionssysteme
Prof. Dr. Alex Waibel
Universität Karlsruhe – SS 1999

Erklärung

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.

Karlsruhe, den 30.9.1999

Zusammenfassung

In dieser Arbeit wird ein Verfahren zur Verfolgung des Grundfrequenzverlaufes in Sprachsignalen vorgestellt. Der Algorithmus sucht den gemäß einer gegebenen Kostenfunktion optimalen Grundfrequenzverlauf in einem Cepstrogramm beziehungsweise Korrelogramm. Das Verfahren wurde zur Steigerung der Verarbeitungsgeschwindigkeit in eine grobauflösende Suche und eine sich daran anschließende Feinsuche zerlegt. Es erzielt auf einer deutschen Sprachdatenbasis eine Grobfehlerrate von 2%, das sind 50% weniger Fehler als das bisher beste uns bekannte Verfahren liefert.
Die berechneten Grundfrequenzkonturen werden zur Sprechernormierung und als zusätzliche Merkmale in einem Worterkenner für die chinesische Sprache benutzt. Für die Sprechernormierung mit Hilfe der F0-VTLN wird ein Maximum-Likelihood Ansatz zur Bestimmung der Normierungsparameter vorgestellt. Durch Einsatz der F0-VTLN konnte die Wortfehlerrate eines chinesischen und eines deutschen Worterkenners um 10% beziehungsweise 7% reduziert werden. Mittels des bisher benutzten Verfahrens zur Sprechernormierung - der ML-VTLN - werden ähnlich gute Resultate erzielt, die F0-VTLN ist jedoch um ein Vielfaches schneller als die ML-VTLN. Durch die Integration von Grundfrequenzinformationen in den Merkmalsvektor des chinesischen Erkenners konnte dessen Wortfehlerrate um 4% reduziert werden.

1 Einleitung

Die Grundfrequenz ist ein wichtiger Parameter der Sprachproduktion. Durch Modulation seiner Sprachgrundfrequenz kann ein Sprecher Grenzen zwischen Phrasen deutlich machen, Satzelemente betonen und den Satzmodus (zum Beispiel Frage oder Aussage) festlegen. In tonalen Sprachen werden Grundfrequenzverläufe zusätzlich zur Festlegung der Bedeutung einzelner phonetisch gleicher Silben verwendet. Außerdem läßt die Grundfrequenz eines Sprechers Rückschlüsse auf sein Geschlecht zu und kann somit zur Sprechernormierung verwendet werden.
Für die Aufgabe der Bestimmung von Grundfrequenzverläufen aus Sprachsignalen wurde bereits eine Vielzahl von Verfahren entwickelt. Probleme dieser Verfahren sind meistens mangelnde Robustheit und Genauigkeit. Ziel dieser Arbeit ist es, ein robustes und schnelles Verfahren zur Verfolgung des Grundfrequenzverlaufs zu entwickeln.
Mit der dadurch verfügbaren Grundfrequenzinformation soll das am Institut eingesetzte Spracherkennungssystem JANUS verbessert werden. Dieses System wurde bereits in verschiedenen Applikationen eingesetzt, enthielt aber bisher noch keine Methode zur Verfolgung von Grundfrequenzverläufen in Sprachsignalen. Als Anwendungen wurden die Sprechernormierung mittels F0-VTLN und die Integration von Grundfrequenzmerkmalen in einen Spracherkenner für die chinesische Sprache untersucht.

2 Sprachproduktion

In diesem Abschnitt wird kurz auf den Prozeß der menschlichen Sprachproduktion eingegangen, um in diesem Zusammenhang darzustellen, was unter Sprachgrundfrequenz verstanden wird.

2.1 Artikulation

Der Sprechvorgang ist eine Kombination aus der Stimmgebung, einer durch Ausatmen verursachten Schallanregung an der Stimmritze (Glottis), und der Resonanzbildung im Vokaltrakt, zu dem Mundraum, Rachenraum und Nasenraum gehören. Der Vokaltrakt wird an einem Ende durch die Stimmbänder, und am anderen durch die Lippen und Nase begrenzt. Seine Form wird durch die Positionen von Zunge, Lippen, Rachen und Velum bestimmt.

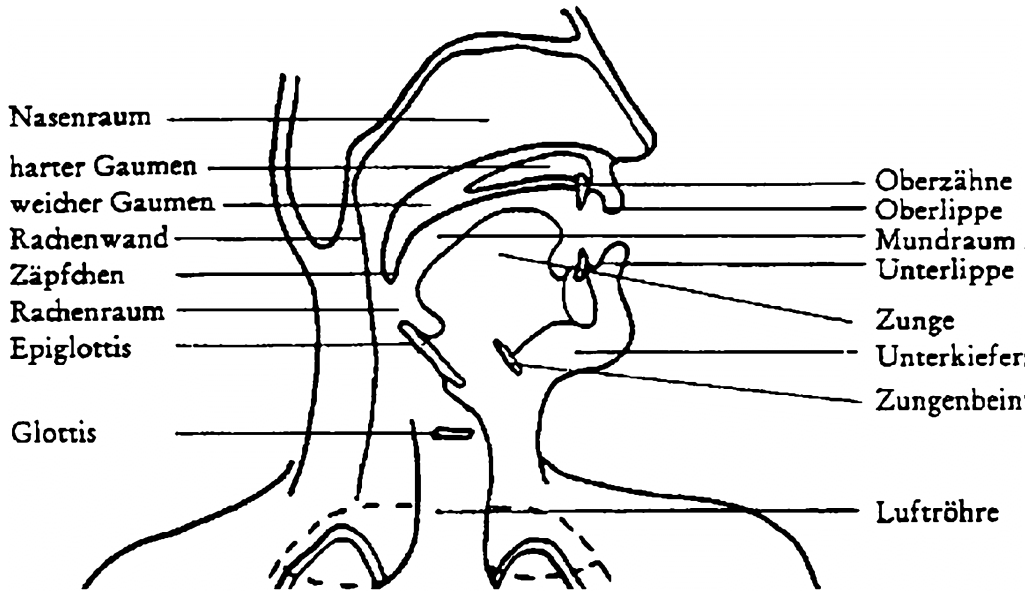

Abbildung 2.1 Das Artikulationssystem des Menschen [Koh77]

Beim Sprechprozeß wird ein Luftstrom aus den Lungen durch die Luftröhre gepreßt und gelangt durch die Stimmritze in den Vokaltrakt. Je nach Öffnungsgrad der Stimmritze gibt es zwei grundlegende Phonationsarten: stimmhaft und stimmlos. Steht die Stimmritze weit offen, bewirkt die vorbeiströmende Luft Verwirbelungen an den Stimmbändern oder an Verengungsstellen im Vokaltrakt und erzeugt so stimmlose Laute, wie zum Beispiel [s]. Im anderen Fall ist die Stimmritze verengt, und es kommt zu einem annähernd periodischen Öffnen und Schließen der Stimmbänder. Aus diesen quasiperiodischen Luftstößen gehen stimmhafte Laute wie [a], [n] oder [z] hervor.

Der erzeugte Anregungsschall wird beim Durchlaufen des Vokaltraktes zu einer Vielzahl verschiedener Laute geformt. Dabei wird der zu erzeugende Sprachlaut durch die Form des Vokaltraktes bestimmt, das heißt durch die Stellung von Velum, Zunge und Lippen, sowie durch Verengungen oder Verschlüsse an verschiedenen Positionen des Vokaltraktes. Zischlaute (Frikative) wie zum Beispiel [s], [sch] oder [f] werden durch Verengung des Vokaltraktes gebildet. Verschlußlaute (Plosive) wie etwa [t],[k] oder [b] entstehen durch einen vollständigen Verschluß im Vokaltrakt, dem Aufbau eines erhöhten Drucks hinter der Verschlußstelle, und der plötzlichen Freigabe des Luftstroms. Je nach dem zeitlichen Versatz zwischen der plötzlichen Öffnung der Verengung und dem Einsatz der Stimmbandschwingung werden stimmhafte und stimmlose Plosive unterschieden, beispielsweise [g] und [k].

2.2 Die Grundfrequenz in der Sprachproduktion

Abbildung 2.2 zeigt einige beispielhafte Ausschnitte aus Sprachsignalen. Im linken Bildteil sind stimmhafte Laute abgebildet: ein [] und ein [z]. Das rechte Bild zeigt die entsprechenden stimmlos ausgesprochenen Laute: ein geflüstertes [a], und ein [s].

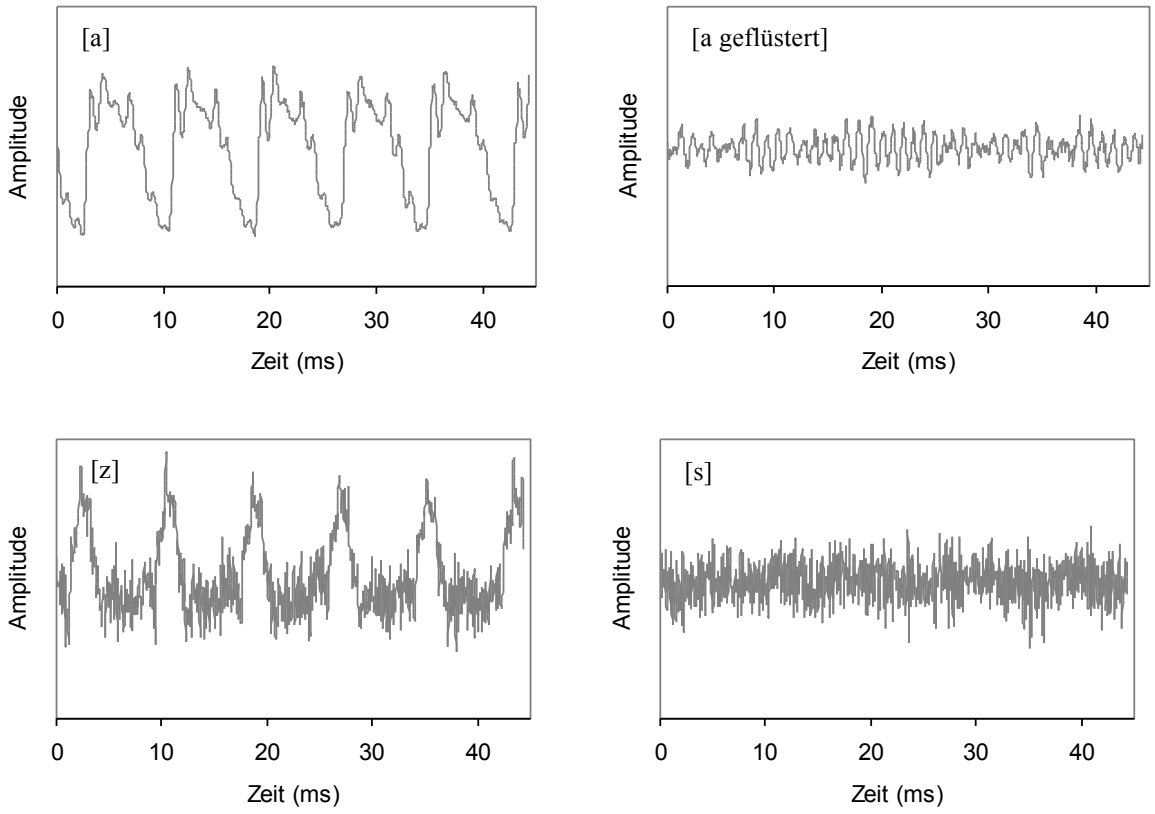

Abbildung 2.2 Zeitsignale stimmhafter und stimmloser Laute

Auffällig ist die quasiperiodische Struktur der stimmhaften Laute, verursacht durch die quasiperiodischen Luftstöße, mit denen der Vokaltrakt angeregt wird. Dieses Anregungssignal kann mit Hilfe geeigneter Instrumente aufgezeichnet werden. Das wahrscheinlich bekannteste ist der Laryngograph [FA71]: das Anregungssignal wird bestimmt, indem an den Kehlkopf eine Spannung angelegt wird, so daß er wie ein elektrischer Widerstand wirkt. Der gemessene Strom ist von der Öffnung der Stimmritze abhängig. Abbildung 2.3 zeigt ein mit diesem Verfahren aufgezeichnetes Anregungssignal, ein Laryngogramm.

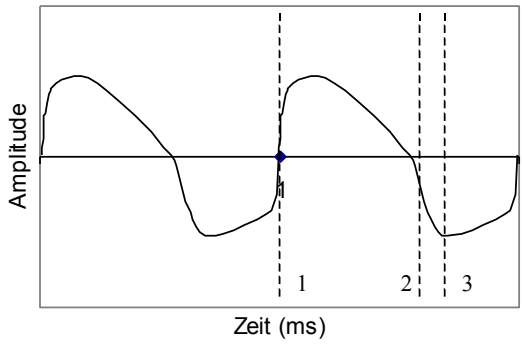

Abbildung 2.3 Laryngogramm [Har94]

Wie in dieser Abbildung zu sehen ist, kann eine einzelne Periode der Stimmbandschwingung in drei Phasen eingeteilt werden (nach [Ind87]). Die Anfangszeitpunkte der 3 Phasen sind:

1) Der Zeitpunkt des vollständigen Verschlusses der Stimmritze, gekennzeichnet durch das absolute Maximum innerhalb der Periode im differenzierten Laryngogramm.
2) Der Zeitpunkt des Öffnens der Stimmritze, gekennzeichnet durch das absolute Minimum innerhalb der Periode im differenzierten Laryngogramm.
3) Der Zeitpunkt der maximalen Öffnung der Stimmritze, gekennzeichnet durch das absolute Minimum innerhalb der Periode im Laryngogramm.

Die Grundperiodendauer T_0 ist als Zeitdauer zwischen zwei aufeinanderfolgenden Stimmritzenverschlüssen definiert. Die Grundfrequenz F_0 (pitch) ist dann festgelegt als:

$$F_0 = \frac{1}{T_0} \tag{2.1}$$

2.2.1 Sprecherabhängigkeit

Die durchschnittliche Grundfrequenz beträgt bei Männern etwa 150 Hz und bei Frauen ungefähr 200 Hz. Außerdem liegt die durchschnittliche Vokaltraktlänge bei Männern bei etwa 18 cm, bei Frauen bei etwa 13 cm. Die Bestimmung der Grundfre-

quenz läßt damit also Rückschlüsse auf das Geschlecht des Sprechers - genauer gesagt auf seine Vokaltraktlänge – zu: die Vokaltraktlänge ist annähernd umgekehrt proportional zur mittleren Grundfrequenz des Sprechers. Aufgrund dieser Korrelation können aus der Grundfrequenz einer Äußerung Parameter für die Sprechernormierung bestimmt werden. So ein Normierungsverfahren wird detaillierter in Kapitel 5.1 beschrieben.

2.2.2 Prosodie

Die Frequenz F_0 der Stimmbandschwingung wird durch den Luftdruck unterhalb der Stimmritze (subglottaler Druck) gesteuert, und durch den Anpreßdruck der Stimmlippen. Diese Grundfrequenz wird vom Gehör als Tonhöhe wahrgenommen. Ihr zeitlicher Verlauf - die Satzmelodie - ist zusammen mit Lautheit, Sprechtempo und zeitlicher Strukturierung der Äußerung ein wichtiges prosodisches Merkmal, das heißt ein Hinweis auf die Betonung einzelner Worte oder Satzteile, also Einheiten die größer sind als ein einzelnes Phonem. Beispielsweise macht ein Sprecher durch Steuerung des Verlaufs der Tonhöhe am Satzende deutlich, ob es sich bei diesem Satz um einen Aussage- oder einen Fragesatz handelt: Fragen ohne einleitendes Fragewort sind im Allgemeinen durch einen Anstieg der Grundfrequenz F_0 am Satzende gekennzeichnet, Aussagen durch einen Abfall. Diese Satzmodusmarkierung ist besonders bei kurzen Äußerungen wichtig. Zum Beispiel läßt dich durch Untersuchung des Grundfrequenzverlaufs in der Erwiderung des Sprechers A im Dialog

A: "Wir treffen uns also am Donnerstag ./?"
B: "Am Donnerstag ./?"

feststellen, ob Sprecher A seinen Satz als abschließende Feststellung verstanden haben will (terminal), oder als Frage, auf die er eine Bestätigung des Termins erwartet (interrogativ). Ähnliches gilt für die Erwiderung des Sprechers B. Angenommen A wird von ihm aufgrund ihres Grundfrequenzverlaufs als Feststellung interpretiert. Dann kann er A beispielsweise durch einen Anstieg des Grundfrequenzverlaufs in "Am Donnerstag?" mitteilen, daß er (B) sich nicht sicher ist, ob er das Datum richtig verstanden hat, und eine Bestätigung des Datums wünscht. Die Auswertung von Grundfrequenzverläufen in Äußerungen ist somit also für automatische Dialogsysteme von Bedeutung, um Hinweise auf den Satzmodus zu erhalten.
Grundfrequenzänderungen werden auch zur Betonung einzelner Worte und Satzteile eingesetzt, wodurch die Bedeutung des Satzes variiert werden kann. Wichtig sind Betonungsanalysen - also unter anderem auch Grundfrequenzverläufe - für die automatische Übersetzung gesprochener Sprache. Je nachdem auf welchem Satzteil die Betonung liegt, können so verschiedene Übersetzungen für einen Satz angemessen sein. Eine detaillierte Untersuchung solcher prosodischer Probleme findet sich in [Kom95].

2.2.3 Tonale Sprachen

Der zeitliche Verlauf der Grundfrequenz ist in manchen Sprachen nicht nur auf Wort- oder Satzgliedebene bedeutungstragend (Prosodie), sondern auch auf der Phonemebene. Beispiele für solche Sprachen finden sich im asiatischen Raum, wie etwa im Chinesischen. Bei diesen Sprachen kann der Grundfrequenzverlauf als zusätzliches Merkmal für Phonemmodellierung im Worterkenner dienen. In Kapitel 5.2 wird genauer auf diese Möglichkeit eingegangen.

2.3 Mathematische Modellierung der Spracherzeugung

Ein mathematisches Modell des Spracherzeugungsprozesses kann dazu beitragen, die wesentlichen Zusammenhänge dieses Vorgangs zu verdeutlichen, kann aber auch als Ausgangspunkt für die Entwicklung eines Systems zur Erzeugung synthetischer Sprache dienen.
Das Modell muß die wesentlichen Bestandteile des Artikulationssystems in angemessener Form berücksichtigen. Ein bekanntes Modell ist das sogenannte Source-Filter-Modell wie es beispielsweise in [Rab78] beschrieben wurde. Dieses geht von einem Signalerzeugungsprozeß aus, der aus zwei hintereinandergeschalteten linearen, zeitinvarianten Systemen besteht.
Der Anregungsschall, der an der Stimmritze erzeugt wird, wird durch eine Anregungsfunktion $A(z)$ modelliert. Bei der stimmlosen Artikulation entspricht diese Funktion einem Rauschsignal, das im zeitdiskreten digitalen Modell durch einen Zufallszahlengenerator erzeugt wird. Bei der stimmhaften Artikulation entspricht die Anregungsfunktion einem quasiperiodischen Impulszug. Die zeitlich veränderliche Vokaltraktform wird durch ein angemessenes Vokaltraktmodell $V(z)$ beschrieben, welches durch einen Satz von Vokaltraktparametern (Filterkoeffizienten) bestimmt wird.

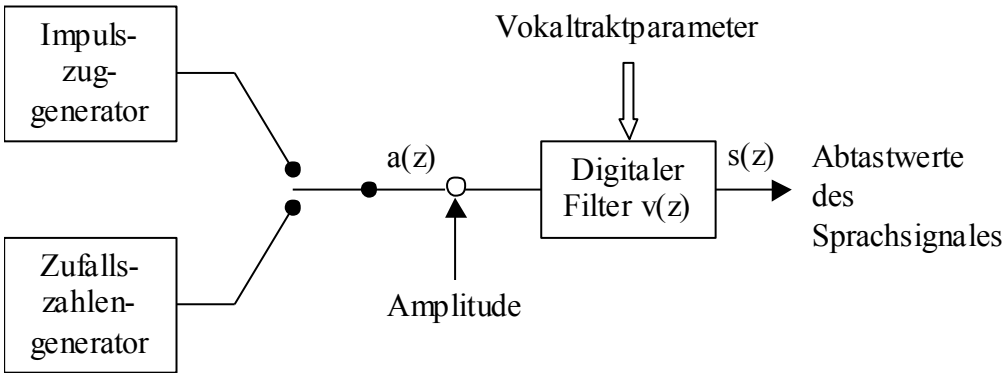

Abbildung 2.4 Digitales der Sprachproduktion (nach [Rab78])

Das diskrete Ausgangssignal $s(t)$ läßt sich als Faltung $s(t)=a(t)*v(t)$ schreiben, womit sich für die z-Transformierten die Darstellung $S(z)=A(z)\cdot V(z)$ ergibt.

3 Automatische Grundfrequenzbestimmung

Verfahren zur automatischen Grundfrequenzbestimmung sollen für ein vorliegendes Sprachsignal Informationen über die zugehörige Anregungsfunktion liefern. In dieser Arbeit werden diese Informationen im Rahmen eines automatischen Spracherkennungssystems weiterverarbeitet, und sollen letzten Endes dazu beitragen, die Erkennungsrate des Spracherkenners zu verbessern. Der Detaillierungsgrad der Informationen über die Anregungsfunktion hängt also von den Anforderungen der nachgeschalteten Spracherkennermodule ab. Der höchste Detailgrad ist erreicht, wenn die Anregungsfunktion vollständig aus dem Sprachsignal rekonstruiert wird. Diese Repräsentation ist allerdings äußerst redundant, da diese Funktion in stimmhaften Bereichen quasiperiodisch ist. Deshalb kann die Information über die Anregungsfunktion komprimiert werden, in dem man das Sprachsignal in stimmhafte und stimmlose Bereiche einteilt, und für die stimmhaften Anteile die Startzeitpunkte aller Perioden bestimmt. Diese Information wird beispielsweise von Verfahren der grundperiodensynchronen Analyse [Rab78] benötigt. Für eine Vielzahl von Anwendungen ist diese Information immer noch zu detailliert, für diese Anwendungen sind die Startzeitpunkte der einzelnen Grundperioden uninteressant. Hier reicht es bereits aus, an äquidistanten Zeitpunkten in den stimmhaften Sprachsegmenten - beispielsweise alle 10 ms - die vorliegende Grundfrequenz zu bestimmen.

3.1 Probleme bei der Grundfrequenzbestimmung

Die Bestimmung des Grundfrequenzverlaufes in Sprachsignalen ist aus verschiedenen Gründen keine einfache Aufgabe. Zwar existiert eine Vielzahl von Algorithmen für die Lösung dieser Aufgabe, allerdings stellt [Kie96 S.139] fest, daß es bisher noch keinen Algorithmus gibt, der für alle Anwendungen, alle Sprecher und alle Aufnahmebedingungen verläßlich und genau funktioniert. Die Schwierigkeit des Problems der Grundfrequenzbestimmung hat mehrere Ursachen. Ein Grund dafür ist, daß die Grundfrequenz verschiedener Sprecher sich in einem breiten Frequenzspektrum bewegt: sie reicht von 50 Hz bei besonders tief sprechenden Männern, bis zu 550 Hz bei hoch sprechenden Frauen oder Kindern. Dadurch kann es zu Verwechslungen zwischen der tatsächlichen Grundfrequenz und ihrer Oberwellen kommen. Außerdem kann selbst bei einem einzelnen Sprecher die Grundfrequenz innerhalb eines Satzes über mehrere Oktaven variieren, siehe Abbildung 3.1.

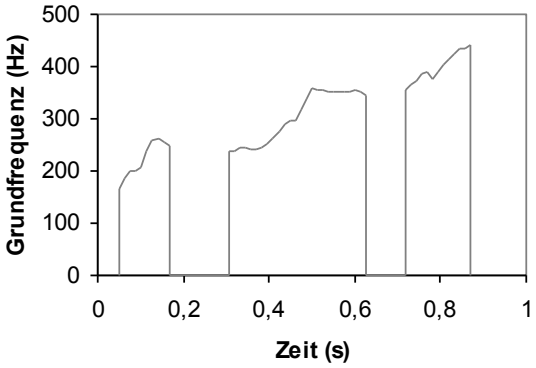

Abbildung 3.1 Stark variierende Grundfrequenz in einem Satz

Da die erste Formante (Resonanzfrequenz des Vokaltraktes) sich in einem Bereich von 200 bis etwa 800 Hz bewegt, kann es hier zur Verwechslung zwischen Grundfrequenz und erster Formante kommen. Außerdem ändern sich während des Sprechens die physikalischen Bedingungen des Vokaltraktes. Diese ändern sich zwar in den meisten Fällen relativ langsam, wenn man von Plosivlauten einmal absieht. Sie können etwa in Übergangsbereichen zwischen einzelnen Phonemen zu unregelmäßigen Formen im Sprachsignal führen, die das Auffinden von Grundperioden im Sprachsignal erheblich erschweren können.
Außerdem ist die Kehlkopfanregung nicht perfekt periodisch, die Grundperiodendauer kann von einer Periode zur nächsten um mehrere Prozent schwanken (Jitter).
Kurzzeitig kann es auch zu besonders extremen Unregelmäßigkeiten in der Anregung der Stimmbänder kommen. Diese werden als Laryngalisierungen bezeichnet, und vom Hörer als knarrende Stimme wahrgenommen. Laryngalisierungen können in verschiedene Unterklassen eingeteilt werden, Untersuchungen darüber wurden in [Bat93a] angestellt. Gründe für diese Phänomene sind zum Beispiel zu geringer Druck unterhalb der Stimmritze oder eine zu schwache Stimmbänderspannung. Einige Beispielsignale von laryngalisierten Sprachabschnitten sind in der folgenden Abbildung dargestellt.

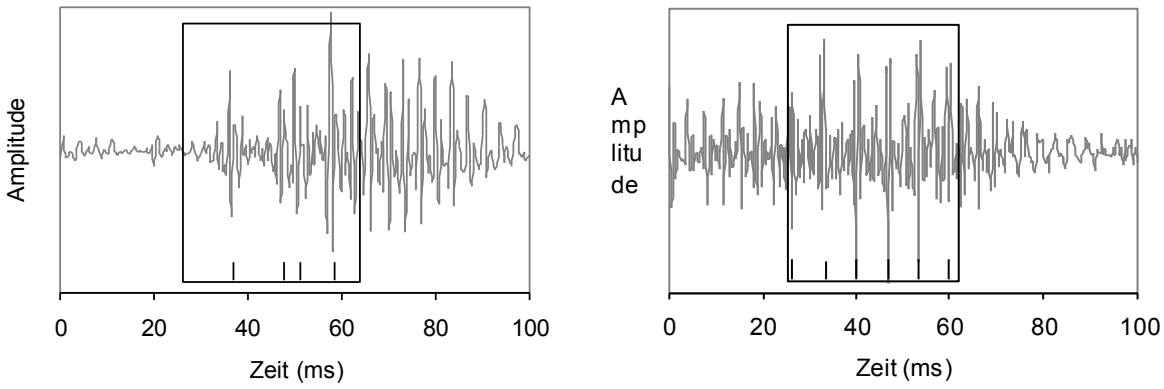

Abbildung 3.2 Laryngalisierungen: Aperiodizität und Oktavsprung

In den meisten Sprachen, wie auch der deutschen, wird nicht zwischen normal ausgesprochenen und laryngalisierten Phonemen unterschieden. Beispielsweise wird ein kurzzeitiger Oktavsprung vom Hörer zwar als Knarren wahrgenommen, er hat aber trotz der kurzzeitig verdoppelten Grundperiodendauer keinen Einfluß auf den empfundenen Tonhöhenverlauf. Das heißt, solche kurzen Störungen werden ignoriert und nicht als bedeutungstragend eingestuft. Allerdings werden in einigen westafrikanischen Sprachen und im Dänischen Laryngalisierungen zur Laut- oder Wortdifferenzierung benutzt [Har94 Kapitel 2.3.1]. Eine etwas umfassendere Beschreibung der Funktion von Laryngalisierungen findet sich in [Kie96].

3.2 Vorverarbeitung

Die Eingabedaten des automatischen Spracherkennungssystems, und damit auch der Grundfrequenzbestimmung, sind abgetastete und quantisierte Sprachsignale. Die Vorverarbeitung des Erkenners soll Störungen, wie zum Beispiel Rauschen, aus dem Signal entfernen, die Menge an redundanten Informationen im Sprachsignal reduzieren, und das Signal schließlich in eine für die Weiterverarbeitung geeignete Repräsentationsform bringen. Typische Abtastraten für Spracherkenner liegen zwischen 8 und 16 kHz. Da die Grundfrequenz normalerweise kleiner als 550 Hz ist, wird bei vielen Verfahren vor der Grundfrequenzbestimmung eine Tiefpaßfilterung mit einer Grenzfrequenz von > 1100 Hz durchgeführt. Diese Operation dient der Unterdrückung von Vokaltrakteinflüssen im Bereich über 1100 Hz und auch störendem hochfrequentem Rauschen. Zu beachten ist dabei, daß die Tiefpaßfilterung nicht zu einer Reduktion der Vokaltrakteinflüsse in den unteren Frequenzbereichen des Spektrums führt. Im Anschluß an die Tiefpaßfilterung kann eine Neuabtastung des Signals vorgenommen werden, ohne daß Informationen verloren gehen. Damit wird gleichzeitig eine Reduktion der von den Folgestufen der Grundfrequenzverfolgung zu verarbeitenden Datenmenge erreicht, und infolgedessen eine Verkürzung der Laufzeit der weiteren Verarbeitungsschritte.

3.3 Rekonstruktion des Anregungssignals

Um aus dem Sprachsignal $s(t)$ das Anregungssignal an der Stimmritze zu rekonstruieren, muß der Einfluß des Vokaltraktes rückgängig gemacht werden. Der Vorgang der Rekonstruktion der beiden Signale $a(t)$ und $v(t)$ aus dem Ausgangssignal $s(t)$ wird inverse Filterung genannt.

3.3.1 Lineare inverse Filterung

Nach dem in Kapitel 2.3 vorgestellten linearen Modell der Spracherzeugung wird $s(t)$ durch Faltung des Anregungssignals a(t) mit der Vokaltraktfilterfunktion $v(t)$ gebil-

det. Damit ergibt sich die Fouriertransformation $S(z)$ des Sprachsignals $s(t)$ aus der Multiplikation der Fouriertransformierten von $a(t)$ und $v(t)$:

$$S(z) = A(z) \cdot V(z) \tag{3.1}$$

Die Anregungsfunktion $a(t)$ könnte also aus dem Sprachsignal $s(t)$ rekonstruiert werden:

$$A(z) = \frac{S(z)}{V(z)} = S(z) \cdot I(z) \text{ mit } I(z) = \frac{1}{V(z)} \tag{3.2}$$

Der Filter $I(z)$ wird inverser Filter genannt. Die Rekonstruktion gelingt für die Frequenz z allerdings nur unter der Bedingung, daß $V(z)$ nicht verschwindet. Wird der Vokaltraktfilter als

$$V(z) = \frac{1}{I(z)}, \text{ mit } I(z) = \sum_{i=0}^{M} a_i z^{-i} \tag{3.3}$$

in z^{-1} modelliert, so können die Vokaltraktparameter a_i mit dem Verfahren der linearen Vorhersage bestimmt werden. Bei diesem Verfahren werden die Abtastwerte des Zeitsignals $s(t)$ durch eine Linearkombination der vorangegangenen M Abtastwerte vorhergesagt [Wie66].
Ein Problem der Modellierung der Vokaltraktübergangsfunktion $V(z)$ wie oben ist, daß der Einfluß des Nasalraumes auf das resultierende Sprachsignal nur ungenügend mitmodelliert wird [ST95]. Bei nasaler Artikulation werden nämlich bestimmte Frequenzbereiche im Signal gedämpft, diese Dämpfung würde jedoch durch eine kompliziertere rationale Übertragungsfunktion der Form

$$V(z) = \frac{B(z)}{I(z)} \tag{3.4}$$

angemessener beschrieben werden. Filter dieser Art sind jedoch nicht mehr linear, und dementsprechend schwieriger handhabbar.

3.3.2 Inverse Filterung mit KNN

Da die Vokaltraktübergangsfunktion nur ungenügend durch ein lineares Filter approximiert wird, liegt es nahe, zu nichtlinearen Filtern überzugehen. In [Den92,Kah93] wird der Ansatz beschrieben, künstliche neuronale Netze (KNN) für die Rekonstruktion der Anregungsfunktion einzusetzen.
Die Eingabe in das Netz besteht aus 78 Abtastwerten des tiefpaßgefilterten und mit 2000 Hz abgetasteten Sprachsignals, also aus einem 39 msec langen Signalauschnitt. Zusätzlich wird das Sprachsignal noch auf den Wertebereich [1,-1] normiert. Die Ausgabeschicht des Netzes besteht aus einem einzelnen Knoten, dessen Aktivierungsgrad als Abtastwert des rekonstruierten Anregungssignals betrachtet wird. Um die gesamte Abtastfolge dieses Signals zu erhalten, wird das Eingabefenster Punkt für Punkt über das Sprachsignal geschoben. Die Ausgaben in jedem Schritt werden

zu einer Ausgabefolge verkettet, die letztendlich noch mit mehreren Mittelwertfiltern geglättet wird. Mehrere Netztopologien wurden im Hinblick auf die Qualität der erzeugten F_0-Kontur (Verfahren siehe unten) untersucht, die besten Ergebnisse wurden mit 3 verborgenen Schichten mit je 100 Knoten erzielt.
Trainiert wird dieses Netz mit Paaren von Sprachsignalen und der zugehörigen Anregungsfunktion, die mittels eines Laryngographen aufgezeichnet wurde. Dabei wird die gewünschte Ausgabe auf den Abtastwert der Anregungsfunktion gesetzt, der sich in der Mitte des Eingabefensters befindet. Für das Training wurde der Quick-Propagation Algorithmus zusammen mit der gewohnten Minimierung des mittleren quadratischen Fehlers eingesetzt.
Zum Vergleich verschiedener Netztopologien mußte ein geeigneter Gütemaß gefunden werden. Da das Netz in [Den93] letztendlich zur Bestimmung von F_0-Konturen eingesetzt werden sollte, war der Vergleich des mittleren quadratischen Fehlers kein geeignetes Gütemaß. Statt dessen wurden auf der rekonstruierten Anregungsfunktion an äquidistanten Zeitpunkten Kurzzeitenergiespektren berechnet, in denen durch einfache Maximumsuche die Grundfrequenz zu diesem Zeitpunkt bestimmt wurde. Diese Methode der Grundfrequenzbestimmung lieferte allerdings im Vergleich mit einem anderen Grundfrequenzverfahren (DPF0-SEQ, siehe Kapitel 3.5.2) schlechtere F_0-Konturen. In [Kah93] wurde die inverse Filterung mit neuronalen Netzen zur Detektion von Laryngalisierungen eingesetzt. Damit wurden deutlich bessere Ergebnisse erzielt als mit linearer inverser Filterung. Da die inverse Filterung mittels KNN sehr rechenintensiv ist, wurde dieser Ansatz dort nicht weiter verfolgt.

3.4 Verfahren zur Detektion stimmhafter Sprachsegmente

Ein Teil der Verfahren zur Bestimmung des Grundfrequenzverlaufs arbeiten ausschließlich auf stimmhaften Regionen des Sprachsignals, denn nur in diesen Bereichen liegt überhaupt eine Stimmbandschwingung vor. Die Alternative besteht darin, die stimmhaft/stimmlos (‚SH/SL') Entscheidung mit der Bestimmung des Grundfrequenzverlaufes zu kombinieren. Da die Verfahren aus der erstgenannten Gruppe auf eine Unterteilung des Sprachsignals in eine Folge von sich abwechselnden stimmhaften und stimmlosen Bereichen angewiesen sind, wird im Folgenden kurz auf Möglichkeiten zur Unterscheidung der beiden Phonationsarten SH und SL eingegangen.
Stimmhafte Regionen im Sprachsignal zeichnen sich im Vergleich zu stimmlosen durch eine niedrigere Nulldurchgangsrate, größere Energie und höhere maximale Signalamplituden aus. Viele Verfahren zur Unterscheidung stimmhafter und stimmloser Bereiche teilen das Sprachsignal in kurze, typischerweise 10 ms lange, Abschnitte ein, und messen auf jedem dieser Segmente die Kurzzeitenergie, Nulldurchgangsrate und maximale Amplitude. Diese Messungen aus einem Segment werden in einem Merkmalsvektor zusammengefaßt, was eine Folge von 100 Merkmalsvektoren pro Sekunde ergibt. Mit einer der bekannten Klassifikationsmethoden wird für jeden Merkmalsvektor entschieden, ob er in die Klasse stimmhaft, stimmlos oder Stille ge-

hört. Ein Schwellwertverfahren, dessen Schwellwerte durch ein Koordinatenabstiegsverfahren optimiert wurden, wird in [Hag95] beschrieben. Als Klassifikatoren sind auch neuronale Netze einsetzbar, oder Hidden Markov Modelle, bei denen die Emissionsverteilungen durch parametrisierte Normalverteilungsmixturen approximiert werden können. Alle diese Verfahren können bei Vorliegen einer genügend großen Trainingsdatenmenge automatisch optimiert werden.
Da der Grundfrequenzalgorithmus, der im Rahmen der vorliegenden Arbeit entwickelt wurde, keine Segmentierung des Sprachsignals in SH/SL-Bereiche voraussetzt, wird an dieser Stelle nicht detaillierter auf die SH/SL-Unterscheidung eingegangen.
Fehlerraten einiger SH/SL-Klassifikatoren sind in Tabelle 4.12 aufgelistet [Kie96].

3.5 Verfahren zur Verfolgung des Grundfrequenzverlaufs

Die Eingabe für Verfahren zur Verfolgung des Grundfrequenzverlaufs besteht aus dem abgetasteten und quantisierten Sprachsignal, das in der Vorverarbeitungsstufe geeignet aufbereitet wird (Kapitel 3.2). Manche Methoden verarbeiten das Sprachsignal nicht direkt, sondern operieren auf dem von einem geeigneten Verfahren aus dem Sprachsignal rekonstruierten Anregungssignal (Kapitel 3.3). Die F_0-Verfahren lassen sich laut [Hes83] nach der erzeugten Ausgabe in zwei Hauptgruppen einteilen, in periodensynchrone Verfahren und Kurzzeitverfahren.
Periodensynchrone Verfahren geben eine Folge von Grundperiodenanfängen aus. Die Ausgabe von Kurzzeitverfahren ist weniger detailliert. Diese Verfahren zerlegen das Sprachsignal in eine Folge äquidistanter Zeitfenster (frames) und berechnen für jedes Fenster einen Schätzwert für die darin enthaltene Grundfrequenz.

Bevor genauer auf die einzelnen Verfahren eingegangen wird, soll erst einmal gezeigt werden, wie sich die Stimmbandanregung im Zeitbereich und im Fourierspektrum des Sprachsignals bemerkbar macht.

Die Vokaltraktform ändert sich nur relativ langsam: betrachtet man einen kurzen Ausschnitt aus dem Sprachsignal - etwa im Bereich von 10 ms -, kann man den Vokaltrakt als näherungsweise konstant ansehen. Nach dem in Abschnitt 2.3 vorgestellten mathematischen Modell der Spracherzeugung erhält man das Sprachsignal $s(t)$ durch Faltung der Anregungsfunktion $a(t)$ mit der Vokaltraktfilterfunktion $v(t)$. Dabei kann die Anregungsfunktion in stimmhaften Bereichen durch einen quasiperiodischen Impulszug modelliert werden. Für den vereinfachten Fall, daß das Anregungssignal periodisch mit der Grundperiodendauer T_0 ist, gilt:

$$a(t) = \sum_{k=-\infty}^{\infty} \delta(t - k \cdot T_0) \tag{3.5}$$

Dann ist auch das resultierende Modellsprachsignal $s(t)$ eine periodische Funktion mit der Periodendauer T_0. Das Fourierspektrum $S(f)$ des Modellsprachsignals entsteht aus der Multiplikation der Fouriertransformierten $V(f)$ und $A(f)$.

$$s(t)=a(t)*v(t)$$

$$S(f)=A(f)\cdot V(f)$$

Da die Fouriertransformierte der Kammfunktion $a(t)$ mit Impulsabstand T_0 eine Kammfunktion mit Impulsabstand $F_0=1/T_0$ ist, ist das Modellsprachspektrum nur für $f=k\cdot F_0$ mit $k\in Z$ ungleich Null. Diese Zusammenhänge werden in der unteren Abbildung dargestellt.

Weiterhin gilt:

$$F(s(t))=F(a(t))\cdot F(v(t))$$

$$|F(s(t))|=|F(a(t))|\cdot|F(v(t))|$$

$$log|F(s(t))|=log|F(a(t))|+log|F(v(t))|$$

$$F^{-1}\{|logF(s(t))|\}=F^{-1}\{log|F(a(t))|\}+F^{-1}\{log|F(v(t))|\}$$

Diese Methode, die das Signal in seine Faltungskomponenten $a(t)$ und $v(t)$ zerlegt, heißt Cepstralanalyse und ist ein Spezialfall der Klasse homomorpher Analyseverfahren [Rab78]. Die Transformierte $F^{-1}\{|logF(s(t))|\}$ wird (reelles) Cepstrum genannt. Die periodische Anregungsfunktion $a(t)$ erscheint im Cepstrum als Kammfunktion $F^{-1}\{log|F(a(t))|\}$. Das heißt, im Cepstrum finden sich Impulse bei Vielfachen der Grundperiodendauer T_0. Abbildung 3.7 zeigt das Cepstrum eines stimmhaften Ausschnittes aus einem Sprachsignal.

Nach diesen Betrachtungen lassen sich also drei grundsätzliche Methoden zur Bestimmung der Grundfrequenz aus dem Sprachsignal ableiten. Im Zeitbereich kann nach Periodizitäten im Sprachsignal gesucht werden. Im Spektralbereich kann das Fourierspektrum des Signals auf Vorkommen von Vielfachen der Grundfrequenz F_0 untersucht werden. Und im Cepstrum kann nach Vielfachen der Grundperiodendauer T_0 gesucht werden.

3.5.1 Periodensynchrone Verfahren

Die periodensynchronen Verfahren zur Grundfrequenzbestimmung liefern eine Folge von Grundperiodengrenzen. Die Grundperiodendauer wurde in Abschnitt 2.2 als Dauer zwischen zwei aufeinanderfolgenden Stimmritzenverschlüssen definiert. Der genaue Zeitpunkt des Stimmritzenverschlusses ist für die anschließenden Verarbeitungsstufen jedoch im Allgemeinen uninteressant, abgesehen davon ist eine genaue Zuordnung dieses Zeitpunktes zu einer bestimmten Stelle im Sprachsignal schwierig [Har94]. Deshalb bieten sich als zu markierende Periodengrenzen solche Stellen im Signal an, die leicht detektierbar sind. In Frage kommen hier Maxima bzw. Minima,

oder positive bzw. negative Nulldurchgänge. Die periodensynchronen Verfahren versuchen diese Periodengrenzen zu lokalisieren, indem nach sich periodisch wiederholenden Strukturmerkmalen im Sprachsignal gesucht wird. Ein relativ alter Vertreter dieser Gruppe von Algorithmen ist beispielsweise in [Rab78 Kapitel 4.5] beschrieben. Dort werden aus lokalen Extrema im Sprachsignal Impulszüge gebildet, welche geeignet kombiniert werden.
Im Folgenden wird auf einen anderen Vertreter dieser Gruppe von Algorithmen etwas genauer eingegangen, da dieses Verfahren erst vor wenigen Jahren entwickelt wurde, und wir die Leistungsfähigkeit des von uns entwickelten Algorithmus mit diesem Verfahren auf den gleichen Referenzdaten vergleichen konnten.

Das periodensynchrone PDDP Verfahren

Dieses Verfahren wurde an der Universität Erlangen-Nürnberg im Rahmen einer Diplomarbeit implementiert [Har94]. PDDP steht für ‚Periodendetektion mittels dynamischer Programmierung'. Mit diesem können verschiedene Arten von Signalen verarbeitet werden: es verarbeitet sowohl reine, als auch invers gefilterte Sprachsignale.
Da das Verfahren nach positiven Nulldurchgängen sucht, wird im Vorverarbeitungsschritt eine Mittelwertbefreiung durchgeführt. Auf eine Tiefpaßfilterung und Neuabtastung wird verzichtet, da die Position der Nulldurchgänge möglichst genau detektiert werden soll.
Im nächsten Schritt wird das Signal in stimmhafte und stimmlose Abschnitte segmentiert. Der SH/SL-Klassifikator besteht aus einer Schwellwertentscheidung auf den drei Merkmalen Nulldurchgangsrate, Signalenergie und Signalmaximum, welche auf kurzen Zeitfenstern gemessen werden.
Auf jedem stimmhaften Sprachsegment - wahlweise auch auf der gesamten Äußerung - wird daraufhin die mittlere Grundfrequenz mit einem Kurzzeitverfahren bestimmt. Hier wurde das ADMF-Verfahren (‚average difference magnitude function') verwendet, dessen genaue Implementation in [Har94] beschrieben ist. Im Zusammenhang mit Kurzzeitverfahren (Kapitel 3.5.2) wird noch kurz auf ADMF eingegangen.
Im folgenden Schritt wird eine Folge aller positiven Nulldurchgänge im Sprachsignal aufgestellt. Diese Folge enthält außer den gesuchten Periodengrenzen allerdings im Normalfall eine Vielzahl von Nulldurchgängen, die auf den Einfluß des Vokaltraktes zurückzuführen sind. Das heißt, diese Folge enthält die Folge der gesuchten Grundperiodengrenzen als Teilfolge.

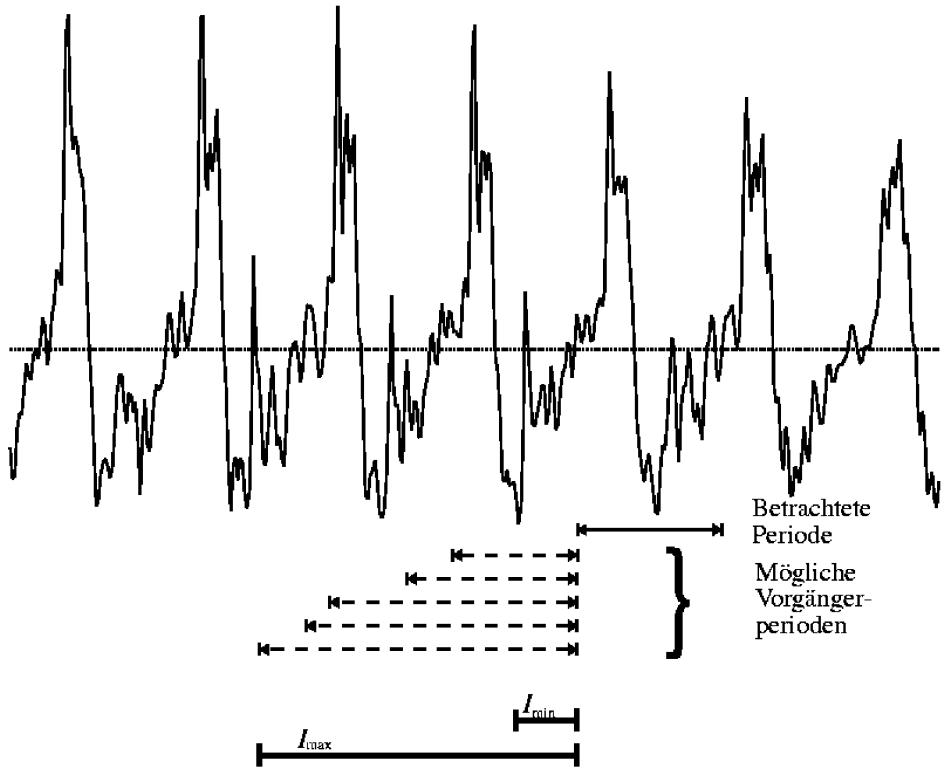

Abbildung 3.3 Auswahl eines Hypothesenvorgängers [Har94]

Um die korrekte Teilfolge der Grundperiodengrenzen in der Folge von Nulldurchgängen zu finden, wird eine Kostenfunktion definiert. Diese bewertet für jede Teilfolge von Grundperiodengrenzenhypothesen sowohl eine einzelne Grundperiode der Hypothese, als auch die Ähnlichkeit benachbarter Grundperioden. Zu diesem Zweck werden über 20 heuristische Teilkostenfunktionen definiert, die zum Beispiel Maximum, Energie und Mittelwert zweier aufeinanderfolgender Grundperiodenhypothesen vergleichen. Diese Teilkostenfunktionen werden zu einer globalen Kostenfunktion kombiniert, die effizient mittels dynamischer Programmierung ausgewertet werden kann.
Das Interessante an der Kombination der Teilkostenfunktionen ist, daß sie mittels eines neuronalen Netzes automatisch optimiert wird. Das Netz wird dabei benutzt, um die Ähnlichkeit zweier aufeinanderfolgender Grundperiodenhypothesen aufgrund der heuristischen Teilkostenfunktionen zu bestimmen. Weitere Details zu Netztopologie und Training finden sich in [Har94].

3.5.2 Kurzzeitverfahren

Kurzzeitverfahren zerlegen das Sprachsignal in eine Folge äquidistanter Zeitfenster (frames) und berechnen für jeden dieser Signalausschnitte einen Schätzwert für die darin enthaltene Grundfrequenz. Am Anfang von Kapitel 3.5 wurde bereits gezeigt, daß man aus der Untersuchung von Periodizitäten im Zeitsignal $s(t)$ Hinweise auf die Grundfrequenz ableiten kann. Alternativ kann man im Spektrum nach Vielfachen der

Grundfrequenz F_0 suchen, oder im Cepstrum nach Vielfachen der Grundperiodendauer T_0.

Untersuchung des Zeitsignals:

Um Periodizitäten im Zeitbereich zu detektieren, kann man Korrelationsverfahren verwenden. Das normierte Kreuzkorrelationsverfahren wird beispielsweise in [Med-91] hergeleitet:

Für jeden Zeitpunkt t_0 und jede Fensterbreite r werden zwei Signale $x_r(t,t_0)$ und $y_r(t,t_0)$ definiert:

$$\begin{aligned} x_r(t,t_0) &= s(t)\cdot w_r(t-t_0) \\ y_r(t,t_0) &= s(t+r)\cdot w_r(t-t_0) \end{aligned} \tag{3.6}$$

wobei $w_r(t)$ ein rechteckiges Fenster der Länge r ist, d.h.

$$w_r(t) = \begin{cases} 1, \text{ für } 0 \le t \le r \\ 0, \text{ sonst} \end{cases} \tag{3.7}$$

Damit sind x_r und y_r zwei aufeinanderfolgende Ausschnitte aus dem Sprachsignal ab t_0, wobei beide Segmente die Länge r haben. Unter der Annahme, daß das Sprachsignal in diesen beiden Ausschnitten stimmhaft ist, der Vokaltrakt sich nur langsam verändert, und die Grundperiodendauer etwa konstant ist, wird r gleich der Grundperiodendauer T_0 in diesem Segment gesetzt. Somit gilt:

$$x_{T0}(t,t_0) = a(t_0) y_{T0}(t,t_0) + e(t,t_0) \tag{3.8}$$

wobei $a(t_0)$ ein unbekannter positive Faktor für die Amplitudenmodulation der beiden Grundperioden ist. Dieser modelliert die mögliche Veränderung der Energie der Anregungsimpulse. Der Fehlerterm $e(t,t_0)$ spiegelt Unterschiede zwischen beiden Grundperioden wieder, hervorgerufen durch Hintergrundgeräusche, Kanalrauschen, oder langsame Vokaltraktveränderungen.
Allerdings ist die Grundperiodendauer zum Zeitpunkt t_0 nicht bekannt, sondern diese soll aus dem Sprachsignal bestimmt werden. Dazu wird aus der Menge aller möglichen T_0-Kandidaten derjenige ausgewählt, der den Unterschied $e(t,t_0)$ zwischen beiden Perioden minimiert. Das führt zu dem folgenden Problem der Minimierung des normierten quadratischen Fehlers:

$$T_0 = \underset{r, a_r(t_0)>0}{\operatorname{argmin}} \left\{ J_r(t_0) = \frac{\int_{t_0}^{t_0+r} [x_r(t,t_0) - a_r(t_0) y_r(t,t_0)]^2 dt}{\int_{t_0}^{t_0+r} [x_r(t,t_0)]^2 dt} \right\} \tag{3.9}$$

Der Normierungsterm im Nenner kompensiert die verschiedenen Längen der Ausschnitte aus dem Sprachsignal, und die uneinheitliche Energieverteilung in Sprachsignal.
Um den optimalen Wert für $a_r(t_0)$ zu erhalten, wird $J_r(t_0)$ für festes r und t_0 nach dem gesuchten $a_r(t_0)$ abgeleitet, und die Ableitung gleich Null gesetzt:

$$\frac{\partial J_r(t_0)}{\partial a_r(t_0)} = \frac{\int_{t_0}^{t_0+r} 2 \cdot [x_r(t,t_0) - a_r(t_0) y_r(t,t_0)] \cdot y_r(t,t_0) dt}{\int_{t_0}^{t_0+r} [x_r(t,t_0)]^2 dt}$$

$$a_r(t_0) = \frac{\int_{t_0}^{t_0+r} x_r(t,t_0) \cdot y_r(t,t_0)}{\int_{t_0}^{t_0+r} [y_r(t,t_0)]^2} \tag{3.10}$$

Einsetzen des optimalen a(t_0) in $J_r(t_0)$ liefert

$$J_r(t_0) = 1 - \rho_{r,t_0}(x,y)^2 \text{ mit } \rho_{r,t_0}(x,y) = \frac{\int_{t_0}^{t_0+r} x_r(t,t_0) \cdot y_r(t,t_0)}{\sqrt{\int_{t_0}^{t_0+r} [x_r(t,t_0)]^2} \cdot \sqrt{\int_{t_0}^{t_0+r} [y_r(t,t_0)]^2}} \tag{3.11}$$

$\rho_{r,t0}(x,y)$ wird als normierte Kreuzkorrelation zwischen den Segmenten x und y bezeichnet. Da praktisch nicht direkt mit einer kontinuierlichen, sondern einer abgetasteten Funktion gearbeitet wird, lassen sich die Funktionen x und y als Vektoren der Dimension r repräsentieren. Im Diskreten kann die normierte Kreuzkorrelationsfunktion dann als

$$\rho_{r,t_0}(x,y) = \frac{(x,y)}{|x| \cdot |y|} \tag{3.12}$$

geschrieben werden. Das Ergebnis des Optimierungsproblems ist damit

$$T_0 = \arg\max_r \rho_{r,t_0}(x,y) \tag{3.13}$$

Ein Problem dieser Korrelationsmethode ist, daß für kleine r nur zwei sehr kurze aufeinanderfolgende Signalabschnitte in die Berechnung des Korrelationskoeffizienten eingehen. Der resultierende Korrelationskoeffizient über diesen kurzen Abschnitten kann groß sein, wenn das Eingangssignal größtenteils niedrigfrequente Signalanteile aufweist, obwohl im Signal keine Periodizität mit der Periodendauer r vorliegt. Abhilfe schafft die Betrachtung von nicht nur zwei aufeinanderfolgenden Perioden, sondern die Betrachtung eines größeren Signalauschnittes. Bei Wahl einer konstanten Fensterbreite t_{fenster} unabhängig von der Zeitdifferenz r zwischen beiden Signalab-

schnitten muß lediglich die Fensterfunktion $w_r(t)$ in Gleichung 3.7 folgendermaßen modifiziert werden:

$$w_r(t) = \begin{cases} 1, \text{für } 0 \le t \le t_{fenster} \\ 0, \text{sonst} \end{cases} \tag{3.14}$$

Die folgende Abbildung zeigt 2 Sprachsignale mit der zugehörigen Kreuzkorrelationsfunktion nach Gleichung 3.14. Wie man hier erkennen kann, weist die Korrelationsfunktion des stimmhaften Signals ein Maximum bei t=T_0 auf. Da ein Signal mit der Periodendauer T_0 auch periodisch mit Periodendauer $k \cdot T_0$ (k∈N) ist, finden sich auch Maxima bei Vielfachen von T_0. Allerdings sind Sprachsignale nicht perfekt periodisch. Weil sich die Vokaltraktkonfiguration während der Artikulation langsam verändert, zeigen sich zwar in aufeinanderfolgende Grundperioden starke Ähnlichkeiten, weiter auseinanderliegende Grundperioden weisen jedoch im allgemeinen stärkere Unterschiede auf. Diese wachsenden Differenzen machen sich in der Kreuzkorrelationsfunktion dadurch bemerkbar, daß die Höhe der lokalen Maxima bei $k \cdot T_0$ mit wachsendem k immer mehr abfällt (Kurve rechts oben).

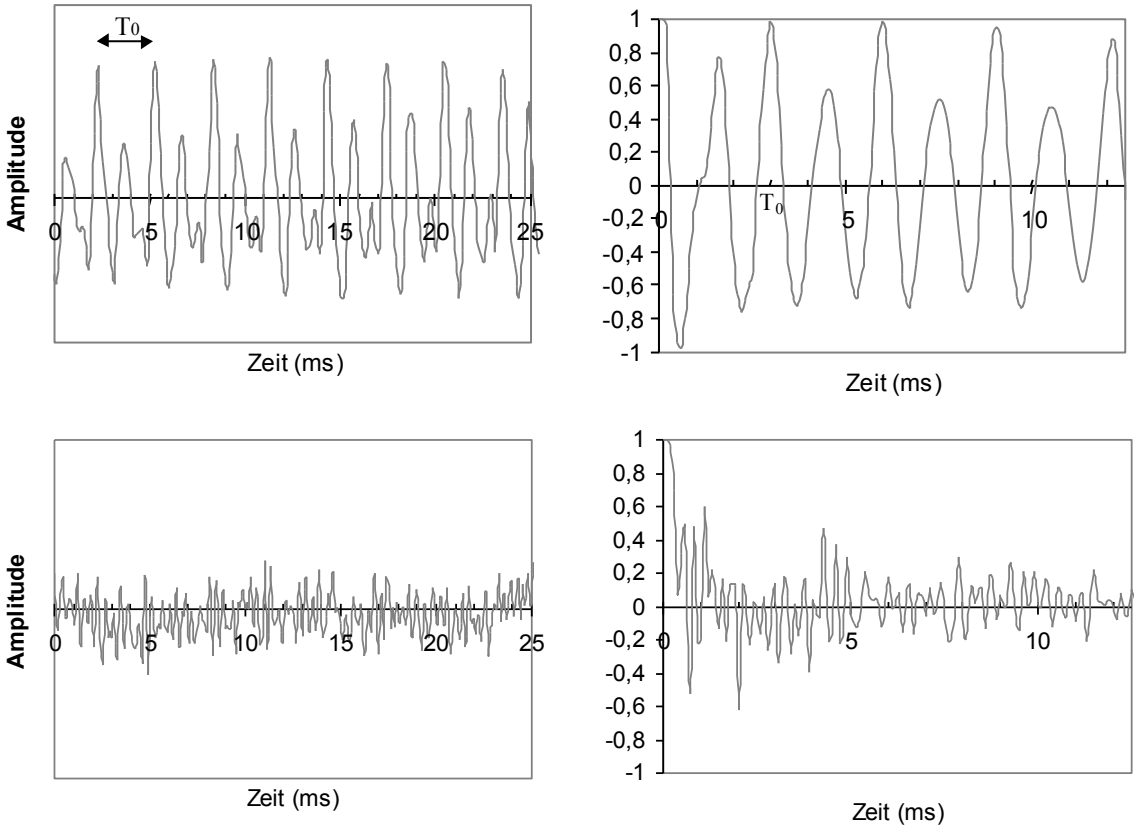

Abbildung 3.4 normierte Kreuzkorrelation stimmhafter und stimmloser Signale (links Zeitsignale, rechts Korrelationsfunktion)

Eine andere Funktion die zur Detektion von Periodizitäten im abgetasteten Zeitsignal $x(k)$ geeignet ist, ist die Autokorrelationsfunktion $\phi(k)$ [Rab78].

$$\varphi(k) = \sum_{m=-\infty}^{\infty} x(m)x(m+k) \tag{3.15}$$

Diese Funktion hat ihr globales Maximum bei *k=0*. Ist das Signal $x(k)$ periodisch mit einer Periodendauer von p Samples, gilt $\phi(k)=\phi(k+p)$. Da das stimmhafte Sprachsignal aber nur innerhalb kurzer Ausschnitte als quasiperiodisch angesehen werden kann, wird in [Rab78] die Kurzzeitautokorrelationsfunktion $\phi_n(k)$ definiert. Diese betrachtet einen mit einer Fensterfunktion $w(k)$ ausgeschnittenen Abschnitt des Sprachsignals, der durch Multiplikation des Sprachsignals $x(k)$ mit der um n verschobenen Fensterfunktion $w(k)$ berechnet wird.

$$\varphi_n(k) = \sum_{m=-\infty}^{\infty} x(m)w(n-m)x(m+k)w(n-k-m) \tag{3.16}$$

Bei der Wahl einer Rechteckfunktion mit einer Länge von N Abtastwerten als Fensterfunktion ergibt sich

$$w(k) = \begin{cases} 1 \text{ für } 0 \le k < N \\ 0 \text{ sonst} \end{cases}$$

$$\varphi_n(k) = \sum_{m=0}^{N-1-k} x(n+m)x(n+m+k) \tag{3.17}$$

Die folgende Abbildung zeigt die Kurzzeitautokorrelation für das stimmhafte Sprachsignal in Abbildung 3.4.

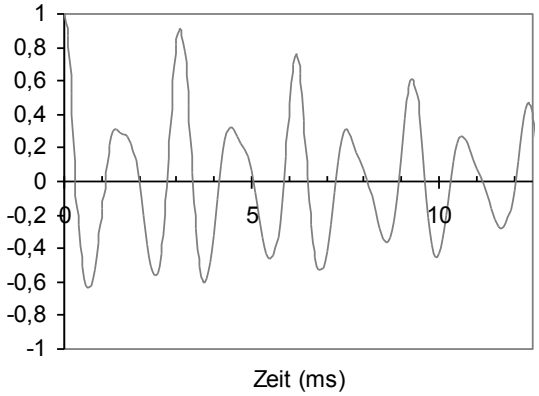

Abbildung 3.5 Kurzzeitautokorrelation für das stimmhafte Signal in Abbildung 3.4

Verglichen mit der normierten Kreuzkorrelation (siehe vorletzte Abbildung) fallen die lokalen Maxima bei der Autokorrelation mit steigendem k im allgemeinen stärker ab, da mit wachsendem k weniger Abtastwerte (N-1-k) in die Summe eingehen.

Weil die Berechnung der Autokorrelationsfunktion über die obige Formel eine beträchtliche Zahl von Multiplikation ($O(N^2)$) benötigt, wurde in [Ros74,Rab78] eine alternative Funktion zur Periodendetektion vorgeschlagen. Diese Funktion basiert auf

der Idee, daß für eine periodische Funktion $x(k)$ der Periode p die Folge $d(n)=x(n)-x(n-k)$. für $k=0,1p,2p,...$ Null ist. Diese Betrachtung führt zur Kurzzeit-ADMF (‚average difference magnitude function') $\gamma_n(k)$:

$$\gamma_n(k) = \sum_{m=-\infty}^{\infty} | x(m)w(n-m) - x(m+k)w(n-k-m) | \tag{3.18}$$

Bei Verwendung eines Rechteckfensters wie eben bei der Autokorrelation ergibt sich:

$$\gamma_n(k) = \sum_{m=0}^{N-1-k} | x(n+m) - x(n+m+k) | \tag{3.19}$$

Im Vergleich zur Autokorrelation befinden sich hier bei Vielfachen der Periodendauer Minima statt Maxima. Die Multiplikationsoperation wird durch eine Differenz und eine Betragsbildung ersetzt, was auf älteren Prozessoren zu einem erheblichen Geschwindigkeitsgewinn bei der Berechnung der ADMF im Vergleich zur Autokorrelation bringt. Bei modernen Prozessoren, bei denen die Multiplikation in Hardware implementiert ist, ist allerdings kein solcher Geschwindigkeitsgewinn mehr zu erwarten.

Um Grundperioden der Länge kleiner gleich 20 ms aufzuspüren - was einer erwarteten Grundfrequenz von mindestens 50 Hz entspricht - müssen die drei genannten Funktionen auf Fenstern der Mindestlänge 40 ms ausgewertet werden, da das Eingangssignal zur Detektion von Periodizitäten wenigstens 2 Grundperioden aufweisen muß. Der Aufwand aller dieser Funktionen in Bezug auf die Fensterlänge N ist $O(N^2)$. Ein Vorteil der Autokorrelationsfunktion ist, daß sich der Aufwand für ihre Berechnung auf dem Umweg über den Spektralbereich auf O(NlogN) verringert: die Korrelationsfunktion läßt sich nach [Jäh97] folgendermaßen berechnen:

$$\int_{-\infty}^{\infty} g(x')h(x+x')dx' = F^{-1}(G(z)\cdot H^*(z)) \tag{3.20}$$

Dabei sind g und h komplexe Funktionen, deren Fouriertransformierte $G(z)$ und $H(z)$ existieren. Im Falle der Autokorrelation gilt $h=g$, die Multiplikation im Spektralbereich kann dann durch Berechnung des Energiespektrums von $g(t)$ ersetzt werden:

$$\int_{-\infty}^{\infty} g(x')g(x+x')dx' = F^{-1}(G(z)\cdot G^*(z)) = F^{-1}(|G(z)|^2) \tag{3.21}$$

Bei der praktischen Berechnung der Autokorrelation mit Hilfe der DFT muß darauf geachtet werden, daß hierbei die Autokorrelation des zyklisch fortgesetzten Signalausschnitts berechnet wird. Zur Vermeidung von Unstetigkeiten an den Randbereichen des Signalausschnitts wird das Signalfenster hier nicht mit einem rechteckigen Fenster ausgeschnitten, sondern mit einem Fenster, das zum Rand hin langsam gegen

Null geht. Hierzu können Hamming-, Hanning-, Gauß- oder ähnliche Fensterfunktionen benutzt werden.

Untersuchung des Cepstrums:

Periodizitäten im Zeitbereich können auch im Cepstralbereich detektiert werden (siehe Kapitel 3.5). Nach dem source-filter-Modell der Spracherzeugung befinden sich hier Maxima bei Vielfachen der Grundperiodendauer T_0, genau wie bei den im letzten Abschnitt betrachteten Korrelationsverfahren. Auch bei der Berechnung der Autokorrelation und des Cepstrums gibt es Gemeinsamkeiten (siehe folgende Abbildung):

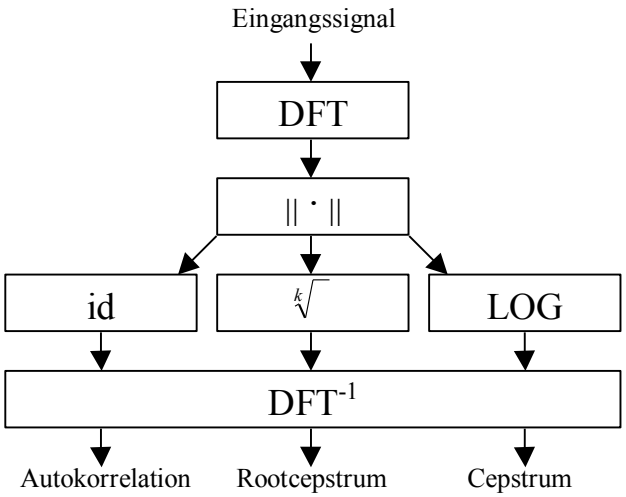

Abbildung 3.6 Berechnung von Autokorrelation und (Root-)Cepstrum

Eine Abwandlung des Cepstrums ist das Rootcepstrum, bei dem die Berechnung des Logarithmus durch die *k*-te Wurzel ersetzt wird. Für *k*=1 ergibt sich gerade die Autokorrelationsfunktion.

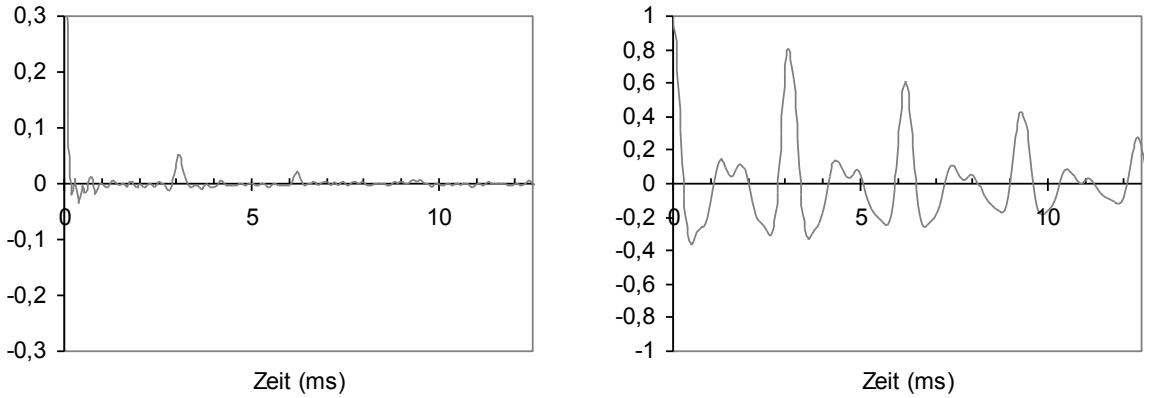

Abbildung 3.7 Cepstrum und Rootcepstrum

Testergebnisse bei Verwendung dieser verschiedenen Funktionen im Zusammenhang mit dem von uns implementierten Grundfrequenzverfahren finden sich in Kapitel 4.3 in Tabelle 4.4.

Untersuchung des Spektrums

DPF0-SEQ ist ein Vertreter der Kurzzeitverfahren, bei denen die F_0-Bestimmung auf der Untersuchung des Spektrums basiert. Das Verfahren ist eine weiter entwickelte Variante des DPF0 Algorithmus, der in [Kom89] und [Kie92] beschrieben wurde. Der DPF0-SEQ wurde im Rahmen des Verbmobil Projektes entwickelt, und ist in das Verbmobil Prosodiemodul integriert. Das Verfahren beruht auf der Annahme, daß das absolute Maximum im Kurzzeitspektrum eines stimmhaften Signalausschnittes ein ganzzahliges Vielfaches der Grundfrequenz F_0 ist. Auf diesen Algorithmus wird etwas genauer eingegangen, da sich dieser Algorithmus gegenüber vielen anderen als überlegen erwiesen hat, und wir ihn mit dem von uns implementierten Algorithmus auf den gleichen Sprachdaten (SPONTAN) vergleichen können.

Der Vorverarbeitungsschritt des Algorithmus teilt das Signal in äquidistante Zeitfenster (Frames) der Länge 10 ms, und führt für jeden Frame eine stimmhaft/stimmlos Entscheidung durch. Benachbarte stimmhafte Frames werden in eine stimmhafte Region gruppiert. Die weiteren Verarbeitungsschritte bearbeiten nur noch diese stimmhaften Regionen.
Die Sprachsignale in den stimmhaften Regionen werden mit einer Grenzfrequenz von 1100 Hz tiefpaßgefiltert und zum Zwecke der Datenreduktion neu abgetastet. Für jeden stimmhaften Frame wird ein Kurzzeitspektrum berechnet. Dabei ist wie bei den Korrelationsverfahren zu beachten, daß die Fenstergröße mindestens doppelt so groß sein muß, wie die maximal erwartete Grundperiodendauer. Im Kurzzeitspektrum wird dann in jedem stimmhaften Frame k das absolute Maximum bei $f_{\max}(k)$ gesucht, diese Frequenz ist gemäß der obigen Annahme ein Vielfaches der gesuchten Grundfrequenz F_0.
Anschließend wird in jeder stimmhaften Region ein sogenannter Zielwert für die Grundfrequenz bestimmt. Dieser F_0-Zielwert geht in der sich anschließenden Bestimmung des F_0-Verlaufs in eine Kostenfunktion ein. Die Berechnung des Zielwertes wird in einem einzelnen Frame der stimmhaften Region mit einem Standard-Algorithmus (ADMF [Ros74] und Seneff [Sen78]) durchgeführt. Da die Berechnung möglichst zuverlässig sein soll, wählt man heuristisch einen Frame mit besonders hoher Energie aus dem Zentrum der stimmhaften Region aus.
Nach diesen Schritten sind jetzt also die stimmhaften Regionen zusammen mit einem F_0-Zielwert $F0_l$ für jede Region l bekannt, außerdem wurde für jeden stimmhaften Frame k ein Vielfaches $f_{\max}(k)$ der Grundfrequenz berechnet. Auf Grundlage dieser Werte wird nun in jedem Frame k der stimmhaften Region l eine Menge von bis zu 5 F_0-Hypothesen $F_{k,l,a}$ aufgestellt:

$$F_{k,l,a} = \frac{f_{\max}(k)}{\left\lfloor \frac{f_{\max}(k)}{F0_l} \right\rfloor + a} \text{ , mit } a \in \{-2,1,0,1,2\} \text{ und } \left\lfloor \frac{f_{\max}(k)}{F0_l} \right\rfloor + a > 0 \tag{3.22}$$

Das aus dem Spektrum bestimmte F_0-Vielfache $f_{\max}(k)$ ist also ein Vielfaches jeder dieser Hypothesen. Das Ergebnis dieses Schrittes ist die Matrix von F_0-Hypothesen $F_{k,l,a}$.

Aufgabe des nächsten Schrittes ist es, aus der F_0-Hypothesenmatrix $F_{k,l,a}$ genau eine Hypothese pro Frame k auszuwählen. Zu diesem Zweck wird eine Kostenfunktion für alle möglichen Pfade durch die Hypothesenmatrix $F_{k,l,a}$ definiert. Unter allen Pfaden wird derjenige mit den geringsten Kosten ausgewählt. Das Kostenkriterium basiert darauf, daß der F_0-Zielwert in jeder Region zuverlässig geschätzt werden konnte, und daß sich die Grundfrequenz während des Sprechvorgangs nur langsam verändert, und damit die Differenz aufeinanderfolgender F_0-Werte gering ist. Die Kosten des Pfades setzen sich also aus dem Unterschied der F_0-Werte zweier aufeinanderfolgender Frames zusammen, und aus dem Abstand des F_0-Wertes vom Zielwert der stimmhaften Region. Dabei ist die Kostenfunktion so gestaltet, daß sie mit dem Verfahren der dynamischen Programmierung [Bel57] für alle möglichen Pfade effizient ausgewertet werden kann.

Der DPF0-SEQ Algorithmus wurde unter anderem auf dem SPONTAN-Korpus evaluiert, ein Vergleich der Ergebnisse mit anderen Algorithmen ist in Kapitel 4.12 zu finden. Eine genauere Beschreibung des Verfahrens findet sich in [Nie94]. Eine iterative Variante (DPF0-ITER) ist in [Zot95] beschrieben.

4 Unser Algorithmus zur Grundfrequenzverfolgung

Die Aufgabe bestand darin, einen Algorithmus zur Bestimmung des Grundfrequenzverlaufs in Sprachsignalen zu entwickeln, der möglichst problemlos in verschiedene Anwendungen integriert werden kann. Vorgaben waren sowohl Sprecherunabhängigkeit, als auch Robustheit gegenüber anwendungsspezifischen Gegebenheiten, wie zum Beispiel Aufnahmerauschen und verwendetem Mikrofon. Zur Evaluierung unseres Algorithmus stand uns die SPONTAN-Stichprobe (siehe Kapitel 7.1) zur Verfügung, die aus insgesamt 28 Minuten Sprachdaten besteht. Diese Sprachdaten sind alle 12.8 ms als stimmhaft/stimmlos klassifiziert, und mit der Grundfrequenz zu jedem Zeitpunkt etikettiert. Die Etikettierung wurde mit automatischen Verfahren erzeugt und anschließend von erfahrenen Phonetikern gehörsadäquat korrigiert.
Der von uns entwickelte Algorithmus, fällt in die Klasse der Kurzzeitverfahren, da er keine Grundperiodengrenzen berechnet, sondern das Sprachsignal in Zeitfenster im Abstand von 10 ms einteilt, und für jedes Fenster einen Schätzwert für die vorliegende Grundfrequenz liefert. Im Gegensatz zu den meisten Verfahren zur Bestimmung des Grundfrequenzverlaufs ist unser Algorithmus nicht auf eine vorhergehende Bestimmung der stimmhaften Abschnitte im Sprachsignal angewiesen. Wir wollten eine harte SH/SL-Entscheidung (stimmhaft/ stimmlos) vor der F_0-Bestimmung vermeiden, da Klassifikatoren für diese Entscheidung relativ unzuverlässig sind, sie klassifizieren typischerweise etwa 10% aller Frames fehlerhaft (siehe Tabelle 4.12). Ein weiterer Unterschied zu den meisten Kurzzeitverfahren ist, daß wir nicht in jedem Frame einen oder nur ein paar wenige F_0-Kandidaten selektieren, unter denen der korrekte F_0-Wert möglicherweise gar nicht enthalten ist. Unser Verfahren verlagert die F_0-Entscheidung für einen einzelnen Frame vollständig in die Bewertung einer äußerungsglobalen Kostenfunktion.

4.1 Verwendete Fehlermaße

Zum Vergleich unseres F_0-Algorithmus mit anderen Verfahren konnten wir die Abweichung zwischen der Ausgabe unseres Algorithmus und den etikettierten SPONTAN-Sprachdaten bestimmen. Damit liegt uns für jeden Frame, das heißt alle 12.8 ms, ein Referenzwert für die Grundfrequenz vor. Zum Vergleich mit dieser Referenz wird durch den zu bewertenden Algorithmus für jeden Frame eine Grundfrequenzhypothese berechnet.
Ein wichtiger Aspekt beim Vergleich von F_0-Verfahren ist die Art der Referenzetikettierung. So macht es einen Unterschied, ob die Referenz aus gehörsadäquat korrigierten F_0-Schätzwerten eines framebasiert arbeitenden Verfahrens besteht (wie beim SPONTAN-Korpus), oder aus manuell korrigierten Grundperioden eines perioden-

synchronen Verfahrens [Kie96]. Vom Standpunkt der Tonhöhenperzeption aus sind gehörsadäquat korrigierte Referenzen den periodensynchronen Etikettierungen vorzuziehen, da nur im ersten Fall kurze laryngalisierte Segmente mit der vom Gehör wahrgenommenen Tonhöhe etikettiert sind (siehe Kapitel 3.1).
In [Kom96] wird als Maß für die Abweichung von F_0-Referenz und -Hypothese die sogenannte Grobfehlerrate vorgeschlagen. Sie ist definiert als Quotient aus der Anzahl der Frames, deren Referenzgrundfrequenz von der berechneten Grundfrequenzhypothese um mehr als 30 Hz abweicht, und der Anzahl der Frames, die sowohl in der Referenz, als auch vom zu bewertenden Algorithmus als stimmhaft markiert wurden ([Kom96] S. 191,193). Dieses Maß ist für sich allein genommen allerdings nur wenig aussagekräftig, da ein Algorithmus, der eine sehr harte SH/SL-Entscheidung trifft, also eine die nur energiereiche harmonische Silbenkerne als stimmhaft, und den Rest als stimmlos klassifiziert, im Allgemeinen eine sehr niedrige Grobfehlerrate liefert. Der Grund dafür ist, daß so Übergangsbereiche zwischen Phonemen und Laryngalisierungen als stimmlos klassifiziert werden, also gerade die Bereiche, in denen die F_0-Bestimmung generell besonders fehleranfällig ist [Kie96]. Deshalb sollte, wie im Folgenden, beim Vergleich von F0-Verfahren über die Grobfehlerrate die SH/SL-Fehlerrate auch angegeben werden.
Vom Standpunkt der Tonhöhenperzeption aus ist die Definition der Grobfehlerrate über die Frequenzdifferenz von 30 Hz unserer Meinung nach allerdings nicht geeignet. Die Frequenzauflösungsfähigkeit des menschlichen Gehörs nimmt bei höheren Frequenzen immer mehr ab. Das heißt, daß eine Änderung von 30 Hz in der Grundfrequenz eines tief sprechenden Mannes - beispielsweise von 80 auf 50 Hz - vom Gehör als viel größer wahrgenommen wird, als eine Änderung von 30 Hz in der Grundfrequenz einer hoch sprechenden Frau - beispielsweise von 530 auf 500 Hz. Da die wahrgenommene Frequenz in etwa der logarithmierten tatsächlichen Frequenz entspricht, bietet sich als gehörsadäquate Differenz zwischen der Grundfrequenzreferenz f_r und der Hypothese f_h der prozentuale Unterschied zwischen beiden Frequenzen an [Har94]:

$$d_{Harbeck}(f_r, f_h) = \frac{|f_r - f_h|}{f_r} \tag{4.1}$$

In unserem Algorithmus wird als gehörsadäquates Abstandsmaß zwischen zwei Frequenzen f_1 und f_2 die folgende Funktion benutzt:

$$d_{Oktave}(f_1, f_2) = \left| \mathrm{ld}\left(\frac{f_1}{f_2} \right) \right| \tag{4.2}$$

Diese Funktion liefert den Abstand zwischen den beiden Frequenzen in Oktaven:

$d_{Oktave}(f, f) = 0$
$d_{Oktave}(f, 2f) = 1$
$d_{Oktave}(f, 4f) = 2$

Da aber in [Kie96] für mehrere F_0-Verfahren die Grobfehlerrate auf Basis eines Abstands von 30 Hz zusammen mit der SH/SL-Fehlerrate angegeben sind, haben auch wir dieses Maß in der Bewertung unseres Algorithmus verwendet. Dies ermöglichte uns den Vergleich dieser Verfahren mit unserem Algorithmus auf dem SPONTAN-Korpus.

4.2 Vorverarbeitung

Eingabe für unseren Algorithmus ist das abgetastete und quantisierte Sprachsignal. Typischerweise liegt die verwendete Abtastrate f_{abtast} im JANUS-System bei 16 kHz, im SPONTAN-Korpus liegt sie bei 10 kHz. Auf diesem Signal wird zunächst eine Tiefpaßfilterung durchgeführt, um hochfrequente Vokaltrakteinflüsse und Störsignale über 1100kHz zu unterdrücken. Die Grundlagen zu digitalen Filtern sind beispielsweise in [Rab78] dargestellt, im Folgenden wird nicht näher auf die Grundlagen eingegangen werden. Der optimale lineare Tiefpaßfilter hat als Übertragungsfunktion $F(z)$ im Frequenzbereich eine Rechteckfunktion mit $F(z)=0$ für $z>f_{\text{grenz}}$. Seine Impulsantwort $f(t)$ im Zeitbereich ist durch eine sinc-Funktion gegeben. Deren Nachteil ist, daß sie für wachsendes t nur langsam gegen 0 geht, so daß zur Approximation dieses Filters eine sehr große Filtermaske im Zeitbereich notwendig ist. Das andere Extrem eines linearen Tiefpaßfilters ist ein Filter, dessen Impulsantwort im Zeitbereich durch eine Rechteckfunktion gegeben ist. Die Übertragungsfunktion dieses Filters ist eine sinc-Funktion, was wiederum bedeutet, daß der Filter höhere Frequenzen nicht monoton stärker dämpft als tiefere.
In unseren Experimenten wurde ein Tiefpaßfilter benutzt, der einen Kompromiß zwischen der Lokalisierung des Filters im Zeitbereich (=effiziente Berechnung) und Frequenzbereich (=gute Tiefpaßfilterung) bietet: ein Tiefpaß mit der Gaußfunktion sowohl als Impulsantwort als auch Übertragungsfunktion.
Die Versuche in Kapitel 4.6 zeigen, daß die Tiefpaßfilterung mit einer Rechteckmaske praktisch die gleichen Grobfehlerraten liefert, wie die Filterung mit der Gaußmaske. Der Vorteil der Rechteckmaske ist, daß die Filterung (=Mittelwertbildung) durch einen rekursiven Filter besonders effizient implementiert werden kann. Dessen Rechenaufwand ist unabhängig von der Filtermaskenbreite, während der Aufwand für die Faltung mit der Gaußmaske linear mit der Maskenbreite wächst.
Im Anschluß an die Tiefpaßfilterung mit Grenzfrequenz ω kann nach dem Abtasttheorem eine Neuabtastung des Signals mit einer Abtastfrequenz von mindestens 2ω durchführt werden, um die Laufzeit nachfolgender Verarbeitungsstufen des Algorithmus zu verringern. In allen Experimenten bis zum Kapitel 4.6 wurde auf eine Neuabtastung verzichtet, da diese Experimente auch ohne Neuabtastung schnell genug durchgeführt werden konnten.
Falls die vorliegenden Sprachsignale nicht mittelwertfrei sind, kann optional auch eine Mittelwertbefreiung durchgeführt werden, beispielsweise auch in Verbindung mit einer Hochpaßfilterung, wobei die Grenzfrequenz kleiner sein muß, als die mi-

nimale erwartete Grundfrequenz - etwa 35-50 Hz. Dieser Schritt war auf dem SPONTAN-Korpus allerdings nicht notwendig.

4.3 Kurzzeitanalyse

Wie bei Kurzzeitverfahren üblich, wird das tiefpaßgefilterte Signal in äquidistante Kurzzeitfenster (Frames) eingeteilt. Der Abstand der Zeitfenster beträgt bei der Evaluation auf dem SPONTAN-Korpus 12.8 ms, bei der Anwendung des Verfahrens im Rahmen des JANUS-Systems typischerweise 10 ms. Für die Detektion von Periodizitäten im Signal muß die Länge eines Zeitfensters so groß gewählt werden, daß es mindestens zwei Grundperioden enthält. Bei einer kleinsten erwarteten Grundfrequenz von typischerweise 50 Hz muß das Fenster also mindestens $2 \cdot 1/(50\ \text{Hz}) = 40$ ms groß sein. Das Fenster weiter zu vergrößern ist nicht sinnvoll, da dadurch einerseits der Rechenaufwand für die Kurzzeitanalyse vergrößert wird, und andererseits auch die Ähnlichkeit von zeitlich weiter auseinanderliegenden Grundperioden aufgrund des sich verändernden Vokaltraktes geringer ist.
Für die Analyse der Signalperiodizität im Kurzzeitfenster stehen uns die in Kapitel 3.5.2 erwähnten Zeitbereichs-, Spektrum- und Cepstrummethoden zur Verfügung. In einem ersten Versuch zum Vergleich dieser Kurzzeitverfahren wurde die Grobfehlerrate auf dem SPONTAN-Korpus untersucht, wobei der F_0-Detektor die F_0-Hypothese nur auf Grundlage der Betrachtung eines einzelnen Zeitfensters bestimmt. Beispielsweise wird die wahrscheinlichste T_0-Hypothese bei Verwendung der normierten Kreuzkorrelation durch Suche des Maximums in der Korrelationsfunktion bestimmt. Die Selektionskriterien für die anderen Kurzzeitverfahren wurden bereits in Kapitel 3.5.2 vorgestellt. Problematisch ist die Festlegung eines F_0-Auswahlkriteriums bei den Autokorrelations- und Cepstrumverfahren. Diese weisen für eine T_0-periodische Funktion zwar lokale Maxima bei $t = k \cdot T_0$, $1 \le k < \infty$ auf, nehmen ihr globales Maximum aber stets bei $t=0$ an. Deshalb wurde bei diesen Verfahren vor der Maximumauswahl das Maximum bei $t=0$ entfernt, indem das Ergebnis f der Kurzzeitanalyse auf dem Intervall $[0,t_1]$ gleich Null gesetzt wurde, wobei t_1 der erste Nulldurchgang ist: $t_1 = \min \{ t \mid t \in \mathbb{N},\ f(t) \le 0 \}$.
Dieser Schritt beeinflußt bei einem mittelwertfreien T_0-periodischen Signal das Maximum bei $t=T_0$ nicht, da sich in der zugehörigen Autokorrelationsfunktion negative Minima mit positiven Maxima bei $k \cdot T_0$ abwechseln. Das Ergebnis dieses Schrittes ist für das Beispiel der Autokorrelationsfunktion in der folgenden Abbildung dargestellt: bei der modifizierten Autokorrelationsfunktion im rechten Bild liegt das globale Maximum bei T_0.

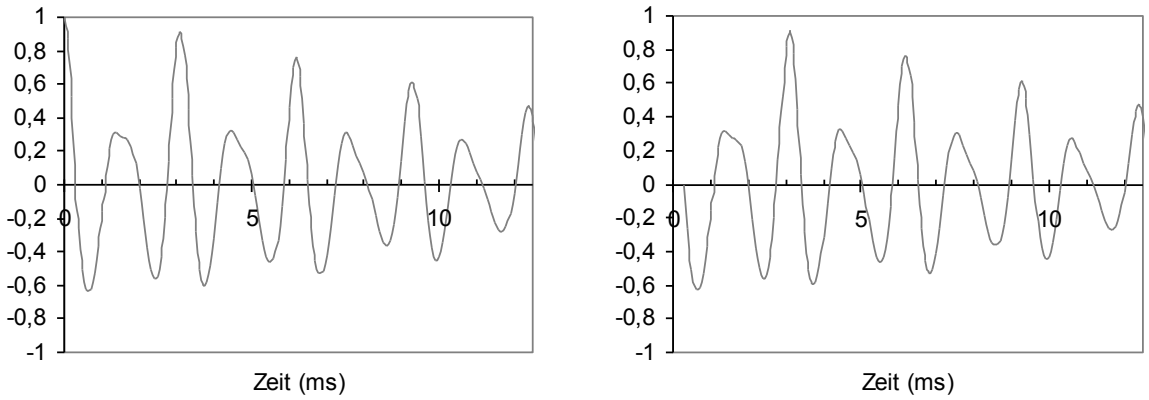

Abbildung 4.1 Entfernung des globalen Korrelationsmaximums bei t=0

Die Grobfehlerraten auf dem SPONTAN-Korpus sind für alle getesteten Kurzzeitverfahren in der folgenden Tabelle eingetragen. Da wir keine SH/SL-Entscheidung vor der Detektion des F0-Verlaufs treffen wollten, gehen alle in der Referenz als stimmhaft etikettierten Frames in die Fehlerrate ein. Man beachte: der Detektor benutzt zur Bestimmung der Grundfrequenz in einem Frame nur lokale Informationen aus diesem einzelnen Kurzzeitfenster.

Verfahren	*Grobfehlerrate*	*Bemerkung*
Cepstrum	32%	Gaußfenster
Norm. Kreuzkorrelation	31%	Fenster aus Gleichung 3.7
Energiespektrum	27%	
Norm. Kreuzkorrelation	19%	Fenster aus Gleichung 3.14
Autokorrelation	7.7%	Gaußfenster, über DFT
Autokorrelation	7.6%	Rechteckfenster
Rootcepstrum	7.6%	Gaußfenster,Quadratwurzel

Tabelle 4.1 Vergleich der Grobfehlerraten verschiedener Kurzzeitverfahren

Die besten Ergebnisse liefern die Autokorrelationsverfahren, und das Rootcepstrum. Wir halten das Energiespektrum im Zusammenhang mit der Grundfrequenzbestimmung für keine geeignete Analysemethode. Der Grund dafür ist die konstante Frequenzauflösung im Spektrum: wie bereits am Anfang dieses Kapitels erwähnt wurde, muß die Fenstergröße mindestens doppelt so groß sein, wie die maximal erwartete Grundperiodendauer. Das Fenster sollte aber nicht zu groß gewählt werden, da sich der Vokaltrakt und auch die Grundfrequenz langsam verändern. Wählt man beispielsweise die minimal zu detektierende Grundfrequenz zu 35 Hz und die Kurzzeitfensterlänge so, daß 2 Grundperioden dieser Frequenz ins Fenster passen, ergibt sich eine Frequenzauflösung von nur 35/2=17.5 Hz, während das menschliche Gehör in diesen unteren Frequenzbereichen gerade seine höchste Frequenzauflösung erreicht. Bei einem tief sprechenden Mann mit einer Grundfrequenz von 70 Hz entspräche diese Auflösung von 17.5 Hz aber bereits einer viertel Oktave. Kleinere, möglicherweise informationstragende F_0-Veränderungen wären mit dem spektralen

Verfahren überhaupt nicht detektierbar. Im Gegensatz dazu ist die variable Frequenzauflösung der Zeitverfahren (Korrelation und Cepstrum) dem menschlichen Gehör eher angemessen. Mit diesen Verfahren wird eine hohe Frequenzauflösung in niedrigen Frequenzbereichen erreicht, und eine geringere in höheren. Aus diesem Grund wurden die folgenden Experimente nur noch mit Korrelationsfunktionen und cepstralen Verfahren durchgeführt.

4.4 Einbeziehung der Information aus Nachbarframes

Mit dem einfachen F_0-Detektor aus dem letzten Kapitel wurde eine relativ hohe Grobfehlerrate von etwa 8% bis 30% erzielt. Dieser Detektor benutzt allerdings zur F_0-Bestimmung in einem Frame nur lokale Informationen aus diesem einzelnen Kurzzeitfenster. Die folgende Abbildung zeigt zwei Fehlerhistogramme, die Hinweise auf die vom Kreuzkorrelation-basierten Detektor gemachten Fehler liefern sollen. Diese wurden erstellt, indem auf allen stimmhaften Referenzframes die Abweichung der F_0-Hypothese f_h von der F_0-Referenz f_r bestimmt wurde. Das linke Histogramm zeigt die Wahrscheinlichkeitsdichte der Differenz $d_{hertz}(f_h,f_r)=f_h-f_r$ in Hertz, das rechte Bild zeigt die Dichte der Differenz $d_{oktave}(f_h,f_r)=\log(f_h/f_r)$ in Oktaven statt Hertz.

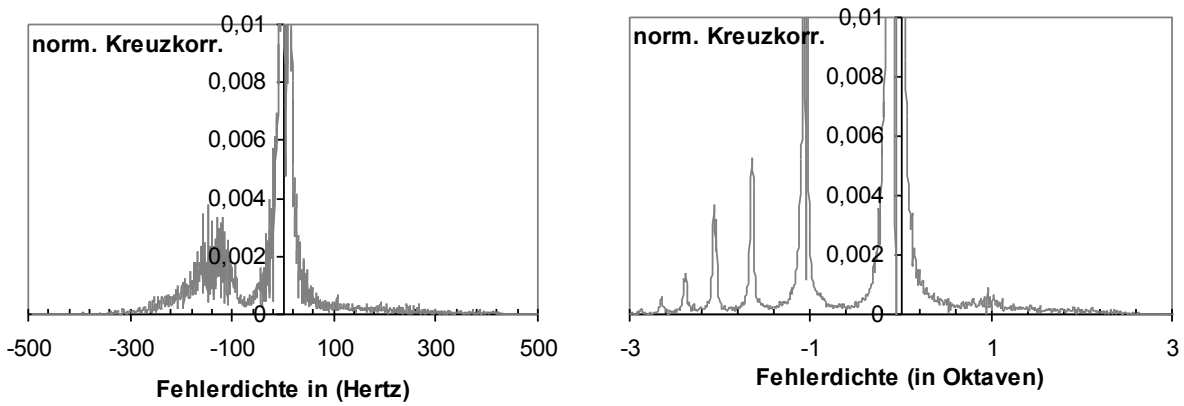

Abbildung 4.2 Fehlerdichten des einfachen F_0-Detektors (Kreuzkorrelation)

Wie man bereits aus der Grobfehlerrate von etwa 31% schließen kann, liegt die Differenz d_{hertz} bei der Mehrzahl der Frames in der Nähe des Nullpunktes. Im rechten Histogramm von d_{oktave} fallen zusätzlich lokale Nebenmaxima in der Fehlerdichte bei $-\log_2(2)$, $-\log_2(3)$, $-\log_2(4)$ und so weiter auf. Diese sind Hinweise darauf, daß ein großer Teil der Grobfehler des F_0-Detektors auf die Auswahl von Maxima der Kreuzkorrelationsfunktion bei $k\cdot T_0$ ($k\geq 2$, $k\in\mathbb{N}$) zurückzuführen ist. Diese lokalen Maxima wurden bereits in Kapitel 3.5.2 damit begründet, daß ein Signal mit Periode T auch periodisch mit Periode $k\cdot T$ ($k\geq 2$, $k\in\mathbb{N}$) ist, und deswegen in den Korrelationsfunktionen und im Cepstrum Maxima bei Vielfachen der Grundperiodendauer auftauchen. Bei einem Teil der Frames war also die Periodizität mit $k\cdot T$ ($k\geq 2$) stärker ausgeprägt, als die Periodizität mit der tatsächlichen Grundperiode, das ist insbesondere bei Laryngalisierungen wie etwa Oktavsprüngen (siehe Kapitel 3.1) der Fall.

Außerdem ist rechts vom Ursprung, wenn auch in viel geringerem Maße, eine Häufung der Fehler bei $2F_0$ (statt T_0) zu sehen. Diese sind zum Teil auf solche Fälle zurückzuführen, in denen die Frequenz der ersten Formante gerade doppelt so hoch ist wie F_0.
Die nächste Abbildung stellt die Fehlerverteilungen auch für alle anderen untersuchten Kurzzeitanalyseverfahren dar. Diesmal werden nur die Dichten von d_{oktave} gezeigt, da diese Verteilung aufschlußreicher ist als d_{hertz}.

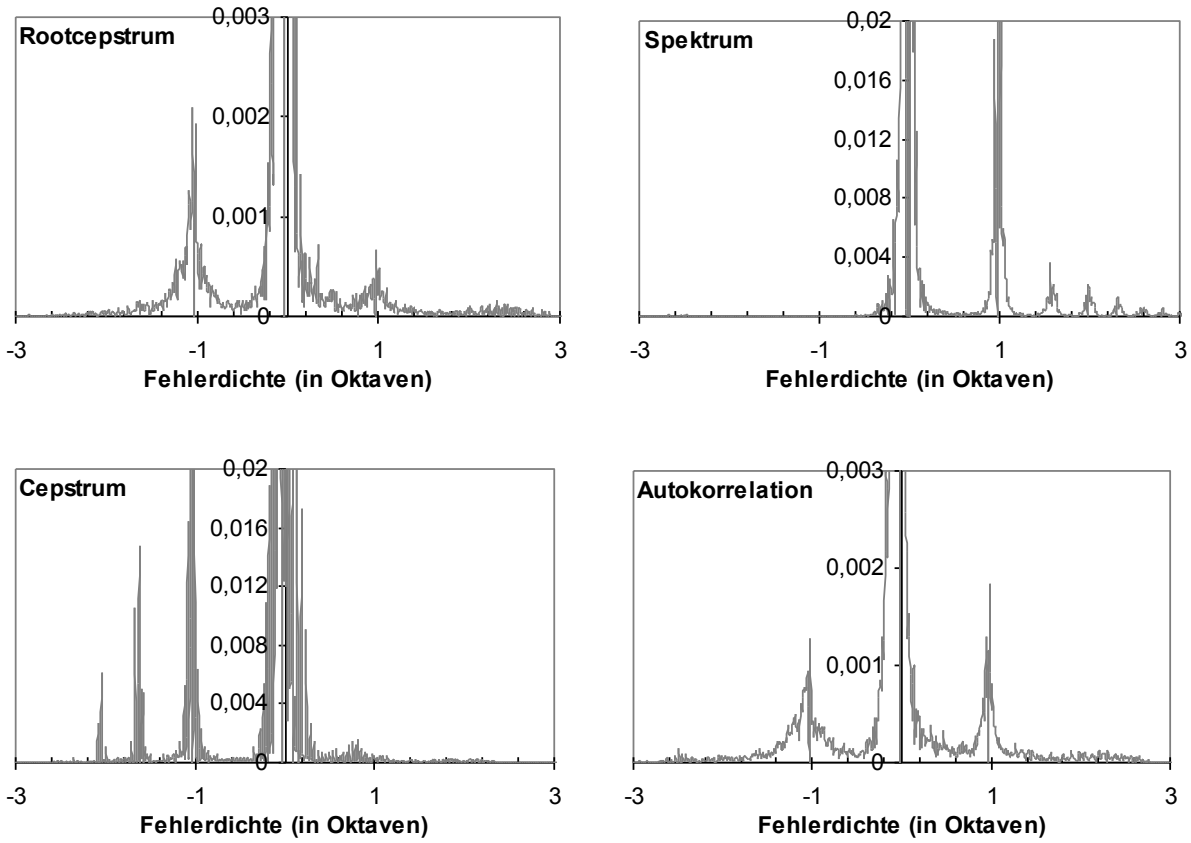

Abbildung 4.3 Fehlerdichten

Für die Fehlerverteilung für die Cepstrumanalyse gilt das gleiche wie für die normierte Kreuzkorrelation, auch hier werden bei Vielfachen von T_0 größere Peaks festgestellt, als bei T_0 selbst. Die Fehlerverteilungen von Rootcepstrum und Autokorrelation weisen bemerkenswert wenige dieser Detektionen von T_0-Vielfachen auf. Bei der spektralen Analyse gilt der umgekehrte Sachverhalt: hier werden oft ganzzahlige Vielfache von F_0 - statt T_0 - im Spektrum detektiert.

Die fehlerhaft detektierten Frames machen nur einen Bruchteil (8 bis 30%) der gesamten Framezahl aus. Der einfache F_0-Detektor aus dem letzten Kapitel wird dadurch erweitert, daß die F_0-Auswahl nicht mehr nur von der Betrachtung eines einzelnen Kurzzeitfensters abhängig gemacht wird, sondern auch Informationen aus benachbarten Frames bei der Auswahl berücksichtigt werden. Dazu wird nutzen die Tatsache ausgenutzt, daß der F_0-Verlauf in Sprachsignalen relativ glatt ist. Das heißt $\partial F_0(t)/\partial t$ - beziehungsweise die Differenz der F_0-Werte in aufeinanderfolgenden

Frames geteilt durch die Zeitdauer zwischen zwei Meßpunkten als dessen diskrete Approximation - ist klein. Die untere Abbildung zeigt die auf dem SPONTAN-Korpus bestimmte Verteilung des Abstands zweier aufeinanderfolgender Referenz-F_0-Werte (Frameabstand 12.8 ms). Dabei wurde - wie oben bei den Fehlerdichten- links der Abstand dieser beiden Werte in Hertz, rechts in Oktaven gemessen (Gleichungen 4.1 und 4.2). Die durchgezogenen grauen Linien in den Histogrammen zeigen diese Verteilung für alle männlichen Sprecher, die gestrichelten schwarzen für die weiblichen.

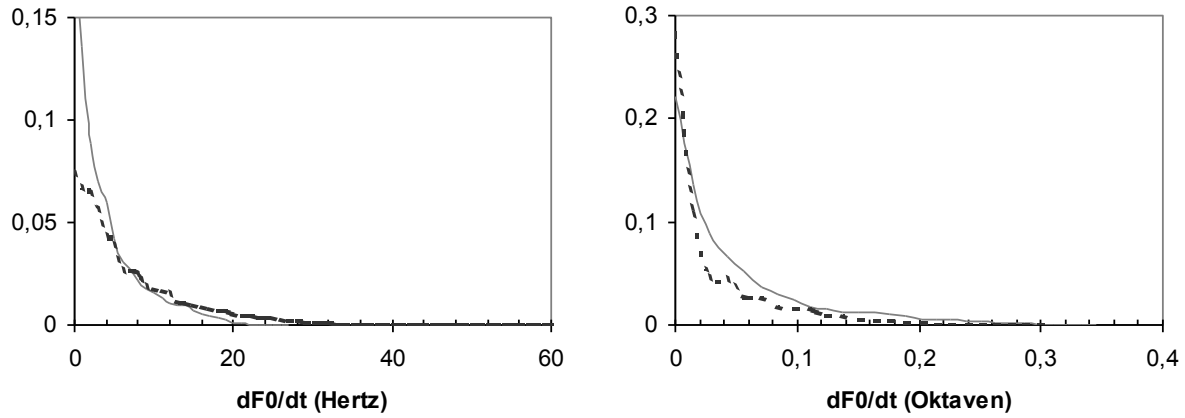

Abbildung 4.4 Verteilungsdichte der F_0-Differenz aufeinanderfolgender Frames

Im linken Histogramm ist zu erkennen, daß die kurzzeitige Grundfrequenzänderung zwischen zwei Frames bei Männern nur wenige Hertz beträgt (< 22 Hz) und bei Frauen größer ist (< 60 Hz). Im rechten Histogramm sehen wir, daß der Abstand aufeinanderfolgender F_0-Werte in Oktaven bei beiden Geschlechtern nie größer als 0.4 Oktaven ist. Außerdem sind die Verteilungen der Abstände im rechten Histogramm für beide Geschlechter ähnlicher als links. Mit der Messung der Differenz in Oktaven erhält man also einen Wert der eher geschlechtneutral, und damit sprecherunabhängiger ist. Bei der Bewertung der Histogramme muß allerdings im Auge behalten werden, daß diese nicht besonders repräsentativ sind, da sie auf der SPONTAN-Stichprobe erstellt wurden, und demzufolge nur auf den Daten von 3 weiblichen und einem einzigen männlichen Sprecher beruhen.
Die von unserem bisherigen Algorithmus berechneten Grundfrequenzverläufe enthalten laut den Fehlerhistogrammen in den Abbildungen 4.2 und 4.3 Grundfrequenzverläufe mit größeren Frequenzsprüngen als diesen 0.4 Oktaven. Die Anzahl dieser Grobfehler wollten wir durch Einführen einer Glattheitsbedingung verringern. Der bisherige Algorithmus berechnet die Grundperiodendauer durch Suche des globalen Maximums auf jedem Kurzzeitfenster. Diese Berechnung können äquivalent als Pfadsuche in der aus den Kurzzeitanalysen für jeden Frame gebildeten Matrix $M_{i,j}$ ausgedrückt werden: in Spalte $j \in \{0,1,..,j_{max}\}$ der Matrix steht das Ergebnis der Kurzzeitanalyse für Frame j, in Zeile $i \in \{0,1,..,i_{max}\}$ steht der i-te Korrelations- beziehungsweise Cepstrumkoeffizient für jeden Frame. Die Spaltenanzahl $j_{max}+1$ entspricht der Anzahl der Frames in dem betrachteten Sprachsignal, die Zeilenzahl i-

$_{max}$+1 ist die Anzahl der berechneten Kurzzeitanalysekoeffizienten pro Frame. Im Falle der Kurzzeitanalyse durch Berechnung des Spektrums entspricht diese Matrix gerade einem Spektrogramm, bei einer Grundfrequenzanalyse per Korrelation oder Cepstrum wird analog ein Korrelogramm beziehungsweise Cepstrogramm berechnet. Die folgende Abbildung zeigt ein Beispiel für ein solches Cepstrogramm, bei dem die Kurzzeitanalyse mit dem normiertem Rootcepstrum (Quadratwurzel) durchgeführt wurde. Im rechten Bild ist das gleiche Cepstrogramm (zur Kontrasterhöhung lediglich etwas heller) mit dem zugehörigen Referenz T_0-Verlauf dargestellt. Zu beachten ist in diesen Bildern, daß die dargestellte Matrix an der Waagerechten gespiegelt wurde, $M_{0,0}$ befindet sich also in der linken unteren Ecke der Abbildung.

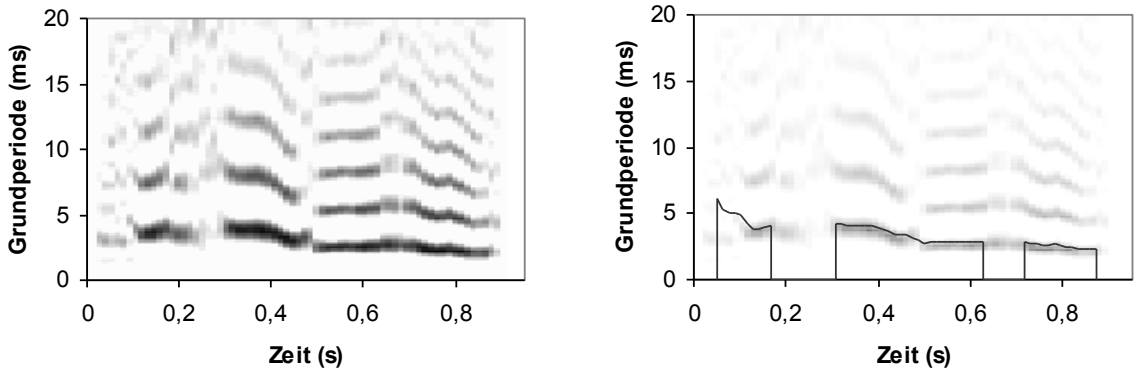

Abbildung 4.5 Cepstrogramme mit Referenz-T_0-Verlauf

Hier ist zu sehen, daß der T_0-Pfad im Cepstrogramm innerhalb der meisten Frames (= Spalte im Cepstrogramm) durch das globale Maximum in der Kurzzeitanalysefunktion in diesem Frame verläuft.
Die Maximumbestimmung auf jedem Frame, die der einfache F_0-Detektor aus Kapitel 4.3 durchführt, ist äquivalent zu einer Bestimmung desjenigen Grundperiodendauerverlaufs $\vec{t}$ in dieser Matrix, der die Gewinnfunktion

$$G(\vec{t}) = \sum_{j=0}^{j_{\max}} M_{t_j,j} \tag{4.3}$$

maximiert. Dabei ist das j-te Element t_j im Ausgabevektor $\vec{t}$ die Hypothese des F_0-Detektors für die Grundperiodendauer in Frame $j \in \{0,1,..,j_{\max}\}$. Diese Gewinnfunktion ist genau dann maximal, wenn in jeder Spalte (Frame) der Matrix der Maximalwert ausgewählt wird, genauso wie der einfache F_0-Detektor in Kapitel 4.3 arbeitete. Die Pfadauswahl erfolgte also mit Hilfe des folgenden Kriteriums:

$$\vec{t} = \underset{\vec{t}' \in P}{\operatorname{argmax}}\, G(\vec{t}') \tag{4.4}$$

Dabei ist $P=\{1,2,...,i_{\max}\}^{j_{\max}+1}$ die Menge aller möglichen Verläufe der Grundperiodendauer im gegebenen Cepstrogramm $M_{i,j}$. In der folgenden Abbildung ist der per normierter Kreuzkorrelation berechnete Pfad in das Cepstrogramm eingetragen. Die dazu gehörige Äußerung ist die gleiche wie in Abbildung 4.5.

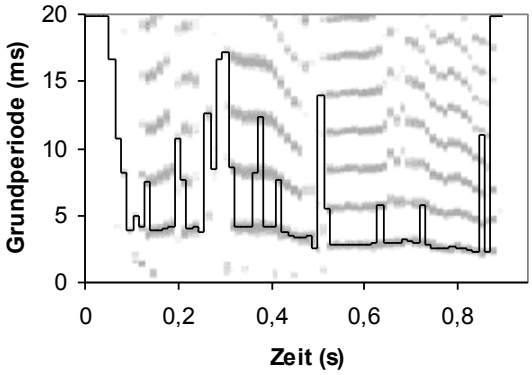

Abbildung 4.6 T0-Pfad nach Gleichung 4.4 im Cepstrogramm (Kreuzkorrelation)

Hier ist zu sehen, daß der Verlauf der Grundperiodendauer einige Oktavsprünge aufweist, die in sprachlichen Äußerungen laut dem Histogramm in Abbildung 4.4 nicht vorkommen. Um das Vorwissen über die möglichen Pfadverläufe in den F_0-Detektor einfließen zu lassen – daß nämlich der Abstand der Grundfrequenzwerte in zwei aufeinanderfolgenden Frames j und j+1 stets unterhalb einer bestimmten Schranke bleibt - kann die Menge der zu testenden Pfade für das Auswahlkriterium aus Gleichung 4.4 beschränkt werden:

$$\vec{t} = \underset{\vec{t}' \in P}{\operatorname{argmax}}\, G(\vec{t}')$$

$$\text{mit } P = \left\{ \vec{t}' \in \{0,1,\ldots,i_{\max}\}^{j_{\max}+1} \mid d(t_j, t_{j+1}) < d_{\max},\ j \in \{0,1,\ldots,j_{\max}-1\} \right\} \qquad (4.5)$$

Mit dieser Beschränkung der Menge der möglichen T_0-Verläufe P ergibt sich im Vergleich zum Pfad in Abbildung 4.6 jetzt mit dem Parameter $d_{\max}$=0.4 der folgende Pfad:

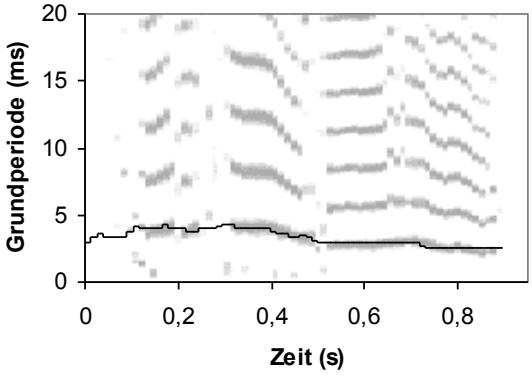

Abbildung 4.7 T_0-Pfad nach Einschränkung der Menge mögliche Pfadverläufe

Die Grobfehlerrate des abgewandelten Algorithmus verbessert sich im Vergleich zu Tabelle 4.1 bei jedem der Kurzzeitverfahren:

Verfahren	*Grobfehlerrate*	*Bemerkung*
Cepstrum	14%	Gaußfenster

Norm. Kreuzkorrelation	11%	Fensterlänge aus Gleichung 3.7
Norm. Kreuzkorrelation	8%	Fensterlänge aus Gleichung 3.14
Autokorrelation	5%	Rechteckfenster
Rootcepstrum	4.5%	Gaußfenster, Quadratwurzel

Tabelle 4.2 Fehlerraten der Pfad-beschränkten F_0-Verfahren ($d_{\max}$=0.4, d=d_{Oktave})

Mit dem auf dem Rootcepstrum basierten F_0-Detektor erreichen wir auf dem SPONTAN-Korpus bessere Grobfehlerraten, als fast alle anderen F_0-Detektoren, deren Grobfehlerraten in [Kie96] angegeben sind.
Alternativ zu der eben dargestellten Ausnutzung der Information aus Nachbarframes mittels einer Glattheitsforderung an den Grundfrequenzverlauf haben wir uns auch andere Varianten überlegt und getestet: darunter die Ergänzung der Gewinnfunktion um Strafkosten bei großen Frequenzsprüngen oder um Strafkosten für große Abweichungen vom bisherigen Mittelwert der Grundfrequenz. Jedes dieser Verfahren war schlechter als das in diesem Kapitel beschriebene.

Eine Fehleranalyse des so um die Glattheitsforderung an den Grundfrequenzverlauf ergänzten Algorithmus zeigt im Vergleich zu Abbildung 4.2 eine starke Reduzierung der Schätzung von T_0-Vielfachen:

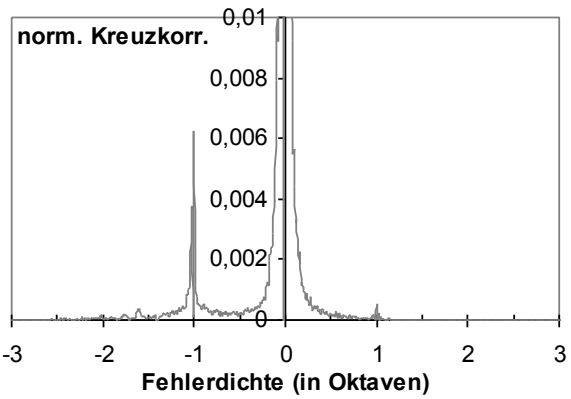

Abbildung 4.8 Fehlerdichte nach Einführung der Glattheitsforderung

Die höhere Grobfehlerrate bei Benutzung der Kreuzkorrelation im Vergleich zu Autokorrelation und Rootcepstrum führen wir darauf zurück, daß die lokalen Maxima bei Vielfachen der Grundperiodendauer in der Autokorrelationsfunktion stärker abfallen, als bei der Kreuzkorrelation (siehe Abbildung 3.4 und Abbildung 3.5 in Kapitel 3.5.2). Abhilfe schafft eine Dämpfung der lokalen Maxima bei Vielfachen der Grundperiodendauer in der Kreuzkorrelationsfunktion. Sei $\rho_i(t)$ die Kreuzkorrelationsfunktion für Frame i, bei der das globale Korrelationsmaximum bei t=0 bereits entfernt sein soll (siehe Abbildung 4.1). Wir haben zwei Methoden für diese Dämpfung überprüft, dabei soll $\rho_i'(t)$ die gedämpfte Kreuzkorrelationsfunktion sein:

1. Methode:
   Hier wird ausgenutzt, daß ein periodisches Signal mit Periodendauer T auch periodisch mit Periodendauer kT ($k \in \{2,3,4,...\}$) ist:

$$\rho_i'(t) = \rho_i(t) - c_1 \sum_{k=2}^{k_{\max}} \rho_i(\frac{t}{k}) \tag{4.6}$$

   Die Wahl des Dämpfungsparameters c_1 mit $0 \le c_1 \le 1$ ist kritisch: zu kleine Werte dämpfen Vielfache der Grundperiodendauer nur schwach, zu große Werte können dazu führen, daß ein durch eine starke Formante verursachtes lokales Maximum das lokale Maximum bei der Grundperiodendauer so stark dämpft, daß der gedämpfte Korrelationskoeffizient bei der Grundperioden kleiner wird, als der bei der Formante. Das wäre beispielsweise in Abbildung 3.4 bei $c_1 \ge 0.3$ der Fall.

2. Methode:
   Hier wird ausgenutzt, daß das absolute Maximum in $\rho_i(t)$ in den meisten Fällen bei der Grundperiodendauer T_0 liegt, sonst normalerweise bei ganzzahligen Vielfachen von T_0 (siehe Abbildung 4.2)

$$\rho_i'(t) = \begin{cases} \rho_i(t), & \text{für } t < c_1 t_{\max}, \text{ mit } t_{\max} = \{x \mid \forall y : \rho_i(x) > \rho_i(y)\} \\ c_2 \rho_i(t), & \text{sonst} \end{cases} \tag{4.7}$$

   Dabei gilt folgender Wertebereich für die beiden Parameter c_1 und c_2:

   $1 \le c_1 \le 2$
   $0 \le c_2 \le 1$

   Wir haben in unseren Versuchen festgestellt, daß die genaue Wahl der beiden Parameter relativ unkritisch ist, das heißt eine Änderung dieser beiden Parameter in einem weiten Bereich hat nur geringe Auswirkungen auf die Grobfehlerrate. Wir setzen c_1=1.5 und c_2=0.5.

Der Nachteil der ersten Methode ist, daß starke niedrige Formanten das lokale Korrelationsmaximum bei der Grundperiodendauer T_0 dämpfen. Die Grobfehlerrate des auf der Kreuzkorrelationsanalyse basierten Algorithmus konnte durch Benutzung der zweiten Methode etwas stärker reduziert werden: sie fiel von 8.1% (siehe vorige Tabelle) auf 5.9%. Die folgende Abbildung zeigt die resultierende Fehlerdichte im Vergleich zur vorigen Abbildung 4.8.

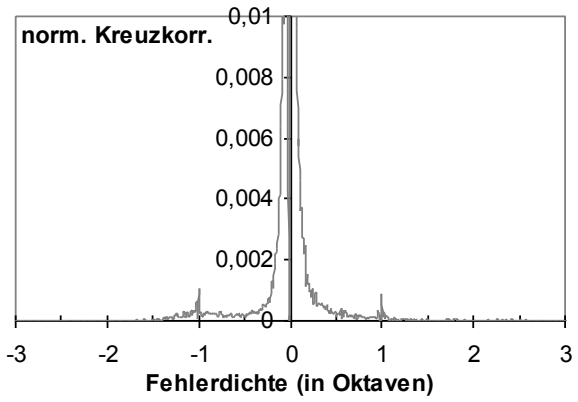

Abbildung 4.9 Fehlerdichte nach der Dämpfung von T_0-Vielfachen

4.5 Nachbearbeitung der Grundfrequenzkontur

Eine Analyse der ausgegebenen Grundfrequenzkonturen zeigte, daß die automatisch erzeugten und später manuell korrigierten Grundfrequenzkonturen aus der SPONTAN-Stichprobe stellenweise ein wenig glatter verlaufen, als die durch unseren Algorithmus erzeugten F0-Konturen. Das lies uns vermuten, daß die F0-Konturen der SPONTAN-Stichprobe nach der automatischen Erzeugung mit einem Glättungsfilter nachbearbeitet wurden. Ein solcher anschließender Glättungsvorgang ist bei den meisten Grundfrequenzalgorithmen üblich, deshalb haben auch wir den Einfluß von Glättungsfiltern auf die Grobfehlerrate bei der Evaluierung auf der SPONTAN-Stichprobe überprüft.

Der erste Filterungschritt besteht bei uns aus einem Medianfilter zur Beseitigung von kurzzeitigen mikroprodischen Schwankungen (Jitter) in der T_0-Kontur, grobe Ausreißer wie Oktavsprünge können aufgrund der Glattheitsbedingung (siehe voriges Kapitel) im Grundfrequenzpfad nicht vorkommen. Im zweiten Schritt wird optional eine Glättung der F_0-Kontur durch einen Mittelwertfilter vorgenommen. Der Einfluß beider Filter ist nur von der Breite der Filtermaske abhängig. Dabei sollte die Filtermaske allerdings nicht zu groß gewählt werden: beim Medianfilter führt die Verwendung von zu breiten Filtermasken sonst zu treppenartigen Artefakten in der ausgegebenen F_0-Kontur. Außerdem werden durch die Filterung Änderungen der Grundperiodendauer, die innerhalb der Maskenbreite auftreten, gedämpft. Wie die folgende Tabelle zeigt, konnte durch den Einsatz der Filterung die Grobfehlerrate unseres Algorithmus (Rootcepstrum, d_{max}=0.11) auf dem SPONTAN-Korpus um etwa 16% relativ von 3.5% auf 3.0% reduziert werden. In der Tabelle ist die Medianfilterbreite in der linken Spalte angegeben, die Mittelwertfilterbreite in der oberen Zeile. Die mit einem Strich gekennzeichneten Parameterkombinationen wurden von uns nicht getestet, da sie auf Grund der anderen Ergebnisse als wenig aussichtsreich eingeschätzt wurden. Das beste Ergebnis konnte bei Verzicht auf den Einsatz eines Medianfilters und mit einer Mittelwertfilterbreite von 3 (in der Tabelle fett gedruckt) erzielt werden.

Filterbreite	1	3	5	7
1	3.5%	**3.0%**	3.0%	-
3	3.5%	3.2%	3.1%	3.5%
5	-	3.4%	-	-

Tabelle 4.3 Einfluß der Filterung auf die Grobfehlerrate (links Median-, oben Mittelwert-Gitterbreite)

4.6 Parameteroptimierung

Die Arbeitsweise des von uns vorgestellten Algorithmus wird durch eine Reihe von Parametern beeinflußt. Im folgenden werden alle Parameter und deren Einfluß auf die Leistung des Verfahrens aufgeführt.

1. Die Breite des Gauß-Tiefpaßfilters.

   Die Verwendung einer Tiefpaßfilterung in der Vorverarbeitung wurde bereits in Kapitel 4.2 begründet.

2. Die kleinste erwartete Grundfrequenz $F0_{min}$.

   Je kleiner dieser Parameter gewählt wird, desto größer muß das Kurzzeitanalysefenster gewählt werden: zur Detektion von Grundperioden mit der Höchstdauer $T0_{max}=F0_{min}^{-1}$ muß das Analysefenster mindestens 2 Grundperioden enthalten, also eine Mindestlänge von $2 \cdot T0_{max}$ aufweisen. Das Fenster sollte nicht größer als nötig gewählt werden, da eine Vergrößerung nur eine Erhöhung des Rechenaufwands mit sich bringt.
   Dieser Parameter $F0_{min}$ wird von den meisten Grundfrequenzverfahren zwischen 35 bis 50 Hz gesetzt. Da auf der SPONTAN-Stichprobe kein Referenzgrundfrequenzwert kleiner als 50 Hz ist, haben wir für alle bisherigen Experimente $F0_{min} = 50$ Hz gesetzt.

3. Die größte erwartete Grundfrequenz $F0_{max}$.

   Dieser Wert stellt eine obere Grenze für die berechneten Grundfrequenzwerte dar. Zusammen mit $F0_{min}$ wird damit die Menge P der zu untersuchenden F_0-Verläufe (siehe Gleichung 4.5) auf diejenigen Verläufe eingeschränkt, bei denen in jedem Frame die Restriktion $F0_{min} \leq F0 \leq F0_{max}$ gilt. Die obere Schranke ist beispielsweise im Hinblick auf geflüsterte Vokale sinnvoll, da bei diesen eine Verwechslungsgefahr zwischen der Frequenz F_1 der ersten Formante und der Grundfrequenz F_0 besteht. Der Einfluß der oberen Schranke auf die Ausführungszeit des Algorithmus ist äußerst gering.
   Da auf der SPONTAN-Stichprobe kein Referenzgrundfrequenzwert größer als 550 Hz ist, haben wir bisher $F0_{max} = 550$ Hz gesetzt.

4. Der Wurzelexponent *k* bei der Kurzzeitanalyse per Rootcepstrum.

Die Auswahlmöglichkeiten im Hinblick auf die Kurzzeitanalysefunktion als nichtnumerischer Entwurfsparameter wurden bereits in Abschnitt 4.3 untersucht. Dabei lieferte das Rootcepstrum (k=1/2) die besten Ergebnisse. Der Einfluß des Parameters k auf die Grobfehlerrate wird im folgenden genauer untersucht.

5. optionale Energienormierung des Rootcepstrums

   Optional kann bei jedem der Korrelations- und Cepstrumverfahren eine Energienormierung durchgeführt werden, indem jeder Koeffizient in der Kurzzeitfunktion durch den ersten Korrelations- beziehungsweise Cepstrumkoeffizienten geteilt wird. Dieser erste Cepstrumkoeffizient ist gerade die Summe der Energien aus allen Bändern in der entsprechenden spektralen Repräsentation des Signals.

6. Der Parameter $d_{\max}$ für die Glattheitsforderung in Gleichung 4.5

   Dieser Parameter beschränkt die zu untersuchenden F_0-Pfade $\vec{t}$ auf diejenigen, die der Glattheitsbedingung $d_{\text{oktave}}(t_i, t_{i+1}) < d_{max}$ genügen. Die Wahl der Abstandsfunktion d_{octave} als weiteren Entwurfsparameter wurde bereits begründet.

7. Die Breite b des Mittelwertfilters im Nachbearbeitungsschritt der F_0-Kontur.

   Der Einfluß dieses Parameters wurde schon in Kapitel 4.5 untersucht.

Damit besteht der Parametersatz unseres Algorithmus aus den 7 angegebenen Parametern, die zu einem Parametersatz Θ zusammenfaßt werden. Gesucht ist der Parametersatz, der die erwartete Fehlerrate des Verfahrens minimiert. Wir können weder ein Verfahren zur gemeinsamen numerischen Optimierung der aufgezählten Parameter angeben, noch zur numerischen Optimierung eines einzelnen Parameters. Deshalb sind wir von einem uns sinnvoll erscheinenden Parametersatz Θ^0 ausgegangen, und haben dann nacheinander jeden Parameter einzeln unter Durchführung einer Rastersuche optimiert, wobei als Gütemaß die Grobfehlerrate $E(\Theta)$ des Algorithmus auf der SPONTAN-Stichprobe herangezogen wurde. Dadurch kann allerdings weder garantiert werden, daß ein global optimaler Parametersatz gefunden wird, noch daß der so bestimmte Parametersatz auf anderen Sprachstichproben gute Ergebnisse liefert. Wir nehmen aber an, daß alle Parameter relativ unabhängig von der untersuchten Sprachstichprobe sind: die Parameter 1 bis 3 sind vom erwarteten Grundfrequenzbereich abhängig, der bei spontansprachlichen Äußerungen von Erwachsenen den Bereich von etwa 35 Hz bis 550 Hz überdeckt, Parameter 4 und 5 optimieren die Kurzzeitanalysefunktion und Parameter 6 stellt eine einfache Glattheitsforderung an die bestimmten F_0-Verläufe.
Auf eine sonst übliche Unterteilung der Sprachdaten in Trainingsmenge und Testmenge haben wir verzichtet, da wir zum Vergleich mit anderen F_0-Detektoren aus [Kie96] den gesamten SPONTAN-Korpus als Testmenge benutzen mußten.
Als Startparametersatz für die Optimierung wählten wir:

Θ^0 =(1/(550 Hz), 6400 Hz, 50 Hz, 550 Hz, 1/2, nein, 0.4, 1).

Der zu betrachtende F_0-Bereich von 50 bis 550 Hz wurde aus den Referenzgrundfrequenzwerten bestimmt. Daraus wird die Neuabtastrate so festgelegt, daß die maximal erwartete Grundperiodendauer von 1/50 s genau zweimal in das Kurzzeitfenster paßt, und sich für die FFT eine Kurzzeitfensterlänge als Potenz von 2 ergibt:

$$f_{\text{neuabtast}} = f_{\text{abtast}} \cdot \frac{2^{\lfloor \log_2 n \rfloor}}{n}, \text{ mit } n = 2 \cdot \frac{f_{\text{abtast}}}{F0_{\text{min}}}$$

Dabei ist n die Anzahl der Abtastwerte im Kurzzeitfenster, das genau zweimal die maximal erwartete Grundperiodendauer enthält. Mit f_{abtast}=10000 Hz und $F0_{\text{min}}$=50 Hz ergibt sich n=400. Nach abrunden auf die nächstkleinere Zweierpotenz ergibt sich ein Kurzzeitanalysefenster mit 256 Werten woraus eine Neuabtastrate von 6400 Hz folgt.
Die Wahl des Parameters d_{max}=0.4 Oktaven erfolgte in Kapitel 4.4 aus der Analyse der Referenz-F_0-Werte der SPONTAN-Stichprobe, das Rootcepstrum berechnen wir anfangs über die Quadratwurzel (k=1/2). Eine nachfolgende Glättung der berechneten Grundfrequenzkontur findet nicht statt (b=1).
Mit diesem Parametersatz erhalten wir den gleichen Algorithmus wie bei dem Experiment in Tabelle 4.4 mit einer Grobfehlerrate von 4.5%. Die folgenden Tabellen zeigen den Einfluß der Variierung der einzelnen Parameter im Startparametersatz.

1. Breite g des Gauß-Tiefpaßfilters

Filterbreite	$(330\text{ Hz})^{-1}$	$(440\text{ Hz})^{-1}$	$(550\text{ Hz})^{-1}$	$(680\text{ Hz})^{-1}$	0
Grobfehlerrate	4.5%	4.3%	4.5%	4.8%	5.5%

Diese Ergebnisse zeigen, daß sich die Grobfehlerrate beim Weglassen der Tiefpaßfilterung (Filterbreite=0) verschlechtert. Ein zu breiter Filter wirkt sich allerdings auch negativ auf die Leistung aus.

Als Alternative zur Gaußfunktion als Tiefpaßfilter haben wir an dieser Stelle auch einen Tiefpaß mit eine rechteckförmigen Impulsantwort getestet (siehe Kapitel 4.2). Interessanterweise lieferte dieser Filter trotz einer schlechteren Lokalisierung im Frequenzbereich (‚Überschwingen') als beim Gaußfilter ähnlich gute Grobfehlerraten:

Rechteckbreite	$(833\text{ Hz})^{-1}$	$(1000\text{ Hz})^{-1}$	$(1250\text{ Hz})^{-1}$
Grobfehlerrate	4.4%	4.5%	4.6%

2. kleinste erwartete Grundfrequenz $F0_{\text{min}}$

$F0_{\text{min}}$	50 Hz	35 Hz
Grobfehlerrate	4.5%	4.2%

Die Verbesserung der Grobfehlerrate nach der Vergrößerung des Kurzzeitanalysefensters ist relativ unerwartet, da die niedrigste in den SPONTAN-

Referenzen vorkommende Grundfrequenz 55 Hz beträgt. Wir führen die Verbesserung darauf zurück, daß auf dem größeren Kurzzeitfenster eine etwas robustere Schätzung der Grundperiodendauer möglich ist, da mehrere Grundperioden in die Schätzung eingehen, und das Fenster noch klein genug ist, so daß sich die Grundperiodendauer und die Artikulatorenstellung innerhalb des Fensters nur wenig verändert. Ein Nachteil bei der Vergrößerung der Fensterbreite ist die Erhöhung des Rechenaufwandes bei gleichzeitiger Vergrößerung des Analysefensters für die FFT.

3. größte erwartete Grundfrequenz $F0_{max}$

$F0_{max}$	550 Hz	3200 Hz
Grobfehlerrate	4.5%	4.6%

Dieser Parameter hat nur sehr geringen Einfluß auf die Grobfehlerrate: selbst ohne Einschränkung der maximal erwarteten Grundfrequenz steigt die Grobfehlerrate nur minimal an. Die Angabe einer obere Schranke für die erwartete Grundfrequenz reduziert aber zusätzlich noch leicht den Rechenaufwand.

4. Wurzelexponent k (Rootcepstrum)

k	0.2	0.4	0.5	0.7	1
Grobfehlerrate	7.9%	4.7%	4.5%	4.4%	4.9%

Bei Wahl des Wurzelexponenten k=1 entspricht das Rootcepstrum genau der Autokorrelation. Die besten Ergebnisse werden mit k=0.7 geliefert. Der Wert k=0.5 bringt nur minimal schlechtere Grobfehlerraten, hat aber den Vorteil, daß die Quadratwurzel auf einigen Prozessorarchitekturen etwas schneller ausgewertet werden kann, als die allgemeine Potenzfunktion.

5. Normierung des Rootcepstrums

Normierung	Nein	ja
Grobfehlerrate	4.5%	4.4%

Die Energienormierung bringt eine leichte Verbesserung der Grobfehlerrate, wobei der Rechenaufwand allerdings geringfügig erhöht wird.

6. Glattheitsparameter d_{max}

d_{max}	0.05	0.11	0.2	0.4	∞
Grobfehlerrate	3.0%	2.9%	3.6	4.5%	7.7%

Mittels dieses Parameters kann die Grobfehlerrate besonders stark reduziert werden: $d_{max}=\infty$ entspricht der ersten Version unseres Grundfrequenzalgorithmus aus Kapitel 4.3, die keine Informationen aus benachbarten Frames zur Bestimmung der Grundfrequenzkontur heranzog. Mit $d_{max}<0.4$ nehmen wir von vornherein einige Grobfehler in Kauf, da in den Referenzkonturen einige we-

nige Sprünge über ungefähr 0.4 Oktaven vorkommen (siehe Abbildung 4.4). Die so verstärkte Glattheitsbedingung führt aber dennoch zu einer robusteren F_0-Schätzung.

7. Mittelwertfilterbreite *b* (Glättung der erzeugten F_0-Kontur)

b	1	3	5
Grobfehlerrate	4.5%	4.5%	4.6%

Dieser Parameter hat hier nur minimalen Einfluß auf die F0-Kontur, er wurde bereits in Kapitel 4.5 besprochen. Erst bei einer gröberen Neuabtastung, und damit der Schätzung von T_0 auf einem gröberen Raster, bringt diese Filterung der F_0-Kontur eine stärkere Verbesserung: bei einer Neuabstastrate von 3200 Hz statt 6400 Hz wie hier (siehe Kapitel 4.7.1) verbessert sich die Grobfehlerrate bei Benutzung der Konturfilterung von 3.0% auf 2.6%.

Das Einsetzen des jeweils besten Wertes für jeden Parameter liefert einen neuen Parametersatz Θ^1 =(1/(440 Hz), 35 Hz, 550 Hz, 0.7, ja, 0.11, 3). Weitere Iterationen der Optimierung des einzelnen Parameter bringen nur marginale Veränderungen der Parameter und der resultierenden Grobfehlerraten. Die Grobfehlerraten und Laufzeiten für den Startparametersatz und den besten Parametersatz Θ=(1/(500 Hz), 35 Hz, 550 Hz, 0.5, ja, 0.11, 3) sind in der unteren Tabelle angegeben. Die Laufzeit ist dabei als Echtzeitfaktor angegeben, das heißt als Quotient aus der für die Berechnung der F_0-Konturen benötigten Zeit, geteilt durch die Dauer alle Äußerungen. Dieser Echtzeitfaktor ist natürlich rechnerabhängig: die Messungen fanden auf allen Äußerungen der SPONTAN-Stichprobe auf einem mit 400 MHz getakteten Pentium II statt.

$F0_{min}$	Grobfehlerrate	Echtzeitfaktor
35 Hz	2.4 %	1/6,7
50 Hz	2.7 %	1/14,8

Die Unterschiede im Echtzeitfaktor resultieren daraus, daß das langsamere Verfahren 512er FFTs berechnet, während das schnellere mit einer 256er FFT auskommt. Die Fehlerdichte des optimierten Algorithmus ist unten abgebildet.

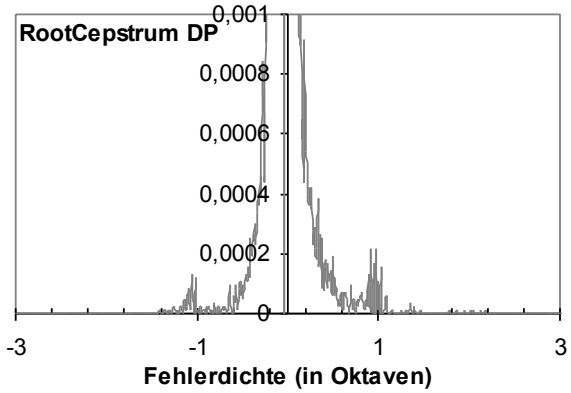

Abbildung 4.10 Fehlerdichte des optimierten F_0-Detektors

4.7 Beschleunigung des Verfahrens

Nachdem es im letzten Abschnitt um Optimierung der Grobfehlerrate des Verfahrens ging, werden jetzt einige Möglichkeiten zur Beschleunigung unseres Grundfrequenzalgorithmus vorgestellt.

4.7.1 Neuabtastung des Sprachsignals

Zur Erhöhung der Geschwindigkeit unseres Grundfrequenzverfahrens kann eine Neuabtastung (downsampling) des Eingabesignals durchgeführt werden. Ein Nachteil dieser Neuabtastung ist, daß dadurch die zeitliche Auflösung der (abgetasteten) Korrelationsfunktion beziehungsweise des Rootcepstrums reduziert wird, und damit auch die Frequenzauflösung des Verfahrens. Beispielsweise kann die Grundperiodendauer bei einem Downsamplingfaktor von 4 nur noch auf 4 Abtastwerte (bezogen auf das ursprüngliche Eingangssignal) genau bestimmt werden. Anders gesagt: die ausgegebenen Schätzwerte für die Grundperiodendauern liegen dann auf einem Raster mit einem Abstand von 4 Abtastwerten. Eine gröbere Neuabtastung bringt also einen Geschwindigkeitsgewinn auf Kosten der Fehlerrate. Die unteren Tabellen zeigen diesen Zusammenhang für verschiedene Abtastraten. Bei der oberen der beiden Tabellen wurde der Mittelwertfilter zur abschließenden Glättung der erzeugten Grundfrequenzkonturen (siehe Kapitel 4.5) benutzt, bei der unteren nicht.

$f_{\text{neuabtast}}$	Grobfehlerrate	Echtzeitfaktor (PII/400Mhz)
6400 Hz	2.7%	1/14.8
3200 Hz	2.6%	1/30
1600 Hz	4.5%	1/68

Tabelle 4.5 Zusammenhang Neuabtastung - Fehlerrate – Laufzeit (Mittelwertfilter)

$f_{\text{neuabtast}}$	Grobfehlerrate	Echtzeitfaktor

6400 Hz	2.8%	1/14.8
3200 Hz	3.0%	1/30
1600 Hz	6.6%	1/68

Tabelle 4.6 Zusammenhang Neuabtastung - Fehlerrate (ohne Mittelwertfilter)

Wie in den Tabellen zu sehen ist, wird bereits durch die Konturfilterung über den Mittelwertfilter die Grobfehlerrate etwas verbessert. Trotzdem steigt die Grobfehlerrate bei einer gröberen Neuabtastung immer noch enorm an.
Da die Neuabtastung die Laufzeit des Verfahrens stark verringert, war unser Ziel die Ausbesserung der durch die Neuabtastung bedingten Grobfehler. Der Ansatz zu dieser Reduzierung der Fehlerrate besteht in der Durchführung einer hochauflösenden Kurzzeitanalyse nur in einem kleinen Bereich um den relativ grob geschätzten T_0-Wert auf jedem Frame. Für die Kurzzeitanalyse bietet sich im Hinblick auf die Versuchsergebnisse in Tabelle 4.1 die Autokorrelationsfunktion $\varphi_n(k)$ an (siehe Kapitel 3.5.2):

$$\varphi_n(k) = \sum_{m=0}^{N-1-k} x(n+m)x(n+m+k) \tag{4.8}$$

Dabei ist n wieder der Beginn des betrachteten Frames (Rechteckfensters) im nicht neuabgetasteten Sprachsignal $x(t)$. N ist die Größe des Rechteckfensters, wobei das Fenster genauso groß gewählt wird, wie schon in der Kurzzeitanalyse im Hauptteil unseres Grundfrequenzalgorithmus, also das Zweifache der maximal erwarteten Grundperiodendauer. Wir werten dabei $\varphi_n(k)$ nur für einige wenige k in einer engen Umgebung des groben T_0.Schätzwertes T aus. Wählen wir die Größe der Umgebung zu u, berechnen wir $\varphi_n(k)$ nur für ganzzahlige $k \in [T-u/2, T+u/2]$. Dabei ist T die grobe Schätzung für die Anzahl der Abtastwerte in einer Grundperiode in $x(t)$ ist, u wird analog auch in Abtastwerten gemessen. Die folgende Tabelle zeigt die Ergebnisse einiger Experimente mit verschieden großen Umgebungen u (angegeben in 10 kHz Samples) bei einer Neuabtastrate von 1600 Hz, der Downsamplingfaktor $d = f_{abtast}/f_{neuabtast}$ beträgt also 6,25. Der Kontur-Mittelwertfilter war in diesem Experiment ausgeschaltet.

$f_{neuabtast}$	U	Grobfehlerrate	Echtzeitfaktor
1600 Hz	0	6.6%	1/68
1600 Hz	3	3.6%	1/67
1600 Hz	5	3.2%	1/66
1600 Hz	7	3.2%	1/65
1600 Hz	9	3.2%	1/65
1600 Hz	15	3.2%	1/62
1600 Hz	∞	7.6%	1/13

Tabelle 4.7 Zusammenhang Größe der Suchumgebung u und Grobfehlerrate

Das Experiment mit $u=\infty$ (zusammen mit einer Entfernung des Autokorrelationsmaximums bei $t=0$) entspricht genau dem Versuch mit der Autokorrelation in Tabelle 4.1, da dann das Ergebnis unabhängig von der vorher bestimmten groben Grundfrequenzkontur wird. Aufgrund dieser Ergebnisse, und denen einiger hier nicht aufgelisteter Versuche mit anderen Neuabtastraten, machen wir die Wahl der Größe u der abzusuchenden Umgebung um die T_0-Hypothese T vom Downsamplingfaktor $d=f_{abtast}/f_{neuabtast}$ abhängig: wir wählen $u=d$. In der folgenden Tabelle sind Verbesserungen durch der Verwendung dieses Verfahrens bei den verschiedenen Neuabtastraten angegeben. Dieser Versuch fand unter den gleichen Bedingungen statt, wie der Versuch, dessen Ergebnisse in Tabelle 4.5 (ohne Verfeinerungschritt) angegeben sind. Insbesondere wurde hier der Mittelwertfilter zur anschließenden Glättung der F_0-Kontur wieder eingeschaltet.

$F_{neuabtast}$	Grobfehlerrate	Echtzeitfaktor
6400 Hz	2.5%	1/14.7
3200 Hz	2.4%	1/30
1600 Hz	2.9%	1/65

Tabelle 4.8 Zusammenhang Größe der Suchumgebung u und Grobfehlerrate

Dies zeigt, daß die Erhöhung der Grobfehlerrate bei niedrigen Neuabtastraten stark vermindert werden konnte, so daß jetzt bei einer starken Beschleunigung des Verfahrens nur noch wenig an Genauigkeit eingebüßt wird.

4.7.2 Auslassung von Frames

Eine naheliegende weitere Möglichkeit zur Beschleunigung des Verfahrens ist die Reduzierung der Framerate, das heißt der Anzahl der Kurzzeitfenster (Frames) pro Sekunde, auf denen eine Kurzzeitanalyse durchgeführt wird. Die Referenzgrundfrequenzverläufe der SPONTAN-Stichprobe geben alle 12,8 ms einen Grundfrequenzwert an, was einer Framerate von etwas 78 Frames pro Sekunde entspricht. Da sich die Grundfrequenz nur langsam verändert, ist anzunehmen, daß der F_0-Verlauf mit einer relativ geringen Framerate verfolgt werden kann. Zur Bewertung des so beschleunigten Verfahrens mit Hilfe der SPONTAN-Referenzen muß anschließend durch Neuabtastung der automatisch erzeugten Grundfrequenzkontur wieder eine Framerate von 78 Frames pro Sekunde hergestellt werden. Die Neuabtastung führten wir in unseren Experimenten mittels linearer Interpolation durch. Außerdem haben wir die im letzten Abschnitt beschriebene feinauflösende Suche nach der Interpolation durchgeführt, da diese Suche im Vergleich zur Berechnung des Cepstrums und der DP-Pfadbestimmung nur einen sehr kleinen Bruchteil des Rechenaufwandes ausmacht.
Bei der Reduzierung der Framerate muß beachtet werden, daß der Glattheitsparameter d_{max} entsprechend vergrößert werden muß. Dieser Parameter beschränkt die Steigung der Grundfrequenzkontur mittels der Bedingung $d_{oktave}(t_i,t_i+1)<d_{max}$, wobei

t_i und t_i+1 die Grundperiodendauern zweier benachbarter Frames auf dem Grundfrequenzhypothesenpfad sind, was bereits in Kapitel 4.4 im Detail beschrieben wurde. Eine obere Schranke für d_{max} wurde aus den Referenz-F_0-Konturen bestimmt: in den Referenzkonturen galt $d_{oktave}(t_i,t_i+1)<0.4$ für alle i (siehe Abbildung 4.4). Die Verteilung von $d_{oktave}(t_i,t_i+1)$ auf den Referenzkonturen zeigt die folgende Abbildung für verschiedene Frameraten. Die dünne gestrichelte Linie ist die Verteilung für die originalen 78 Frames pro Sekunde, die dünne durchgezogene Linie für 39 und die dicke Linie für 19,5 Frames pro Sekunde.

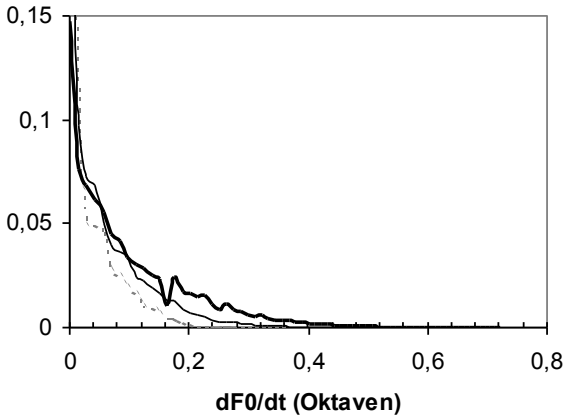

Abbildung 4.11 Verteilung von $d_{oktave}(t_i,t_i+1)$ bei verschiedenen Frameraten

Wie in der Abbildung zu sehen ist, muß d_{max} mit wachsendem Frameabstand vergrößert werden. Den optimalen Wert für d_{max} für verschiedene Frameraten haben wir wieder wie im vorigen Kapitel durch eine Rastersuche bestimmt, wobei wir von der aus dem oberen Histogramm bestimmten oberen Schranke ausgegangen sind.
Die untere Tabelle gibt Grobfehlerraten, Laufzeiten und den optimalen Wert für d_{max} für einige reduzierte Frameraten (beziehungsweise vergrößerte Frameabstände) an. Als Ausgangsalgorithmus wurden die beiden schnellsten Verfahren aus dem letzten Abschnitt mit 3200 Hz beziehungsweise 1600 Hz Neuabtastrate verwendet.

Frameabstand	d_{max}	Grobfehlerrate	Echtzeitfaktor
12,8 ms	0,11	2.9%	1/65
25,6 ms	0,11-0,16	2.8%	1/116
38,4 ms	0,14	3.5%	1/158
51,2 ms	0,25	4.2%	1/183

Tabelle 4.9 Ergebnisse für verschiedene Frameabstände ($f_{neuabtast}$ =1600 Hz)

Frameabstand	d_{max}	Grobfehlerrate	Echtzeitfaktor
12,8 ms	0,11	2.4%	1/30
25,6 ms	0,15-0,2	2,4%	1/55

Tabelle 4.10 Ergebnisse für verschiedene Frameabstände ($f_{neuabtast}$=3200 Hz)

Wie aus den Ergebnissen ersichtlich ist, wird die Grobfehlerrate bei einer Reduzierung des Frameabstandes auf etwa 25 ms nicht verschlechtert, die Laufzeit des Algorithmus aber halbiert.

4.8 Schritthaltende F_0-Bestimmung

Viele Anwendungen verarbeiten Sprachdaten schritthaltend. Sie warten nicht bis ein Satz oder ein noch größerer Textabschnitt eingelesen wurde, und geben erst dann das Endresultat aus, sondern geben Zwischenergebnisse schon während des Einlesevorgangs an übergeordnete Verarbeitungsschichten weiter. Ein Beispiel für so eine Anwendung ist ein automatisches Spracherkennungssystem, das die bisherige beste Hypothese für den Anfang eines Satzes bereits auf dem Bildschirm ausgibt, während der Benutzer den Satz noch gar nicht zu Ende gesprochen hat. Für so eine Anwendung wäre die oben beschriebene Version unseres F_0-Detektors nicht geeignet, da dieser erst am Ende einer Äußerung den optimalen F_0-Pfad gemäß Gleichung 4.3 und 4.5 mittels Backtracking (siehe Kapitel 4.9.4) bestimmt. Um eine schritthaltende Variante unseres Algorithmus zu erhalten, muß das Selektionskriterium für die Suche des optimalen Pfades abwandelt werden: das ursprüngliche Kriterium

$$\vec{t} = \underset{\vec{t}' \in P}{\operatorname{argmax}}\, G(\vec{t}')$$

$$\text{mit } P = \left\{ \vec{t}' \in \{0,1,...,i_{\max}\}^{j_{\max}} \mid d(t_j, t_{j+1}) < d_{\max},\ j \in \{0,1,..., j_{\max} - 1\} \right\}$$

aus Gleichung 4.5 wird zu

$$\vec{t} = \begin{pmatrix} t_1 \\ ... \\ t_{j_{\max}} \end{pmatrix} \text{ mit } t_j = t_j^* \text{ und } \vec{t}_j^{\,*} = \underset{\vec{t}' \in P_j}{\operatorname{argmax}}\, G(\vec{t}')\ \forall j \tag{4.9}$$

Dabei ist P_j die Menge der möglichen Verläufe der Grundperiodendauer über nur j Frames statt j_{max} wie in Gleichung 4.5. Zur Bestimmung von t_j - also der Grundperiodendauer für den j-ten Frame – werden jetzt also nur noch die Frames 1 bis j benutzt, statt wie vorher auch die nachfolgenden Frames für die Entscheidung zu berücksichtigen. Auf dieser Teilfolge wird der optimale F_0-Verlauf genauso wie in der nicht schritthaltenden ursprünglichen Version berechnet.
Da jetzt keine Informationen über nachfolgende Teile des Sprachsignals mehr einbezogen werden dürfen, kommt es zwangsweise zu einer Verschlechterung der Fehlerrate des Algorithmus. Verglichen mit der nicht schritthaltenden Version des Algorithmus aus dem letzten Kapitel steigt die Grobfehlerrate beispielsweise von 2.8% auf 4.5%. Die Fehleranalyse in Abbildung 4.12 zeigt, daß sich im Vergleich zur Fehlerdichte des nicht schritthaltenden Algorithmus in Abbildung 4.10 das lokale Maximum bei -1 erhöht hat. Das läßt vermuten, daß besonders die Anzahl der Fehldetektionen im Zusammenhang mit Laryngalisierungen erhöht wurde. Diese Vermutung

konnten wir leider nicht überprüfen, da uns eine Etikettierung der Laryngalisierungen nicht zur Verfügung stand.

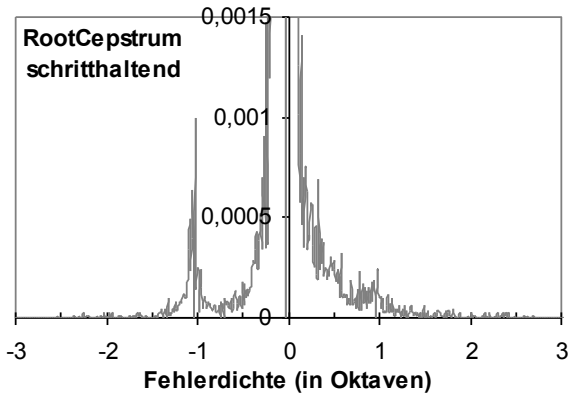

Abbildung 4.12 Fehlerdichte des schritthaltenden F_0-Algorithmus

Die Erhöhung der Grobfehlerrate läßt sich reduzieren, wenn man nicht völlig auf eine Vorausschau auf zukünftige Frames verzichten muß, sondern sich statt dessen eine feste Vorausschau auf die folgenden *n* Frames leisten kann. Dabei müssen dann die der Grundfrequenzanalyse nachgeschalteten Verfahren im allgemeinen auch um die diesen *n* Frames entsprechende Zeitspanne verzögert werden. Die Implementierung der schritthaltenden Variante mit Vorausschau wird in Kapitel 4.9.4 beschrieben. Die folgende Tabelle zeigt die Grobfehlerrate in Abhängigkeit von der genutzten Vorausschau. Dabei sind wir vom Verfahren aus dem letzten Kapitel mit $f_{\text{neuabtast}}$=1600 Hz und einem Frameabstand von 25,6 ms ausgegangen (Echtzeitfaktor 1/116). Das Verfahren mit n=∞ entspricht der nicht-schritthaltenden Version des Algorithmus.

N	Grobfehlerrate
∞	2.8%
≥10	2.8%
5	2.9%
4	3.0%
3	3.0%
2	3.2%
1	3.5%
0	4.5%

Tabelle 4.11 Ergebnisse für verschieden lange Vorausschau

Aus der Tabelle folgt, daß der schritthaltende Algorithmus bereits mit einer kurzen Vorausschau von nur *n*=10 Frames – also etwa einer Viertel Sekunde – praktisch nicht schlechter als seine nicht-schritthaltende Variante ist. Die Durchführung des Backtrackings in jedem Frame führt zu einer leichten Erhöhung der Laufzeit des Grundfrequenzverfahrens. Allerdings ist diese Erhöhung selbst bei einer unnötig

großen Vorausschau von n=40 – ungefähr einer Sekunde – praktisch nicht meßbar: der Echtzeitfaktor bleibt bei 1/116.

4.9 Implementierung

Als Ergebnis unserer Versuche in den vorangegangenen Kapiteln kann unser Grundfrequenzalgorithmus als zweistufiges Verfahren aufgefaßt werden. In der ersten Stufe wird mittels dynamischer Programmierung ein mehr oder weniger grober Grundfrequenzhypothesenpfad aus einer Cepstrogramm- beziehungsweise Korrelationsmatrix bestimmt. In der anschließenden zweiten Stufe wird in jedem Frame mittels einer feinauflösenden Suche in einem kleinen Bereich um den grob bestimmten Grundfrequenzwert eine genauere Hypothese bestimmt. Die Laufzeit und Fehlerrate des Verfahrens werden durch die Auflösung der Korrelationsmatrix in der ersten Stufe bestimmt: bei Wahl einer gröberen Auflösung kann die Laufzeit auf Kosten der Fehlerrate reduziert werden.

I. Eingabe
- mit Abtastrate f_{eingabe} (in Hz) abgetastetes und quantisiertes Sprachsignal

II. Ausgabe
- mit Abtastrate f_{ausgabe} (in Hz) abgetasteter Grundfrequenzverlauf

III. Ablauf
1. Tiefpaßfilterung
2. Einteilung des Eingabesignals in Frames
3. Berechnung des Cepstrogramms (bzw. Korrelationsmatrix)
4. Pfadsuche im Cepstrogramm
5. Feinauflösende Suche
6. Konturfilterung

Parametrisierung:
- Festlegung der Ein-/Ausgabe über $f_{\text{eingabe}}, f_{\text{ausgabe}}$
- abzusuchender Grundfrequenzbereich $[F0_{\min}, F0_{\max}]$ (in Hz)
- maximale Vorausschau $t_{\text{vorausschau}}$ (in s) für den schritthaltenden Ablauf
- Abwägen zwischen kurzer Laufzeit und hoher Genauigkeit:
  - Neuabtastrate $f_{\text{neuabtast}} \leq f_{\text{eingabe}}$ (in Hz)
  - Abstand der Kurzzeitanalysefenster $f_{\text{framerate}} \leq f_{\text{ausgabe}}$ (in Hz)

In den folgenden Abschnitten wird kurz auf die Implementierung der einzelnen Ablaufschritte 1 bis 6 eingegangen. Dabei werden diese Einzelschritte so verzahnt, daß eine schritthaltende Verarbeitung des Eingabesignals erreicht wird.

4.9.1 Tiefpaßfilterung

Die Filterung per Faltung mit einer Gaußmaske lieferte keine besseren Grobfehlerraten als die Filterung über eine Rechteckmaske (siehe Kapitel 4.6). Wir haben uns für die Filterung über eine Rechteckmaske entschieden, da deren Laufzeit bei einer effizienten rekursiven Implementierung unabhängig von der Größe der Filtermaske ist:
Sei n (ungerade zur Vermeidung einer Phasenverschiebung) die Breite der Filtermaske mit Koeffizienten $1/n$, x_k mit $k \in \{1,2,...,k_{max}\}$ die Folge der Signalabstastwerte und y_k die Ergebnisfolge der Abtastwerte des gefilterten Signals. Nach Initialisierung von $y_{(n+1)/2}$ berechnet sich die Folge der y_k ($k>(n+1)/2$) folgendermaßen:

$$y_{(k+1)/2} = \frac{1}{n}\sum_{i=1}^{n} x_i$$

$$y_k = y_{k-1} + \frac{x_{k+(n-1)/2} - x_{k-(n+1)/2}}{n} \qquad (4.10)$$

4.9.2 Einteilung des Eingabesignals in Frames

Der Abstand zweier aufeinander folgender Frames ist als $1/f_{framerate}$ vorgegeben, wobei $f_{framerate}$ größer als $f_{ausgabe}$ sein darf, um die Laufzeit des Verfahrens zu reduzieren. Wie in den Experimenten in Kapitel 4.7.2 festgestellt wurde, nimmt die Grobfehlerrate erst ab einem Frameabstand von mehr als 25 ms (also $f_{framerate} < 40$ Hz) merklich zu.

4.9.3 Berechnung des Cepstrogramms

Die Berechnung des Cepstrogramms beziehungsweise der Korrelationsmatrix wurde in den Kapiteln 4.3 und 4.4 dargestellt. In unserer Implementierung wird ein Frame nach dem anderen abgearbeitet und die Kurzzeitanalyse jedes Frames sofort an die nachfolgende Pfadsuche übergeben. Die Korrelationsmatrix muß also nicht explizit gespeichert werden. Die Länge eines Kurzzeitfensters wird auf $2/F0_{min}$ gesetzt, so daß das Analysefenster mindestens zwei Grundperioden enthält.
Die Kurzzeitanalyse basiert auf der Berechnung des Energie-normierten Rootcepstrums (Kapitel 3.5.2) mit k=2, also der Quadratwurzel. Da die Kurzzeitanalyse der rechenzeitaufwendigste Schritt des Grundfrequenzalgorithmus ist, wird vorher eine Neuabtastung mit $f_{neuabtast} \leq f_{eingabe}$ (downsampling) durchgeführt. In den Versuchen in Kapitel 4.7.1 führte eine relativ grobe Abtastung mit $f_{neuabtast}$=1600 Hz zu einer leichten Erhöhung der Grobfehlerrate, ab $f_{neuabtast}$>3000 Hz konnte keine Verschlechterung der Grobfehlerrate mehr festgestellt werden. Da das Rootcepstrum effizient über zwei FFTs berechnet werden soll, muß die Anzahl der Abtastwerte im Kurzzeitfenster eine Zweierpotenz sein. Die Anzahl der Abtastwerte bei einer vorgegebenen Fensterbreite von $2/F0_{min}$ und der vom Anwender gewünschten Abtastrate $f_{neuabtast}$

ergibt im Allgemeinen keine Zweierpotenz. Aus diesem Grund wird $f_{neuabtast}$ auf die nächstgrößere Abtastfrequenz erhöht, bei der dies der Fall ist.
Mit dieser Vorgehensweise ergeben sich Kurzzeitfenster mit typischerweise 64, 128 oder 256 Abtastwerten. Dabei hat sich in unseren Versuchen gezeigt, daß bei Fenstergrößen mit ≤128 die Berechnung der normierten Kreuzkorrelationsfunktion (n/2 Koeffizienten bei n Abtastwerten im Fenster) im Zeitbereich schneller ist, als die Durchführung zweier FFTs zur Berechnung des Rootcepstrums. Die Verwendung der normierten Kreuzkorrelation erhöht zwar die Grobfehlerrate auf der SPONTAN-Stichprobe, aber in zeitkritischen Anwendungen kann optional auch diese Funktion zur Kurzzeitanalyse verwendet werden.
Anschließend wird noch das Maximum in der Kurzzeitanalyse bei t=0 bis zum ersten Nulldurchgang entfernt (siehe Kapitel 4.3).

4.9.4 Pfadsuche im Cepstrogramm

Das Ergebnis der Pfadsuche besteht aus einer mehr oder weniger groben Schätzung (grober bei kleinerer Neuabtastrate $f_{neuabtast}$) der Grundperiodendauer T_0 für jeden Frame. Die Berechnung des T_0-Verlaufs wird nach den Betrachtungen in Kapitel 4.4 im Cepstrogramm als Suche nach demjenigen T_0-Verlauf $\vec{t}$ aufgefaßt, der das Gewinnkriterium $G(\vec{t})$ in Gleichung 4.3 maximiert.
Nach den Versuchen in Kapitel 4.7.2 wird der Glattheitsparameter d_{max} aus Gleichung 4.5 in Abhängigkeit des Abstands d=1/ $f_{framerate}$ der Kurzzeitfenster gewählt:

d_{max} =0,11 für d≤12.8 ms
d_{max} =0,125 für d=25.6 ms
d_{max} =0,14 für d=38.4 ms
d_{max} =0,25 für d=51.2 ms

Diese Werte folgen nur aus den Experimenten in Kapitel 4.7.2, wir können sonst keine andere Begründung für diese Wahl liefern. Wir nehmen aber an, daß die auf der SPONTAN-Stichprobe gewonnenen Ergebnisse auch für andere spontansprachliche Stichproben repräsentativ sind. Mangels besseren Wissens interpolieren wir d_{max} bei Frameabständen d die zwischen den angegebenen Rasterpunkten liegen linear, beziehungsweise nehmen für d>51,2 den Randwert d_{max}=0,25.

Die Auswertung von $G(\vec{t})$ benötigt die Kurzzeitanalysen aus dem vorigen Schritt. Die zeitliche Auflösung der Pfadsuche wird durch die Auflösung dieser Kurzzeitanalyseframes bestimmt: der kleinstmögliche meßbare Abstand zwischen zwei Grundperiodendauern beträgt hier $1/f_{neuabtast}$, beziehungsweise $f_{eingabe}/f_{neuabtast}$ Samples im Eingabesignal. Eine triviale Implementierung der Suche nach $\vec{t}$ könnte über alle möglichen Pfade iterieren, $G(\vec{t})$ berechnen, und das $\vec{t}$ mit der besten Bewertung als Ergebnis liefern. Der Aufwand für diesen Algorithmus würde allerdings exponentiell mit der Länge des untersuchten Sprachsignals wachsen. Aus diesem Grund haben wir

die Funktion G so entworfen, daß sie mit dem Verfahren der dynamischen Programmierung (DP) [Bel57] in linearer Zeit ausgewertet werden kann.
Dazu erhält der Auswertungsschritt als Eingabe das Ergebnis der Kurzzeitanalyse (=Spalte im Cepstrogramm) für jeden Frame im Eingabesignal. Das Ergebnis für Frame $j \in \{0,1,...,j_{max}\}$ wird in die in der Cepstrogrammatrix $M_{i,j}$ (siehe Kapitel 4.4) in Spalte j eingetragen. Die Zeilenanzahl $i_{max}+1=f_{neuabtast}/F0_{min}$ dieser Matrix entspricht damit der Anzahl der berechneten Cepstrumkoeffizienten, wobei der i-te Cepstrumkoeffizient ($i \in \{0,1,...,i_{max}\}$) der Abtastwert der Kurzzeitanalysefunktion bei der Periodendauer $t=i \cdot T0_{max}/i_{max}$ ist. Die Spaltenanzahl $j_{max}+1$ der Matrix entspricht der Anzahl der Frames im Signal.
Wir zeigen jetzt die Zerlegung des Problems der Auswahl des optimalen Pfades $\vec{t}$ in leichter lösbare Teilprobleme mit Hilfe der dynamischen Programmierung. Zur Erinnerung zeigen wir erst noch einmal die Gewinnfunktion (Gleichung 4.3) und das Pfadauswahlkriterium (Gleichung 4.5) aus Kapitel 4.4:

$$G(\vec{t}) = \sum_{j=0}^{j_{max}} M_{t_j,j}$$

$$\vec{t} = \underset{\vec{t}' \in P}{\operatorname{argmax}}\, G(\vec{t}')$$

$$\text{mit } P = \left\{ \vec{t}' \in \{0,1,...,i_{max}\}^{j_{max}+1} \mid d(t_j, t_{j+1}) < d_{max},\ j \in \{0,1,...,j_{max}-1\} \right\}$$

Zur Zerlegung des Problems der Pfadsuche wird ein Pfad als Folge von Stufen (=Frames) betrachtet, das Vorliegen einer bestimmten Grundperiodendauer in einem Frame als Zustand, und die Änderung der Grundfrequenz von einem Frame zum nächsten als Aktion. Das Optimierungsproblem besteht damit aus:

- die Stufenmenge J legt die Anzahl der Stufen fest. $J=\{0,1,2,...,j_{max}\}$
- der Zustandsraum S ist die Menge der Zustände, also die Menge der im Problem vorkommenden Grundperiodendauern. $S=\{0,1,2,...,i_{max}\}$
- der Aktionenraum $A_n(s)$ ist die Menge der auf Stufe n im Zustand s möglichen Aktionen, also die möglichen Folgegrundfrequenzwerte:
  $A_0(s) = \{ 0,1,...,i_{max} \}$
  $A_n(s) = \{ i \text{ aus } \{0,1,...,i_{max}\} \mid d_{oktave}(s,i) \le d_{max} \}$
  $= \{ i \text{ aus } \{0,1,..., i_{max} \} \mid |\log_2(s/i)| \le d_{max} \}$
  $= \{ i \text{ aus } \{0,1,..., i_{max} \} \mid \max(s,i) / \min(s,i) \le 2^{d_{max}} \}$
  $= \{ i \text{ aus } \{0,1,..., i_{max} \} \mid i / 2^{d_{max}} \le s \le i \cdot 2^{d_{max}} \}$
  für $n>0$
- die Zustandstransformation $z(s,a)$ ordnet jedem Zustand s und einer möglichen Aktion $a \in A(s)$ den neuen Zustand in der nächsten Stufe zu: $z(s,a)=a$
- die einstufige Gewinnfunktion $r_n(s,a)$ gibt den in Stufe n erzielten einstufigen Gewinn an, wenn in Stufe n der Zustand s vorliegt, und Aktion a (=Folgezustand) gewählt wird: $r_n(s,a)=M_{a,n}$
- Eine initiale Gewinnfunktion $V_0(s)$, die den initialen Gewinn im Zustand s auf Stufe 0 angibt: $V_0(s)=0 \ \forall s \in S$

Damit kann die Wahl der Zustandsfolge zur Erzielung des maximalen Gesamtgewinns auf die Lösung der Bellmannschen Optimalitätsgleichungen [Bel57] zurückgeführt werden. Somit ist

$$V_n(s)=\max \{ V_{n-1}(s')+r_n(s',a) \mid s'\in S, a\in A_n(s'), s=z(s',a) \} \quad (4.12)$$

der maximale Gesamtgewinn auf den Stufen 0,...,*n* in Abhängigkeit vom Zustand *s* auf Stufe *n*. Der Zustand *s'* auf Stufe *n*-1 ist der Vorgängerzustand von Zustand *s* auf Stufe *n* auf der optimalen Zustandsfolge $\vec{s}$.
Zur Implementierung der oberen Gleichung wird über die obere Formel $V_n(s)$ stufenweise für alle Zustände *s* aus $V_{n-1}(s')$ berechnet. Der optimale Vorgängerzustand *s'* von Zustand *s* auf Stufe *n* wird in einer Matrix *P* mit $P_{s,n}=s'$ gespeichert. Vorher berechnete Gewinne $V_{n-1}(s')$ werden dann für die folgenden Stufen nicht mehr benötigt, und können ‚vergessen' beziehungsweise überschrieben werden. Der optimale Pfad durch die Matrix $M_{i,j}$ wird erst nach der Berechnung des Gewinns für $V_n(s)$ für die letzte Stufe (=Spalte im Cepstrogramm) mit $n=j_{max}$ ermittelt: der letze Zustand *s** auf diesem optimalen Pfad ist

$$s^* = \underset{s\in S}{\operatorname{argmax}} V_{j_{max}}(s) \quad (4.13)$$

Alle Vorgängerzustände auf diesem optimalen Pfad werden durch Rückverfolgung (backtracking) mit Hilfe der Vorgängermatrix *P* bestimmt: der Vorgängerzustand *s'* von Zustand *s* auf Stufe *n* ist $s'=P_{s,n}$.

Wir wollten eine schritthaltende Variante des Algorithmus mit einer Vorausschau von $t_{vorausschau}$ implementieren, die oberen Betrachtungen gelten nur für den Fall, daß $t_{vorausschau}$ größer als die Dauer des Eingabesignals ist. Das liegt daran, daß die optimale Zustandsfolge $\vec{s}$ ($=\vec{t}$) erst bei der Verarbeitung des letzten Frames (=Stufe j_{max}) festgestellt werden kann. Bei einer begrenzten Vorausschau auf die folgenden $v= t_{vorausschauf} \cdot f_{ramerate}$ Frames muß der Zustand s_j aber spätestens nach der Verarbeitung von Frame *j*+*v* feststehen, und darf später nicht mehr verändert werden. Die Implementierung der schritthaltenden Pfadselektion gemäß der Gleichung

$$\vec{t} = \begin{pmatrix} t_1 \\ \dots \\ t_{j_{max}} \end{pmatrix} \text{ mit } t_j = t_j^* \text{ und } \vec{t}_j^* = \underset{\vec{t}'\in P_{j+v}}{\operatorname{argmax}} G(\vec{t}') \ \forall j \quad (4.14)$$

aus Kapitel 4.8 unterscheidet sich nur minimal von der oben angegebenen nichtschritthaltenden Implementierung: die Größe der Matrix *P* der optimalen Vorgängerzustände wird auf die Einträge für alle Stufen (=Frames) die innerhalb der Vorausschau liegen begrenzt, also auf *v* Spalten. Nach der Verarbeitung von Frame *j*+*v* wird mittels eines auf *v* Stufen begrenzten Backtrackings der (bis hier) optimale Zustand für alle Frames aus dem Intervall [*j*,*j*+*v*] gesucht. Der Zustand (=Grundperiodendauer) in Stufe (=Frame) *j* wird auch bei der Verarbeitung zukünftiger Frames nicht

mehr geändert, da sonst die Bedingung der begrenzten Vorausschau verletzt werden würden. Zustände in den Frames ab einschließlich j+1 können sich nach der Verarbeitung von zukünftigen Frames noch verändern.
Bei dieser Implementierung muß das Backtracking nicht in jedem Fall über alle v Stufen innerhalb der Vorausschau erfolgen: das Backtracking kann bei derjenigen Stufe abgebrochen werden, bei der sich keine Zustandsänderung im Vergleich zum Backtracking nach der Verarbeitung der Vorgängerstufe ergibt. Die Matrix P der optimalen Vorgängerzustände kann effizient als Ringpuffer implementiert werden.

4.9.5 Feinauflösende Suche

Das Ergebnis des vorigen Schrittes ist eine mit $f_{\text{framerate}}$ abgetastete Grundfrequenzkontur. Die Grundperiodendauern können dabei je nach vom Anwender gewählter Neuabtastrate $f_{\text{neuabtast}}$ relativ grob geschätzt sein: die Werte liegen auf einem Raster mit einem Gitterabstand von $f_{\text{eingabe}}/f_{\text{neuabtast}}$ Samples (Abtastwerte im Eingabesignal).
Der erste Schritt besteht in einer Neuabtastung (Upsampling) der Grundfrequenzkontur mit der gewünschten Ausgabeabtastrate f_{ausgabe}. Die Zwischenwerte werden dabei mittels linearer Interpolation bestimmt (siehe Kapitel 4.7.2).
Im nächsten Schritt wird der grobe T_0-Schätzwert T auf jedem Frame durch Berechnung eines hochauflösenden Ausschnittes der Kurzzeitautokorrelation $\varphi_n(k)$ verfeinert (siehe Kapitel 4.7.1). Dieser Ausschnitt umfaßt die Werte aus dem Intervall $[T-d/2,T+d/2]$, wobei der Downsamplingfaktor $_d$ sich aus $d=f_{\text{abtast}}/f_{\text{neuabtast}}$ berechnet. Die Berechnung erfolgt dabei mit der durch die Abtastrate f_{eingabe} der Eingabesignal gegebenen Auflösung auf ein Sample genau. Da die Korrelation auf dem tiefpaßgefilterten Eingabesignal berechnet wird, wird wie die Berechnung nicht gemäß

$$\varphi_n(k) = \sum_{m=0}^{N-1-k} x(n+m)x(n+m+k) \tag{4.15}$$

implementiert, sondern mit einer groben Abtastrate von ≥2000 Hz über

$$\varphi_n(k) = \sum_{m=0}^{\left\lfloor \frac{N-1-k}{v} \right\rfloor} x(n+m\cdot v)x(n+m\cdot v+k) \quad \text{mit } v = \lfloor f_{abtast} / 2000Hz \rfloor \tag{4.16}$$

4.9.6 Konturfilterung

Da eine nachfolgende Glättung der erzeugten Grundfrequenzkontur mit einem Mittelwertfilter der Breite 3 eine leichte Verbesserung der Grobfehlerrate auf der SPONTAN-Stichprobe brachte (Kapitel 4.5), wurde dieser Schritt als optionale letzte Verarbeitungsstufe implementiert.

4.10 Robustheit gegenüber Störungen

Ein wichtiges Kriterium der Güte eines Meßverfahrens – wie unseres Grundfrequenzalgorithmus – ist seine Robustheit gegenüber verschiedenen Störeinflüssen. Zur Messung der Robustheit haben wir deshalb die Sprachsignale der SPONTAN-Stichproben mit verschiedenen Arten von Störungen kontaminiert.

4.10.1 Rauschen

Wir haben mehrere Experimente durchgeführt, die den Einfluß weißen Rauschens auf die Grobfehlerrate der SPONTAN-Stichprobe zeigen sollen. Dazu wurde zum ungestörten Originalsprachsignal s_n ein Störsignal e_n addiert. Die Abtastwerte des Störsignals wurden aus einer Gleichverteilung auf dem Intervall [-a,a] gezogen um weißes Rauschen zu erzeugen. Die folgende Tabelle zeigt die Grobfehlerrate auf der SPONTAN-Stichprobe, wobei der Signal-Rausch-Abstand r (in dB) zur Messung des Rauschens verwendet wurde:

$$r = 10 \log_{10} \frac{E(s_n^{\,2})}{E(e_n^{\,2})} \tag{4.17}$$

$E(f_n^2)$ ist dabei das zweite Moment des Signals f_n, das hier als Folge von Realisierungen einer Zufallsvariable aufgefaßt wird (ST95). Statt des zweiten Moments kann auch äquivalent die Energie der beiden Signale gemessen werden. Bei unserer Angabe von r nehmen wir vereinfachend an, daß das Eingabesignal s_n rauschfrei ist. Die für die Versuche benutzten Parameter sind $F0_{min}$=35 Hz, $f_{neubabtast}$=2250 Hz.

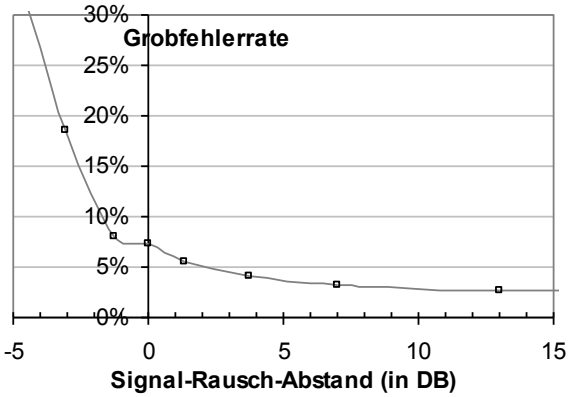

Abbildung 4.13 Einfluß additiven Rauschens auf die Grobfehlerrate

Leider stehen uns keine Fehlerraten anderer Grundfrequenzalgorithmen bei verschiedenen Signal-Rausch-Abständen auf der SPONTAN-Stichprobe zur Verfügung, so daß wir die Robustheit unseres Verfahrens schlecht mit anderen vergleichen können.

Die Analyse der Fehlerverteilung in der unteren Abbildung zeigt, daß bei stark verrauschten Signalen die Wahrscheinlichkeit zunimmt, Vielfache der Grundperiodendauer zu detektieren.

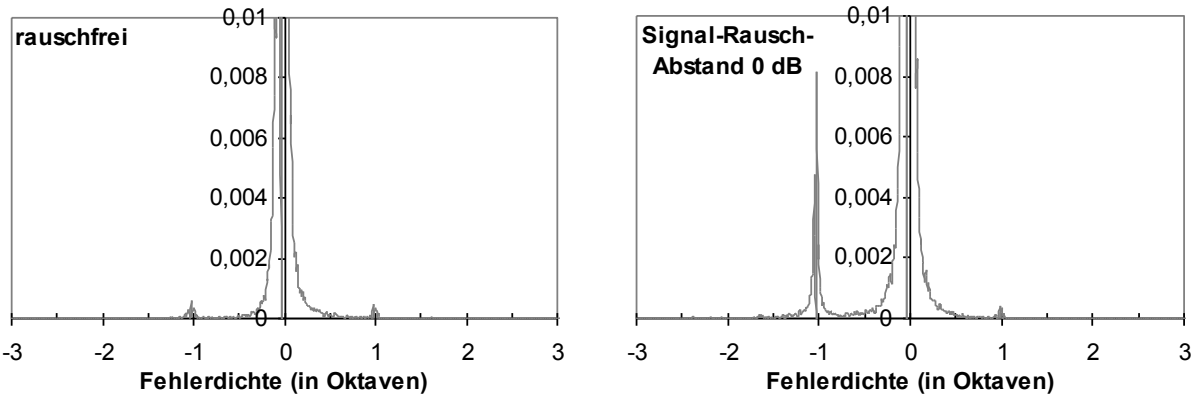

Abbildung 4.14 Fehlerdichte bei verrauschten Eingabesignalen

Bei stark verrauschten Signalen bleibt uns die Möglichkeit, das Rauschen im Signal mittels geeigneter Verfahren zu unterdrücken. Solche Verfahren sind beispielsweise spektrale Subtraktion (‚spectral subtraction') und Rauschmaskierung (‚noise masking'). Eine Einführung in eine Auswahl dieser Verfahren findet sich in [Gon95]. Die Auswirkung solcher Verfahren auf die Fehlerrate unseres Algorithmus haben wir nicht überprüft, da wir es in den von uns untersuchten Anwendungen nicht mit stark verrauschten Eingabesignale zu tun hatten. Eine spätere Erweiterung unseres Grundfrequenzalgorithmus sollte aber nicht nötig sein, da die Verfahren zur Rauschunterdrückung im Allgemeinen in die Vorverarbeitung integriert werden.

Falls es sich beim addierten Störsignal nicht um weißes Rauschen, sondern um ein quasiperiodisches Signal handelt, ist auch eine Verschlechterung der Leistung des Grundfrequenzverfahrens zu erwarten. Aufgrund der Vielzahl an möglichen Störsignalen haben wir auf eine quantitative Analyse dieser Einflüsse verzichtet.

4.10.2 Telefonqualität

Sprachsignale werden bei der Übertragung über herkömmliche analoge Telefonverbindungen mit Grenzfrequenzen von etwa 300 Hz und 3,4 kHz bandpaßgefiltert. Um den Einfluß dieser Signalstörung auf die Grobfehlerrate unseres Grundfrequenzverfahrens zu messen, haben wir die SPONTAN-Daten mit diesen Grenzfrequenzen gefiltert.
Die Evaluation unseres Algorithmus auf den gefilterten Signalen ergab eine starke Verschlechterung der Grobfehlerrate des Algorithmus von 2.3% (ungefiltert) auf 22.8%. Zur Fehleranalyse erzeugten wir wie schon in Kapitel 4.4 Diagramme, die Fehlerdichte der Abweichung zwischen Grundfrequenzhypothese und Referenz in

Oktaven darstellen. Im unteren Diagramm ist zu sehen, daß unser Algorithmus nach der Hochpaßfilterung viel öfter Vielfache der Referenzgrundfrequenz als Ergebnis liefert.

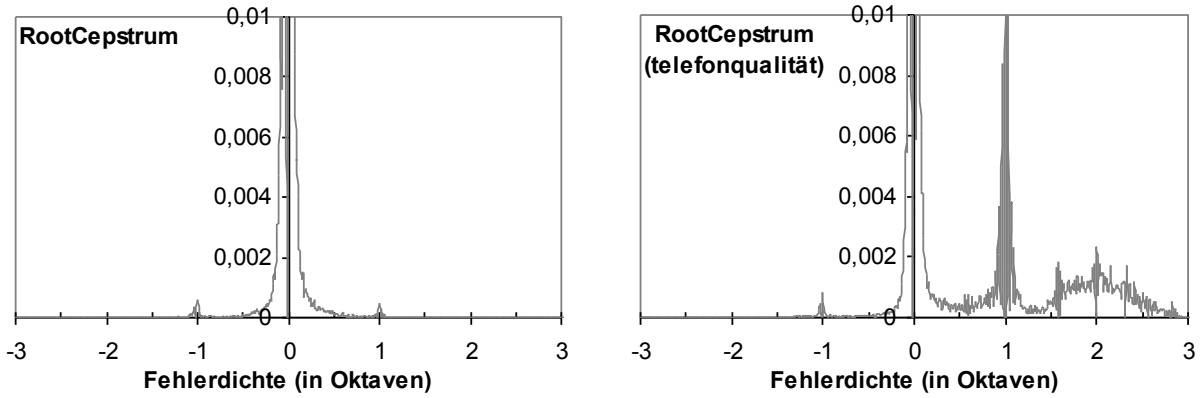

Abbildung 4.15 Fehlerdichten vor und nach der Hochpaßfilterung

Die folgenden beiden Diagramme zeigen wie hoch der Anteil der Frames mit fehlerhaft detektierter Grundfrequenz in Abhängigkeit der Referenzgrundfrequenz ist. Im linken Diagramm ist dieser Anteil auf den originalen (nicht hochpaßgefilterten) Eingabesignalen bestimmt worden: das zeigt, daß die Fehlerdichte relativ unabhängig von der vorliegenden Referenzgrundfrequenz ist. Im rechten Diagramm ist zu sehen, daß nach der Hochpaßfilterung mit 300 Hz der Fehleranteil besonders bei niedrigen Grundfrequenzen stark ansteigt: bei einer Referenzgrundfrequenz von 150 Hz werden etwa in 50% der Frames fehlerhafte Grundfrequenzwerte berechnet, bei noch tieferen Grundfrequenzen steigt der Anteil der fehlerhaften Frames weiter an.

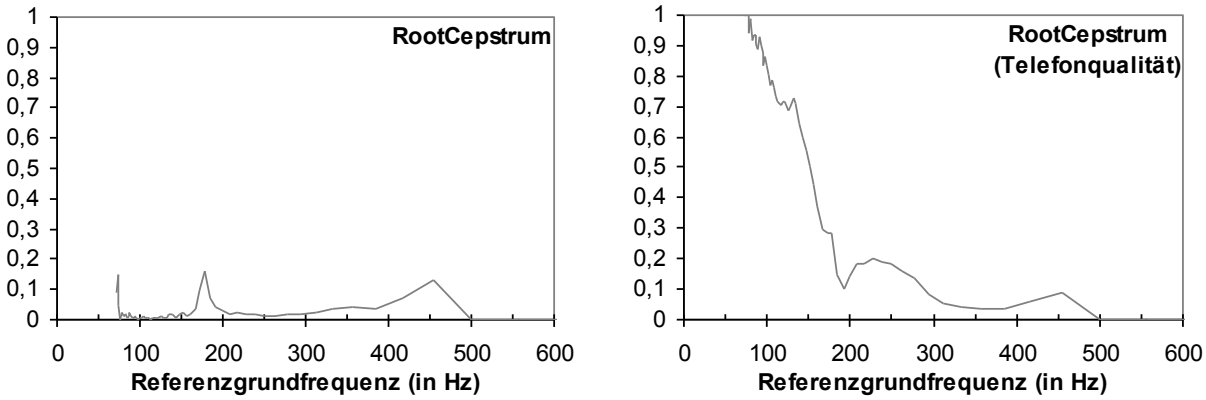

Abbildung 4.16 Anteil der fehlerhaften Frames in Abhängigkeit von der Referenzgrundfrequenz

Bei der Kurzzeitanalyse mittels Kreuz- und Autokorrelation ergeben sich auf den mit 300 Hz hochpaßgefilterten Signalen ähnliche Fehlerdichten und Grobfehlerraten, wobei die normierte Kreuzkorrelation mit 16% Fehlerrate etwas bessere Ergebnisse liefert als Rootcepstrum und Autokorrelation.

Zusammenfassend müssen wir feststellen, daß unser Grundfrequenzalgorithmus für die Analyse von stark hochpaßgefilterten Sprachsignalen – wie beispielsweise Telefonsprache – nicht geeignet ist. Die hohe Fehlerrate führen wir darauf zurück, daß durch die Hochpaßfilterung die lokalen Maxima (‚pitch peaks') im Zeitsignal besonders bei Sprachsignalen mit niedriger Grundfrequenz stark unterdrückt werden. Das führt im Extremfall dazu, daß die Grundperioden im hochpaßgefilterten Signal von einem menschlichen Betrachter nicht mehr mit bloßem Auge identifiziert werden können. Wie unser Algorithmus im Vergleich zu anderen Grundfrequenzalgorithmen abschneidet können wir nicht sagen, da uns die dazu notwendigen Ergebnisse anderer Grundfrequenzalgorithmen nicht zur Verfügung stehen.

4.11 Anschließende Berechnungen

An die Berechnung der Grundfrequenzkontur können sich abhängig von der nachgeschalteten Anwendung noch weitere verwandte Berechnungen im Zusammenhang mit der Grundfrequenzbestimmung anschließen. Dazu zählen die Bestimmung der mittleren Grundperiodendauer für eine ganze Äußerung, die stimmhaft/stimmlos-Klassifikation (SH/SL) und die Markierung von Grundperiodengrenzen.

4.11.1 SH/SL-Klassifikation

Die SH/SL-Klassifikation kann mit den in Abschnitt 3.4 angegebenen Verfahren und Merkmalen durchgeführt werden. Dabei ist unserer Meinung nach als zusätzliches Merkmal die normierte Kreuzkorrelation zweier aufeinanderfolgender Grundperioden empfehlenswert, wobei die Grundperiodendauer für die SH/SL-Klassifikation mit Hilfe unseres F_0-Detektors bestimmt werden kann. Zwar wurde für keine der untersuchten Anwendungen eine harte SH/SL-Klassifikation benötigt. Wir benötigten aber einen SH/SL-Klassifikator für den Vergleich unseres Grundfrequenzverfahrens mit den anderen auf der SPONTAN-Stichprobe untersuchten Verfahren. Da der Klassifikator nur für diesen Zweck eingesetzt werden sollte, mußte er nur zwei Anforderungen erfüllen: die Fehlerrate sollte wie bei den Verfahren in Tabelle 4.12 etwa 11% betragen, und er sollte möglichst einfach zu implementieren und zu optimieren sein. Außerdem sollten Signalabschnitte in denen nicht gesprochen wird, also Stille, als SL klassifiziert werden.
Aus der Anforderung eine einfache Implementierung und Optimierung folgte, daß möglichst ein eindimensionaler Merkmalsvektor zusammen mit einem Schwellwertverfahren als Klassifikator genügen sollte. In [Rab78] werden die innerhalb des Kurzzeitfensters k gemessene Energie E_k und die Nulldurchgangsrate N_k als Merkmale für einen SH/SL-Klassifikator vorgeschlagen:

$$E_k = \sum_{m=0}^{N-1} x_k(m)^2 \tag{4.18}$$

$$N_k = \left|\left\{m \in \{1,2,...,N-1\} \mid \mathrm{sgn}(x_k(m-1)) \neq \mathrm{sgn}(x_k(m))\right\}\right| \tag{4.19}$$

Hier ist $x_k(0)$ ist der Anfang des *k*-ten Kurzzeitfensters. Als Fensterbreite *N* wurde 1/50 s gewählt. Der Autor erwähnt dabei, daß die Nulldurchgangsrate nicht auf dem tiefpaßgefilterten Signal gemessen werden soll. Da die Äußerungen der SPONTAN-Stichprobe in der vorliegenden Form bereits mit einer Grenzfrequenz von 4.5 kHz tiefpaßgefiltert sind, liefert die Nulldurchgangsrate keinen sinnvollen Beitrag zu Unterscheidung von stimmhaften und stimmlosen Frames. Mit der Energie als einzigem Merkmal im Merkmalsvektor erreichten wir eine akzeptable SH/SL-Fehlerrate von 12.0%. Zusätzlich testeten wir noch zwei weitere Merkmale für die SH/SL-Klassifikation: den maximalen Betrag A_k der Amplitude im Kurzzeitfenster, sowie den normierten Kreuzkorrelationskoeffizient K_k zwischen zwei um T_0 versetzten Kurzzeitfenstern:

$$A_k = \max\{x_k(m) \mid 0 \le k \le N-1\} \tag{4.20}$$

$$K_k = \frac{\sum_{m=0}^{N-1} x_k(m)x_k(m+T_0)}{\sum_{m=0}^{N-1} x_k(m)^2 \cdot \sum_{m=0}^{N-1} x_k(m+T_0)^2} \tag{4.21}$$

Mit Hilfe des Merkmals A_k wurde mit 12.8% nur eine etwas schlechtere SH/SL-Fehlerrate als mit der Kurzzeitenergie E_k erreicht. Der normierte Kreuzkorrelationskoeffizient K_k brachte als einziges Element im Merkmalsvektor mit 22% nur eine sehr schlechte SH/SL-Fehlerrate. Der Grund dafür ist unserer Meinung nach, daß in Sprachabschnitten, in denen nicht gesprochen wird, zwangsläufig die Korrelation im Hintergrundstörsignal analysiert wird, so daß in diesen Bereichen K_k je nach Struktur dieses Störgeräuschs zwischen -1 und 1 schwankt. K_k sollte trotzdem einen Beitrag für die Unterscheidung von SH und SL leisten können, wenn es zusammen mit einer Energiemessung im Merkmalsvektor verwendet wird.
Die Verteilungsdichten der Klassen SH und SL für den besten auf einem eindimensionalen Merkmalsvektor, nämlich der Kurzzeitenergie E_k, basierenden SH/SL-Detektor sind in Abbildung 4.17 dargestellt.

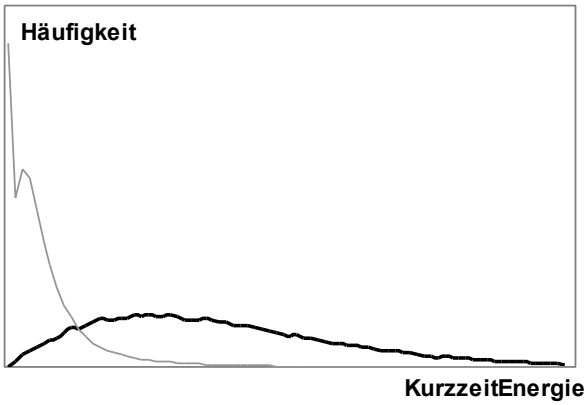

Abbildung 4.17 Histogramme über Kurzzeitenergie in SH (dunkel) und SL (hell)

Wie man in dieser Abbildung sehen kann, kann die Klassifikation mittels eines einfachen Schwellwertvergleiches durchgeführt werden. Die SH/SL-Fehlerrate dieses Klassifikators von 12.0% war für unsere Zwecke ausreichend, so daß wir keine weiteren Versuche mit höherdimensionalen Merkmalsvektoren durchgeführt haben.

4.11.2 Markierung von Grunderiodengrenzen

Auch die Grundperiodengrenzen - sie werden beispielsweise für die pitchsynchrone Analyse [Rab78] benötigt - könnten nach der verhältnismäßig zuverlässigen Berechnung des T_0-Verlaufs mit Hilfe unseres Grundfrequenzalgorithmus bestimmt werden. Da wir im Laufe dieser Arbeit mit keiner Anwendung gearbeitet haben, die solche Grundperiodengrenzen benötigt, und uns auch keine für die Bewertung eines solchen Markierungsverfahrens benötigte Referenzetikettierung zur Verfügung stand, haben wir in dieser Richtung keine Versuche durchgeführt.

4.11.3 Durchschnittliche Grundfrequenz

Für einige Anwendungen, wie beispielsweise die F0-VTLN, muß die mittlere Grundfrequenz beziehungsweise Grundperiodendauer einer Äußerung bestimmt werden. Dazu wird aufgrund des von unserem Grundfrequenzalgorithmus berechneten T_0-Hypothesenpfades $\vec{t}$ der Mittelwert $T0_{\text{avg}}$ der Grundperiodendauer über alle stimmhaften Frames der betrachteten Äußerung bestimmt:

$$T0_{avg} = \frac{1}{|S|}\sum_{k\in S} t_k \qquad (4.22)$$

mit S=$\{k\in\{0,1,...,j_{\max}\}$ und Frame k ist stimmhaft$\}$

S ist dabei die Menge der Indizes aller stimmhaften Frames im T_0-Hypothesenvektor $\vec{t}$ mit der Dimension $j_{\max}+1$, das heißt der Gesamtanzahl der Frames in der Äußerung. Mit der Mengenzugehörigkeitsfunktion $\sigma_S(k)$ mit

$$\sigma_S(k) = \begin{cases} 1, \text{falls } k \in S \\ 0, \text{sonst} \end{cases} \tag{4.23}$$

kann 4.22 auch folgendermaßen geschrieben werden:

$$T0_{avg} = \frac{\sum_{k=0}^{j_{\max}} t_k \cdot \sigma_S(k)}{\sum_{k=0}^{j_{\max}} \sigma_S(k)} \tag{4.24}$$

Das Problem ist, daß die Menge S der stimmhaften Frames unbekannt ist, weil wir auf eine harte SH/SL-Bestimmung im F_0-Detektor verzichten wollen (siehe Einleitung zu Kapitel 4). Eine offensichtliche Möglichkeit zur Bestimmung von S ist die Durchführung einer SH/SL-Klassifikation im Anschluß die Bestimmung des Grundfrequenzverlaufs. Die SH/SL-Klassifikatoren mit ihren mindestens 10% Fehlerrate (SPONTAN, siehe Tabelle 4.12) würden auch Ausreißer aus stimmlosen Regionen in die Berechnung von $T0_{\text{avg}}$ eingehen lassen und so das Ergebnis verfälschen. Aus diesem Grund stellten wir statt der scharfen Zugehörigkeitsfunktion $\sigma_S(k)$ eine (nichtnormierte) unscharfe Variante auf:

$$\sigma_S(k) = E_k \cdot \max(0, C_k) \tag{4.25}$$

Dabei ist E_k die Energie im Frame $k \in \{0,1,...,j_{\max}\}$, C_{k} ist der normierte Kreuzkorrelationskoeffizient zweier benachbarter Grundperioden der Länge t_{k} in Frame k. Hier wird ausgenutzt, daß die zu stimmhaften Frames gehörende normierte Kreuzkorrelationsfunktion ein Maximum bei T_0 mit einem Wert von ungefähr 1.0 aufweist, während die Korrelationsfunktion in stimmlosen Segmenten flach verläuft. Zusätzlich gewichten wir den Korrelationswert mit der Kurzzeitenergie E_{k} in diesem Frame, da stimmhafte Sprachsegmente meist lauter sind als stimmhafte.
Den Wert von $\sigma_S(k)$ haben wir in einem weiteren Experiment als einziges Merkmal für einen SH/SL-Klassifikator auf der SPONTAN-Stichprobe verwendet. Dabei erreichten wir eine SH/SL-Fehlerrate von etwa 14%.
Eine andere Möglichkeit zur Definition der unscharfen Zugehörigkeitsfunktion $\sigma_S(k)$ wäre, $\sigma_S(k)$ gleich der Ausgabe eines auf einem neuronalen Netz basierten SH/SL-Klassifikators zu setzen, oder analog gleich der Ausgabewahrscheinlichkeit eines Bayes-Klassifikators für die SH/SL-Entscheidung. Wir haben allerdings für unsere weiteren Experimente, speziell im Zusammenhang mit F0-VTLN (Kapitel 5.1.4), die mittlere Grundperiodendauer über die relativ einfachen Gleichungen 4.24 und 4.25 berechnet.

4.12 Ergebnisse und Vergleiche

Die folgende Tabelle vergleicht die Grobfehlerrate unseres F_0-Detektors auf dem SPONTAN-Korpus mit den Ergebnissen von anderen ebenfalls auf diesem Korpus getesteten Verfahren. Sie listet die Ergebnisse von zwei Konfiguration unseres Algo-

rithmus auf: die unseres langsamen aber genauen Verfahrens aus Kapitel 4.6 und die Ergebnisse des etwas ungenaueren, dafür aber schnelleren und schritthaltenden Verfahrens aus Kapitel 4.8. Die Ergebnisse der anderen Verfahren wurden aus [Kie96] entnommen.

Verfahren	*SH/SL-Fehlerrate*	*Grobfehlerrate*	*Echtzeitfaktor*
DPF0-SEQ[1]	10.8%	6.0%	?
ESPS[1]	10.8%	5.4%	?
DPF0-ITER[1]	10.5%	4.4%	?
unser Algorithmus	-	2.8%	1/200
unser Algorithmus	-	2.2%	1/6.7
unser Algorithmus	12.0%	2.0%	1/6.7

Tabelle 4.12 Grobfehlerraten einiger F_0-Verfahren (SPONTAN)

Laut [Kie96] sind viele andere ältere Verfahren zur Detektion von Grundfrequenzverläufen - wie zum Beispiel ADMF [Ros74] oder Seneff [Sen78] - den DPF0-Verfahren unterlegen.
Das zeigt, daß unser Algorithmus auf der SPONTAN-Stichprobe eine um mehr als 50% geringere Grobfehlerrate liefert, als die Algorithmen aus [Kie96]. Zu den Ergebnissen ist noch zu sagen, daß alle diese Algorithmen aus [Kie96] zur Berechnung ihrer Grobfehlerrate nur die Frames einbeziehen, die von sowohl in den Referenzdaten als auch vom vorgeschalteten SH/SL-Detektor als stimmhaft markiert wurden. Diese SH/SL-Fehlklassifikationrate liegt aber schon bei über 10%. Würden die Verfahren aus [Kie96] auf allen in der Referenz als stimmhaft etikettierten Frames getestet werden, würde sich ihre Grobfehlerrate höchstwahrscheinlich verschlechtern. Diese Hypothese wird von unserem in der letzten Tabellenzeile angegebenen Resultat gestützt: eigentlich verzichtet unser Verfahren auf eine der F_0-Verfolgung vorhergehende SH/SL-Detektion, so wir die Grobfehler auf allen in der Referenz als stimmhaft markierten Frames akkumulieren müssen. Beim Versuch in der letzten Tabellenzeile haben wir die Akkumulierung der Grobfehler nur auf die zusätzlich vom einfachen SH/SL-Detektor aus Kapitel 4.11.1 als stimmhaft markierten Frames beschränkt. Dadurch reduzierte sich die Grobfehlerrate um etwa 10% von 2,2% auf 2,0%.
Die Laufzeit unseres Verfahren konnten wir leider nicht mit denen aus [Kie96] vergleichen, weil uns diese Programme nicht zum Testen zur Verfügung standen. Wir wissen lediglich aus [Kie96] daß der DPF0-Algorithmus mit ungefähr 1/3 Echtzeit arbeitet, wobei aber nicht angegeben ist, auf welchem System dieser Wert gemessen wurde.

[1] nach [Kie96]

5 Anwendungen

5.1 Sprechernormierung

Eines der aktuellen Probleme beim Entwurf sprecherunabhängiger Spracherkenner ist die Sprecherabhängigkeit des Sprachsignals, welche zu einer schlechteren Erkennungsrate im Vergleich zu sprecherabhängigen Erkennern führt. Die Gründe für diese Sprecherabhängigkeit des Sprachsignals sind vielfältig: Dialekt und Vokaltraktform des Sprechers gehören zu den wichtigsten. Um den Einfluß der Vokaltraktform auf die aus dem Sprachsignal gewonnenen Merkmale zu reduzieren, kann eine sogenannte Vokaltraktlängennormierung (VTLN) auf dem Signal durchgeführt werden. Dazu wird im Spektralbereich eine Transformation der Frequenzachse vorgenommen, um über diesen Weg eine Verringerung der Varianz der Merkmalsvektoren innerhalb jeder Phonemklasse zu erreichen. Das bisher an der Universität Karlsruhe eingesetzte Verfahren basiert auf einer Maximum-Likelihood Schätzung der Vokaltraktlänge (ML-VTLN) auf cepstralen Merkmalen und war leider zu zeitaufwendig, um in Erkennern eingesetzt zu werden, die in Echtzeit arbeiten sollen. Wir stellen in dieser Arbeit ein schnelleres Verfahren vor, das für die Bestimmung des Normierungsparameters ausschließlich die mittlere Grundfrequenz einer Äußerung heranzieht. Die Bestimmung der Parameter für die Abbildung der mittleren Grundfrequenz auf den Normierungsparameter basiert dabei auf einer Maximum-Likelihood Schätzung.

5.1.1 Vokaltraktlängennormierung VTLN

Einer der wichtigsten Gründe für die Sprecherabhängigkeit des Sprachsignals ist die Vokaltraktlänge des Sprechers. Männer haben eine durchschnittliche Vokaltraktlänge von etwa 18 cm, bei Frauen liegt dieser Wert im Mittel bei 13 cm. Diese Unterschiede in den Vokaltraktlängen der Sprecher führen dazu, daß Formanten, also die Resonanzfrequenzen im Spektrum, bei Frauen durchschnittlich 20% höher liegen, als bei Männern. Diese Verschiebung hat einen negativen Einfluß auf die Erkennungsrate des sprecherunabhängigen Spracherkenners, weil damit die 'Ähnlichkeit' der Spektren und damit der Merkmalsvektoren innerhalb von ein und derselben (Sub-)Phonemklasse verringert wird. Das heißt, die Merkmalsvektoren eines bestimmten Phonems, das von verschiedenen Sprechern stammt, liegen im Merkmalsraum relativ weit auseinander. In Abbildung 5.1 wird die Formantenverschiebung bei 3 Phonemklassen K_1, K_2 und K_3 dargestellt, wobei auf den Achsen die beiden Formanten F_1 und F_2 aufgetragen sind. Daß durch diese Verschiebung die Klassifizierung erschwert werden kann, zeigen die Phonemklassen K_2 und K_3: wenn nur ein einzelner Sprecher betrachtet wird, überlappen sich die beiden Klassen nicht, und sind somit leicht un-

terscheidbar. Wenn jedoch mehrere Sprecher betrachtet werden, kommt es zu Überlappungen der beiden Phonemklassen im mit X gekennzeichneten Bereich.

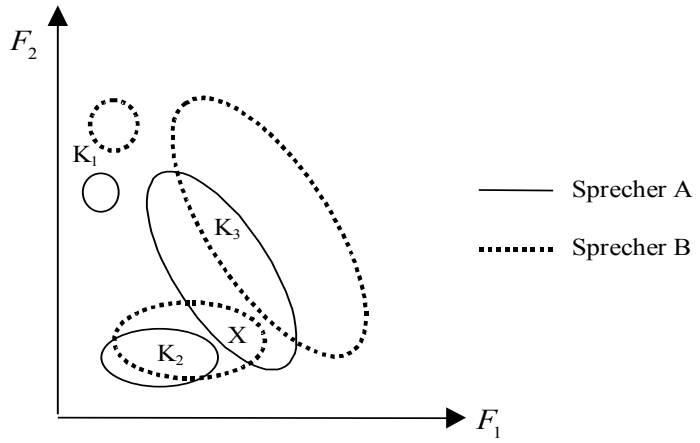

Abbildung 5.1 Phonemklassen zweier Beispielsprecher

Um dieses Manko teilweise zu kompensieren, müssen für eine ausreichende Modellierung der Klassenverteilungen im Merkmalsraum im Vergleich zu sprecherabhängigen Erkennern viel mehr Trainingsdaten aufgewendet werden.
Da sprecherunabhängige Erkenner alleine durch Vergrößerung der Trainingsdatenmenge aber noch nicht die Leistung vergleichbarer sprecherabhängiger Erkenner erreichen, ist es sinnvoll die Varianz innerhalb der einzelnen Phonemklassen (bzw. Subphonemklassen) durch eine geeignete Transformation zu reduzieren. Hierzu bietet sich die VTLN an. Diese nutzt die Tatsache aus, daß die Lage der Resonanzfrequenzen im Spektrum bei ein und demselben Phonem annähernd umgekehrt proportional zur Länge des Vokaltraktes des Sprechers ist. Um mit diesem Wissen die Sprachsignale verschiedener Sprecher zu normieren, wird auf einem Ausschnitt des Sprachsignals ein zur Vokaltraktlänge des Sprechers korrelierter Parameter geschätzt, mit dem das Spektrum gestreckt beziehungsweise gestaucht wird. Dieser Parameter wird folgend in Anlehnung an die englische Literatur Warpingfaktor genannt.
Sowohl für die Schätzung des Warpingfaktors basierend auf dem Sprachsignal, als auch für die Normierung des Spektrums existieren mehrere Verfahren. In [Ei96] wurden 2 Klassen von Warpingfunktionen zur Normierung des Spektrums getestet: lineare und nichtlineare.

$$f' = k_s f \qquad \text{(linear)} \tag{5.1}$$

$$f' = k_s^{3f/8000} f \qquad \text{(nichtlinear)} \tag{5.2}$$

Diese Funktionen wurden in [Ei96] aus einfachen Vokaltraktmodellen abgeleitet: dem 'uniform tube model' und dem Helmholtz-Resonator-Modell. Tests mit einem HMM-Erkenner in diesem Artikel zeigten eine geringfügige Überlegenheit der nichtlinearen Warpingfunktion. Mit ihr wurden 57.1% Wortfehlerrate erreicht, gegenüber

57.3% mit linearem Warping. Eine weitere stückweise lineare Warpingfunktion wurde in [Weg96] vorgestellt und in [ZW97] mit der nichtlinearen Version verglichen:

$$f' = \begin{cases} \alpha_s^{-1} f & , f < F \\ bf + c & , f \geq F \end{cases} \tag{5.3}$$

$$f' = \begin{cases} \alpha_s^{-3f/8000\text{Hz}} f & , f < F \\ bf + c & , f \geq F \end{cases} \tag{5.4}$$

wobei α_s^{-1} der Warpingfaktor für den jeweiligen Sprecher ist. b und c sind Konstanten, die gemäß $\alpha_s F = bF + c$ und $8000b + c = 8000$ berechnet werden. F ist die konstante Schranke, an der die 2 linearen Teilstücke der Warpingfunktion zusammentreffen, im Test wurde sie auf 5600Hz gesetzt. Im Vergleich mit der nichtlinearen Warpingfunktion zeigte die stückweise lineare in [ZW97] bessere Erkennungsraten. Aus diesem Grund haben wir im Rahmen dieser Arbeit diese Warpingfunktion für unsere Experimente gewählt.

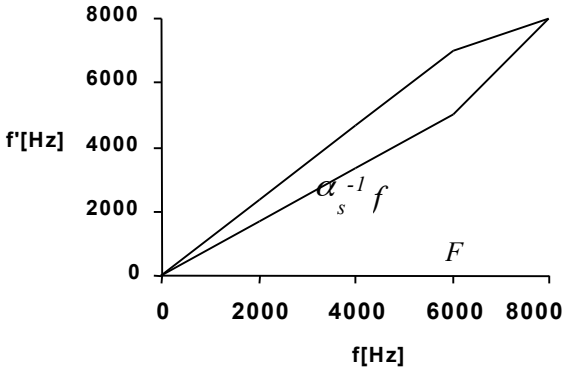

Abbildung 5.2 Die stückweise lineare Warpingfunktion

Wie bereits erwähnt, gibt es nicht nur für die Art und Weise der Verzerrung des Spektrums, sondern auch für die Wahl der Warpingparameter verschiedene Methoden. Dabei lassen sich diese Methoden in drei Gruppen einteilen: formantenbasierte, pitchbasierte und ML-basierte (Maximum Likelihood) Verfahren.
Die Methoden aus der erstgenannten Gruppe bestimmen die durchschnittliche Formantenfrequenz f für F_1,F_2 oder F_3 auf einem genügend großen Ausschnitt einer Äußerung (meist mehrere Sätze). Außerdem wird der durchschnittliche Wert f^* für diese Formantenfrequenz über alle Sprecher aus dem Korpus bestimmt. Die Spektren für diese Äußerung, bzw. diesen Sprecher, werden dann genau so verzerrt, daß die betrachtete Formante, die im unverzerrten Spektrum im Mittel bei f liegt, im verzerrten Spektrum im Mittel mit f^* zusammenfällt. Beispielsweise würde der Warpingfaktor bei Benutzung der linearen Warpingfunktion mit dieser Methode gleich f / f^* gesetzt werden. Genaueres zu dieser Methode wird in [ZW97] beschrieben.
Die pitchbasierten Verfahren machen sich die Korrelation zwischen Vokaltraktlänge und durchschnittlicher Stimmbandfrequenz zunutze: jemand, der eine hohe durch-

schnittliche Stimmbandfrequenz besitzt, hat im Allgemeinen auch einen kleinen Vokaltrakt. Deshalb bestimmen diese Verfahren, ähnlich wie die formantenbasierten, auf einem genügend großen Ausschnitt einer Äußerung (wieder meist mehrere Sätze) den durchschnittlichen Wert für die Stimmbandfrequenz F_0, und ermitteln mit Hilfe einer vorher parametrisierten Abbildung den optimalen Warpingfaktor für diesen F_0 Wert. Diese Abbildung kann angelegt werden, indem auf der gesamten Trainingsmenge für jeden Sprecher der durchschnittliche F_0 Wert bestimmt wird, und dazu der Warpingfaktor, der für diesen Sprecher die beste Erkennungsrate gebracht hat. Aus dieser Information lassen sich die Parameter der Abbildung durch ein geeignetes Verfahren (z.B. Regression) bestimmen. Eine detailliertere Beschreibung dieser Methode kann [Dai97] und dem folgenden Kapitel entnommen werden. Eine der F0-VTLN sehr ähnliche Methode namens ‚STRAIGHT-TEMPO morphing' wird in [Gir98] beschrieben.
Die dritte und letzte Gruppe bilden die ML-basierten Verfahren. Diese Verfahren können im Gegensatz zu den beiden erstgenannten nur mit Hilfe des akustischen Modells Λ des Erkenners benutzt werden. Es wird derjenige Faktor α_s zum Strecken des Spektrums benutzt, der die Wahrscheinlichkeit für die Beobachtung einer Äußerung maximiert:

$$\alpha_s^* = argmax_\alpha P(X(\alpha)|\Lambda, W) \tag{5.5}$$

Dabei ist $X(\alpha)$ die Folge der Merkmalsvektoren der Äußerung, deren Spektren mit Faktor α verzerrt wurden. W ist die zugehörige Transkription.

Vergleiche von Formanten und ML-Verfahren finden sich in [ZW97], wo alternativ die Formante F_1, F_2 oder F_3 zur Bestimmung des Normierungsparameters benutzt wurde (siehe folgende Tabelle). In [Dai97] wurden pitchbasierte und ML-Verfahren verglichen (Tabelle 2). Dabei hat sich eine Überlegenheit der ML-Verfahren gegenüber pitch- und formantenbasierten Verfahren gezeigt. Ein Nachteil der ML-Verfahren im Vergleich zu den beiden anderen Verfahren ist ihr höherer Zeitaufwand.

Modes	Basis	F1	F2	F3	ML
Linear	21.8%	20.5%	21.9%	21.6%	19.8%
Nonlinear	21.8%	21.5%	22.7%	21.6%	21.0%

Tabelle 5.1 Wortfehlerraten von formanten- und ML basierten VTLN [ZW97]

Basis	F0	ML
26.1%	24.4%	24.0%

Tabelle 5.2 Wortfehlerraten von pitch- und ML basierten Verfahren [Dai97]

5.1.2 Integration der VTLN in die Vorverarbeitung

Ziel der Vorverarbeitung ist es, das analoge Sprachsignal in eine Folge von Merkmalsvektoren zu transformieren. Diese Merkmalsvektoren sollen möglichst alle für die Spracherkennung relevanten Informationen enthalten, und dennoch niedrigdimensional genug sein, um die Erkennung in vertretbarer Zeit durchführen zu können.
Die Vorverarbeitungsstufe erhält als Eingabe ein analoges Sprachsignal. Dieses Signal muß zunächst zu äquidistanten Zeitpunkten abgetastet werden ('sampling'), was eine Folge von Amplitudenwerten liefert. Nach dem Nyquist-Theorem kann aus dieser Folge das analoge Signal reproduziert werden, wenn die Abtastrate mindestens doppelt so groß ist, wie die größte im analogen Signal vorkommende Frequenz. Da Frequenzanteile, die über der doppelten Abtastfrequenz liegen, im Signal für Verzerrungen sorgen ('aliasing'), muß vor der Abtastung noch eine Tiefpaßfilterung mit dieser Grenzfrequenz durchgeführt werden.
Aus der so erhaltenen Amplitudenwertfolge werden durch Multiplikation mit einer geeigneten Fensterfunktion alle 10 ms Blöcke von etwa 15 bis 20 ms Länge ausgeschnitten. Bei den gebräuchlichen Abtastraten von 8 bis 20 kHz entspricht das meistens einer Blockgröße von 256 Amplitudenwerten. Für die Weiterverarbeitung dieser Blöcke ('frames') existieren viele verschiedene Methoden, wobei Fouriertransformation und Lineare Vorhersage (‚linear prediction') zu den meistbenutzten zählen. Im letzten Schritt wird der Merkmalsvektor mittels einer LDA-Transformation auf einen niedriger dimensionalen Merkmalsvektor reduziert, der schließlich das Ergebnis der Vorverarbeitungsstufe darstellt.

Abbildung 5.2 Vorverarbeitung ohne VTLN

Die VTLN wird, wie in der Einleitung bereits erwähnt wurde, durch eine Transformation der Frequenzachse vorgenommen. Dabei gibt es mehrere Möglichkeiten diesen Schritt in die Vorverarbeitung zu integrieren:

1. Transformation des Spektrums direkt nach der Berechnung der Fouriertransformation. Diese Vorgehensweise ist in JANUS[2] implementiert, und wird auch in [Weg96] vorgeschlagen.
2. Modifikation der Abstände und der Breite der MEL-Filterbänke wird in [Lee96] vorgeschlagen. Vorteil gegenüber der erstgenannten Methode ist, daß der Rechenaufwand geringfügig verringert wird, wobei aber der Rechenaufwand für die Transformation schon in der ersten Methode im allge-

[2] Siehe Anhang A.1

meinen vernachlässigbar klein ist.
3. Abtastratenwandlung des Signals im Zeitbereich. Ein Nachteil dieser Methode ist, daß damit nur lineare Transformationen der Frequenzachse durchgeführt werden können.

In der unteren Abbildung wird die im JANUS verwendete Methode der Integration der VTLN in die Vorverarbeitung dargestellt.

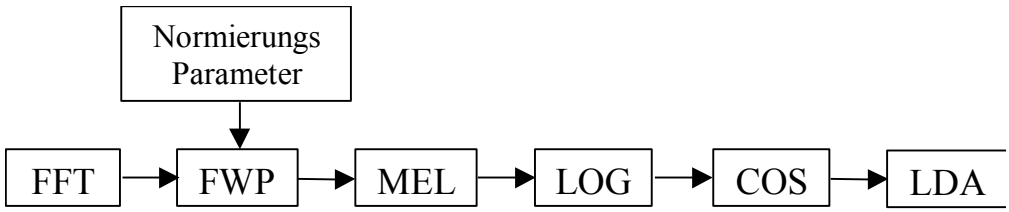

Abbildung 5.2 Vorverarbeitung mit VTLN

5.1.3 ML-VTLN

Im JANUS-Erkennungssystem, mit dem wir unsere Experimente durchführen konnten, ist bereits eine Maximum-Likelihood basierte VTLN integriert. Da wir die Ergebnisse unserer Experimente im Zusammenhang mit F0-VTLN mit der ML-VTLN vergleichen wollen, werden wir zunächst einmal kurz auf den Ablauf des Trainings und der Erkennung mit dem ML-VTLN basierten System eingehen. Auf die Grundlagen der Spracherkennung mit Hidden Markov Modellen (HMMs) wird an dieser Stelle nicht eingegangen. Eine detaillierte Einführung in dieses Verfahren findet sich beispielsweise in [ST95].

5.1.3.1 HMM Training mit ML-VTLN

Ziel der Trainingsprozedur mit VTLN ist es, die HMM Parameter über dem sprechernormierten Merkmalsraum zu bestimmen. Während des Trainings müssen jetzt nicht nur wie bei der Standardtrainingsprozedur ohne VTLN die Emissionswahrscheinlichkeiten der HMM-Zustände bestimmt werden, sondern zusätzlich noch der (im ML-Sinne) optimale Warpingfaktor für jeden Sprecher. Die so erweiterte Trainingsprozedur läuft folgendermaßen ab:

1. Initialisiere den Warpingfaktor für jeden Sprecher mit 1.0
2. Führe die Standard HMM-Trainingsprozedur (EM-Algorithmus) basierend auf den aktuellen Warpingfaktoren durch (z.B. mit Viterbi)
3. Ersetze den Warpingfaktor für jeden Sprecher jetzt durch den Faktor, der Gleichung

$$\alpha_s^* = argmax_\alpha P(X(\alpha)|\Lambda, W) \qquad (5.6)$$

erfüllt. W ist die Transkription des Sprachsignals X, und $X(\alpha)$ ist das mit α

gewarpte Signal.
4. Solange es signifikante Änderungen in den Warpingfaktoren gibt, gehe zu Schritt 2

Hauptproblem in dieser Prozedur ist die Lösung der Gleichung 5.6 im dritten Schritt. Da für die Lösung dieser Gleichung im allgemeinen keine geschlossene Form angegeben werden kann, wird α_s^* durch Suche auf einem Raster bestimmt. In [Lee96] wurde ein Raster vorgeschlagen, das 13 Werte im Bereich von 0.88 bis 1.12 im Abstand von 0.02 enthält. Dieser Bereich spiegelt die erwarte 25% Schwankung in der Vokaltraktlänge von Männern und Frauen wider. Die Rastersuche in Schritt 3 sieht dann in der Praxis so aus, daß die Merkmalsvektoren für jeden der 13 Warpingfaktoren im Raster bestimmt werden, und der ML-Score[1] entlang der schon für das HMM-Training berechneten Pfade berechnet wird.
Zu beachten ist noch, daß der Score nur auf den stimmhaften Phonemen bestimmt wird, weil dadurch eine bessere Erkennungsrate erreicht wird, also $\alpha_s^* = argmax_\alpha P(X(\alpha)|\Lambda,W,\sigma)$, wobei σ der Pfad ist. Eine Erklärung für diese Verbesserung ist, daß die Vokaltraktlänge sich hauptsächlich auf stimmhafte Laute, insbesondere Vokale, auswirkt und weniger auf stimmlose Phoneme wie Zischlaute. Die Unterscheidung zwischen stimmhaft und stimmlos ist sicher nicht optimal, wenn man bedenkt, daß auch Zischlaute stimmhaft ausgesprochen werden können und zum Beispiel das Phonem 'h' vor Vokalen zwar stimmlos ausgesprochen wird, aber trotzdem die Formantenstruktur des Folgevokals zeigt. Trotzdem ist dieses Unterscheidungskriterium gut genug, um eine Verbesserung der Erkennungsleistung zu bringen.

5.1.3.2 Erkennung mit ML-VTLN

Die VTLN-Erkennungsprozedur soll wie auch die ML-Erkennungsprozedur ohne VTLN die wahrscheinlichste Wortfolge für das gegebene Sprachsignal bestimmen. Bei Einsatz von VTLN muß zuvor allerdings der Warpingparameter des Sprechers geschätzt werden. Wie bereits im vorigen Kapitel erwähnt, wird derjenige Faktor α_s^* zum Strecken des Spektrums benutzt, der die Wahrscheinlichkeit für die Beobachtung dieser Äußerung maximiert:

$$\alpha_s^* = argmax_\alpha P(X(\alpha)|\Lambda,H(\alpha)) \qquad (5.7)$$

Die Lösung dieser Gleichung muß wie im Training durch eine Rastersuche approximiert werden. Im Gegensatz zum Training ist jetzt aber keine Transkription W der zu dekodierenden Äußerung gegeben. Infolgedessen muß während der Rastersuche für jeden Warpingfaktor α die HMM-Erkennungsprozedur auf dem mit diesem Warpingfaktor gestreckten Sprachsignal $X(\alpha)$ durchgeführt werden, um so die vom War-

[1] $-\log P(X|\Lambda)$

pingfaktor abhängige beste Hypothese $H(\alpha)$ zu finden. Unter den getesteten Warpingfaktoren wird dann der beste gemäß Gleichung 5.7 ausgewählt. Mittels dieses ML-VTLN-Verfahrens konnte die Erkennungsrate des Ausgangserkenners auf dem GSST (siehe Anhang A.1) von 83.9% ohne VTLN auf 85.0% mit VTLN gesteigert werden. Die mit der stückweise linearen Warpingfunktion theoretisch maximal erreichbare Erkennungsrate lag bei 87.7%. Es werden also ca. 30% aller durch diese VTLN-Methode korrigierbaren Fehler korrigiert. Die maximal erreichbare Erkennungsrate bestimmten wir, indem wir für jede Äußerung nicht das α_s^* auswählten, das den besten ML-Score (laut Gleichung 5.7) liefert, sondern dasjenige, mit dem die beste Erkennungsrate erzielt wird.

Basissystem (ohne VTLN)	83.9%
VTLN (13 Hypothesen)	84.9%
Theoretisches Optimum	87.7%

Tabelle 5.3 Verbesserung der Wortfehlerraten durch die VTLN

Die Implementierung des ML-VTLN-Verfahrens ist unkompliziert, problematisch ist jedoch der Anstieg des Zeitbedarfs für die Erkennung: wo bei einem vergleichbaren Erkenner ohne VTLN nur eine einzige Hypothese bestimmt werden muß, müssen beim Erkenner mit VTLN und der Benutzung eines Rasters mit üblicherweise 13 Warpingfaktoren auch 13 Hypothesen berechnet werden. Dies ist einer der Gründe dafür, daß dieses Verfahren in zeitkritischen Anwendungen noch nicht eingesetzt werden kann.
In [ZW97] konnte das Verfahren jedoch bedeutend beschleunigt werden, wobei die Leistung in bezug auf die Fehlerrate nur wenig verschlechtert wurde. Dabei wird nicht für jeden Warpingfaktor im Raster eine eigene Hypothese $H(\alpha)$ bestimmt, sondern der Erkenner bestimmt eine einzige Hypothese $H(\alpha=1.0)$ ohne Einsatz der VTLN, und bestimmt dann basierend auf dieser Hypothese wie schon im Training den ML-Score für jeden Warpingfaktor. Dadurch spart man sich die Durchführung der zeitaufwendigen Viterbi-Suche für jeden Warpingfaktor, nur die im Verhältnis dazu viel schnellere ML-Scoreberechnung muß noch für jeden einzelnen Faktor aus dem Suchraster durchgeführt werden.

Basissystem (ohne VTLN)	83.9%
Langsame VTLN (13 Hypothesen)	85.0%
Schnellere VTLN (nur noch eine Hypothese)	84.7%

Tabelle 5.4 Erkennungsraten mit beschleunigter VTLN

5.1.4 F0-VTLN

Die F0-VTLN nutzt den Zusammenhang zwischen der Vokaltraktlänge eines Sprechers und der mittleren Grundfrequenz in seinen Äußerungen aus: Männer haben im

Durchschnitt einen längeren Vokaltrakt und eine tiefere mittlere Grundfrequenz als Frauen. Der Sprechernormierungsparameter α_s für eine Äußerung wird nur aufgrund der auf dieser Äußerung bestimmten mittleren Grundfrequenz $F0_{avg}$ festgelegt.

$$\alpha_s = g(F0_{avg}) \tag{5.8}$$

Das Hauptproblem besteht damit in der Festlegung der Funktion g. Um Hinweise auf die Form der Abbildung g zu erhalten, haben wir zunächst einmal die mit Hilfe der ML-VTLN berechneten Normierungsparameter α_s und die mittlere Grundfrequenz $F0_{avg}$ für jede Äußerung aus unserer Trainingsstichprobe in das folgende Histogramm eintragen.

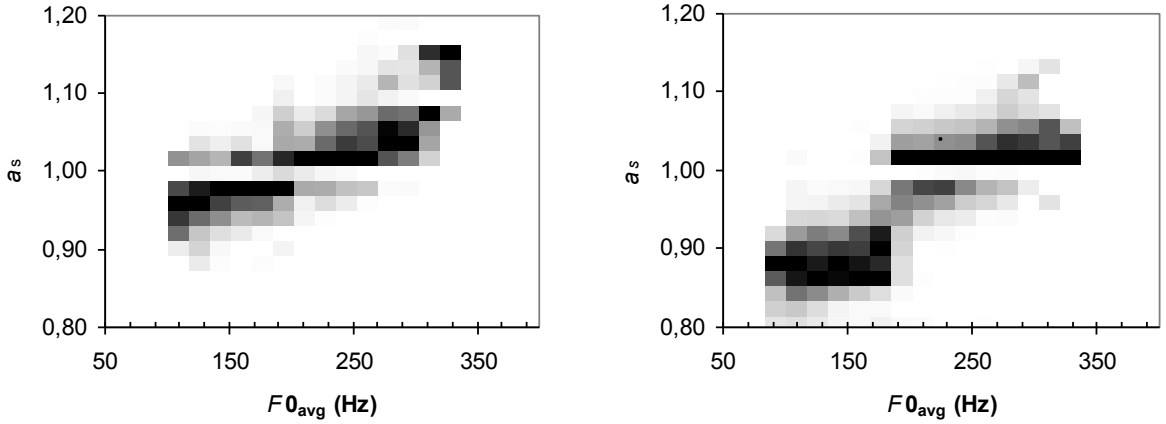

Abbildung 5.3 Normierungsparameter und mittlere Grundfrequenz (chin./deutsch)

Wir haben uns dafür entschieden, die Abbildung g in einer ersten Näherung durch eine Gerade zu modellieren, so daß das Problem der Bestimmung der Abbildung g auf das Problem der Suche nach den beiden Abbildungsparametern m und b reduziert wird:

$$g(F0_{avg}) = g_{m,b}(F0_{avg}) = m \cdot F0_{avg} + b \tag{5.9}$$

Die Modellparameter m und b können mit Hilfe der linearen Regression aus dem für jede Äußerung x der Trainingsmenge T vorliegenden Stichprobenvektor ($F0_{avg}(x)$, $\alpha_s(x)$) bestimmt werden. Dabei ist $F0_{avg}(x)$ die mittlere Grundfrequenz der Äußerung x, und $\alpha_s(x)$ ist der mit Hilfe der ML-VTLN bestimmte optimale Sprecher-normierungsparameter für diese Äußerung. Das Verfahren minimiert die Summe der Abstände der Vektoren (α_s,$F0_{avg}$) aus der Stichprobenmenge von der Modellfunktion $g_{m,b}$:

$$d_{m,b}(\alpha_s, F0_{avg}) = | g_{m,b}(F0_{avg}) - \alpha_s |^2 \tag{5.10}$$

$$(m^*,b^*) = \operatorname{argmin}_{(m,b)\in R^2} \sum_{x\in T} d_{m,b}(F0_{avg}(x), \alpha_s(x)) \tag{5.11}$$

Voraussetzung für die Anwendung der linearen Regression zusammen mit der Methode der kleinsten Quadrate ist, daß die Fehler $d_{m^*,b^*}(F0_{avg}(x),\alpha_s(x))$ normal-verteilt sind. Diese Voraussetzung ist sicher nur näherungsweise erfüllt, da in der

Grundfrequenzbestimmung $F0_{avg}(x)$ mit dem Auftreten von Oktavsprüngen, also Ausreißern, gerechnet werden muß.
Alternativ zur Bestimmung der Modellparameter *m* und *b* über die lineare Regression haben wir eine Schätzung mittels eines Maximum-Likelihood Ansatzes durchgeführt. Die Idee ist dabei, die Parameter *m* und *b* so zu wählen, daß die vom HMM berechnete Wahrscheinlichkeit für die Beobachtung der Äußerungen aus der Trainingsmenge *T* maximiert wird:

$$(m^*,b^*) = \arg\max_{(m,b)\in R^2} \left\{ \prod_{x\in T} P\left(X\left(g_{m,b}\left(F0_{avg}(x)\right)\right)|\,\Lambda,W\right) \right\} \tag{5.12}$$

W ist hierbei wieder die zur Äußerung *X* gehörende Transkription und Λ das Sprachmodell. Diese Maximierung ist äquivalent zur Minimierung der negierten logarithmierten Wahrscheinlichkeit (des Likelihood-Scores) $S(m,b)$ für diese Beobachtung:

$$\begin{aligned}(m^*,b^*) &= \arg\min_{(m,b)\in R^2} \left\{ S(m,b) := \sum_{x\in T} L_x\left(g_{m,b}\left(F0_{avg}(x)\right)\right) \right\} \\ &\text{mit } L_x(\alpha) = -\log\left(P\left(X(\alpha)|\,\Lambda,W\right)\right)\end{aligned} \tag{5.13}$$

Dabei ist $X(\alpha)$ wieder das mit dem Normierungsfaktor α gestreckte Sprachsignal *x* aus der Menge *T* der Trainingsäußerungen und $F0_{avg}(x)$ ist die mittlere Grundfrequenz dieser Äußerung. Diese Art der Parameterschätzung hat gegenüber dem Regressionsansatz den Vorteil, daß der Fehler $d_{m^*,b^*}(F0_{avg}(x),\alpha_s(x))$ nicht normalverteilt sein muß. Außerdem kann - wie bei der ML-VTLN – garantiert werden, daß die Produktionswahrscheinlichkeiten für die Trainingsäußerungen mit Hilfe der VTLN erhöht werden.
Wie schon bei der ML-VTLN können wir auch hier keine geschlossene Form für die Lösung des Optimierungsproblems 5.13 angeben. Genau wie beim Training der ML-VTLN wird deshalb $L_x(\alpha)$=-log($P(X(\alpha)|\Lambda,W)$) für einige α aus einem festem Raster ausgewertet (siehe Kapitel 5.1.3.1). Dies liefert für jede Äußerung *X* aus der Trainingsmenge eine Abtastung von $L_x(\alpha)$ auf diesem Raster. Diese abgetastete Funktion ist für eine Beispieläußerung in der unteren Abbildung dargestellt. Hier wurde wie bei allen unseren Versuchen ein Rasterabstand von 0,2 auf dem Intervall [0.8, 1.2] benutzt. Der optimale Normierungsfaktor aus der ML-VTLN liegt in diesem Beispiel bei 1,02.

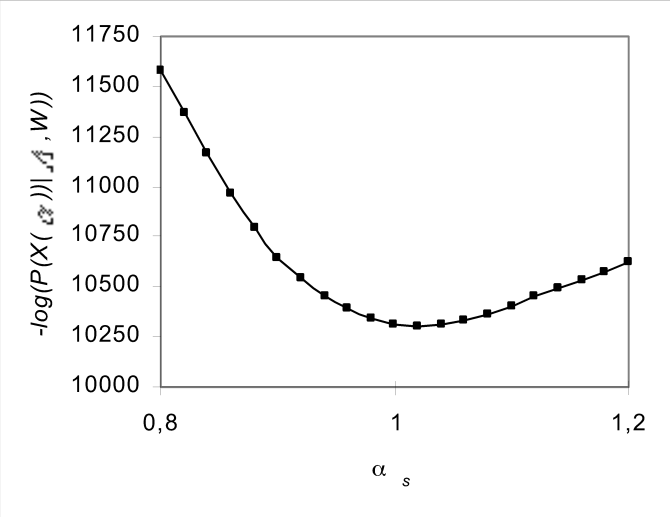

Abbildung 5.4 ML-Score $L_x(\alpha_s)$ in Abhängigkeit des Normierungsparameters α_s

Basierend auf dieser Abtastung von $L_x(\alpha)$ haben wir zwei alternative Methoden zur Lösung des Optimierungsproblems in der oberen Gleichung überprüft: erstens das erschöpfende Durchsuchen des durch m und b aufgespannten Parameterraums, und zweitens die Approximation der abgetasteten Funktion durch ein Polynom mit einer anschließenden numerischen Lösung mit Hilfe der Pseudoinversen.

1. Durchsuchen des Parameterraums

Hierzu wird $S(m,b)$ aus der oberen Gleichung auf einem äquidistanten Raster im durch m und b aufgespannten Parameterraum ausgewertet. Ergebnis ist derjenige auf dem Raster liegende Parametersatz (m,b), der das Optimierungskriterium $S(m,b)$ maximiert. $P(X(g_{m,b}(F0_{avg}))|\Lambda,W,F0_{avg})$ wird wir dabei durch lineare Interpolation der Abtastwerte von $L_x(\alpha)$ approximiert. Falls $\alpha=g_{m,b}(F0_{avg})$ außerhalb des Intervalls [0.8, 1.2] liegt, wird die Wahrscheinlichkeit beziehungsweise der Likelihood-Score des entsprechenden Intervallrandpunktes geliefert.
Da nur ein endlicher Ausschnitt des Parameterraums abgesucht werden kann, werden nur Paare (m,b) mit $0\le m\le m_{max}$ und $b_{min}\le b\le b_{max}$ betrachtet. Die Grenzen des Suchbereichs werden so festgelegt, daß $m_{max}=2m^*$, wobei m^* die Lösung des am Anfang dieses Kapitels vorgestellten Regressionsansatzes ist.
Die Abstände benachbarter Rasterpunkte auf dem Suchraster sollen möglichst klein sein, um ausreichend genaue Schätzwerte für m und b zu erhalten. Da der Suchaufwand linear mit der Anzahl der untersuchten Gitterpunkte wächst, muß ein Kompromiß zwischen Auflösungsgenauigkeit und Zeitaufwand geschlossen werden. Wir haben ein Suchraster mit 10000 Gitterpunkten verwendet, damit lag der Zeitaufwand für die Bestimmung des optimalen Parametersatzes bei einigen Minuten, was verglichen mit dem Aufwand für die Berechnung der Abtastwerte von $L_x(\alpha)$ (mehrere Stunden) verschwindend gering ist.

2. **Approximation durch Polynome**

Eine Alternative zum Durchsuchen des Parameterraumes bietet die Approximation der Funktionen $L_x(\alpha)$ durch Polynome vom Grad g=2. Gegeben sind die Werte von $L_x(\alpha)$ für die m+1=21 Abtastwerte α_k=0,8+0.02k, $k\in\{0,1,2,...,m\}$, gesucht sind die g+1 Polynomkoeffizienten $p_{i,x}$ mit $i\in\{0,1,..,g\}$. Mit der Methode der kleinsten Quadrate ergibt sich das folgende Optimierungsproblem:

$$\min \sum_{k=0}^{m} \left(\left(\sum_{i=0}^{g} p_{i,x} \alpha_k^{\ i} \right) - L_x(\alpha_k) \right)^2 \tag{5.14}$$

Dabei wird $L_x(\alpha)$ durch die Polynomkoeffizienten $(p_{0,x},...,p_{g,x})$ approximiert

$$L_x(\alpha) \approx L_x^{\ *}(\alpha) = \sum_{i=0}^{g} p_{i,x} \alpha^i \tag{5.15}$$

$S(m,b)$ aus Gleichung 5.13 kann dann folgendermaßen geschrieben werden:

$$\begin{aligned} S(m,b) &= \sum_{x\in T} L_x\left(g_{m,b}\left(F0_{avg}(x)\right)\right) \approx \sum_{x\in T} L_x^{\ *}\left(g_{m,b}\left(F0_{avg}(x)\right)\right) \\ &= \sum_{x\in T} \sum_{i=0}^{g} p_i \left(g_{m,b}\left(F0_{avg}(x)\right)\right)^i = \sum_{x\in T} \sum_{i=0}^{g} p_i (m \cdot F0_{avg}(x) + b)^i \end{aligned} \tag{5.16}$$

Gesucht ist das Minimum von $S(m,b)$: S wird nach m und b abgeleitet und die partiellen Ableitungen werden gleich Null gesetzt:

$$\begin{aligned} \frac{\partial S(m,b)}{m} &= \left(\sum_{x\in T} 2 p_{2,x} f_x^{\ 2} \right) m + \left(\sum_{x\in T} 2 p_{2,x} f_x \right) b + \left(\sum_{x\in T} p_{1,x} f_x \right) = 0 \\ \frac{\partial S(m,b)}{b} &= \left(\sum_{x\in T} 2 p_{2,x} f_x \right) m + \left(\sum_{x\in T} 2 p_{2,x} \right) b + \left(\sum_{x\in T} p_{1,x} \right) = 0 \end{aligned} \tag{5.17}$$

Dies liefert ein einfaches lineares Gleichungssystem zur Bestimmung von m^* und b^*. In unserer Implementierung dieser Methode werden nur diejenigen Äußerungen $x\in T$ in die Optimierung einbezogen, bei denen $p_{2,x}>0$ gilt (nach oben geöffnete Parabel), und außerdem das Minimum von $L_x^{\ *}(\alpha)$ im Intervall [0.8, 1.2] liegt. Diese Nebenbedingung war notwendig weil in unserer Trainingsstichprobe auch Äußerungen vorkamen, die nur Geräusche (Atmen, Schlucken) enthalten. Auf diesen Äußerungen ist eine sinnvolle Approximation von $L_x(\alpha)$ mittels einer Parabel im Allgemeinen nicht möglich.

Die Polynomapproximation ist zwar um ein Vielfaches schneller als das Durchsuchen des Parameterraumes auf einem Raster, aber die Ergebnisse weichten in unseren Versuchen um bis zu 50% von dem (nahezu) optimalen per Rastersuche bestimmten Parametersatz ab. Aus diesem Grund benutzen wir den per Polynomapproximation bestimmten Parametersatz nur zur Einschränkung des Suchbereichs für eine nachfolgende Rastersuche.

5.1.4.1 Training

Nach den Ausführungen im letzten Kapitel sieht das Training des F0-VTLN Erkenners folgendermaßen aus:

1. Bestimmung von $F0_{avg}(x)$ für alle Trainingsäußerungen x.
2. Abtastung von $L_x(\alpha)$ für alle Trainingsäußerungen x auf dem Intervall [0.8, 1.2] mit einem Rasterabstand von 0.2 mit einem ML-VTLN Erkenner.
3. Bestimmung einer Näherungslösung (m',b') für die Abbildung $\alpha_s=g_{m,b}(F0_{avg})$ mit Hilfe einer Polynomapproximation zweiten Grades für $L_x(\alpha)$.
   (Gleichungen 5.14 und 5.17)
4. Rastersuche in einer Umgebung um (m',b') zur Bestimmung der optimalen Abbildungsparameter (m^*,b^*).
5. Trainings des Erkenners. Einziger Unterschied zum Training des ‚normalen' Erkenners ohne VTLN ist die Durchführung der Sprechernormierung mit dem Normierungsparameter $\alpha_s=g_{m^*,b^*}(F0_{avg}(x))$.

5.1.4.2 Erkennung

Im Vergleich zum Erkenner ohne VTLN muß zusätzlich eine Grundfrequenzkontur für jede Äußerung bestimmt werden, aus der die mittlere Grundfrequenz $F0_{avg}$ berechnet wird. Der Normierungsparameter α_s wird wieder gemäß $\alpha_s=g_{m^*,b^*}(F0_{avg})$ bestimmt.
Die Durchführung einer schritthaltenden Erkennung sollte bei genügend großer Vorschau für die Erzeugung der Grundfrequenzkontur (250 bis 500 ms, siehe Kapitel 4.8) möglich sein, allerdings haben wir bis jetzt keine Versuche in dieser Richtung durchgeführt.

5.1.5 Ergebnisse und Vergleiche

Wir haben die F0-VTLN mit zwei Sprachen getestet: der deutschen und der chinesischen (Mandarin). Die folgende Tabelle vergleicht die Erkennungsraten des F0-VTLN-Systems mit einem Basissystem ohne VTLN und einem ML-VTLN-System.

System (GSST)	Worterkennungsrate
Basissystem ohne VTLN	83.7%
ML-VTLN (13 Hypothesen)	85.0%
ML-VTLN (1 Hypothese)	84.9%
F0-VTLN	84.8%

System (mandarin)	Worterkennungsrate
Basissystem ohne VTLN	76.8%
ML-VTLN (1 Hypothese)	78.9%
F0-VTLN	79.1%

Tabelle 5.5 Einfluß der F0-VTLN auf die Erkennungsrate

Die Ergebnisse zeigen, daß die Sprechernormierung mit F0-VTLN im Vergleich zum Erkenner ohne VTLN eine Reduzierung der Wortfehlerrate um 7% (GSST) beziehungsweise 10% (Mandarin) bringt. Die ML-VTLN erreicht zwar auch eine Verbesserung der Wortfehlerraten, ist aber langsamer als die F0-VTLN.

5.2 F_0-Merkmale bei der Erkennung tonaler Sprachen

In tonalen Sprachen bekommen phonetisch gleiche Worte durch verschiedene Grundfrequenzverläufe unterschiedliche Bedeutung. Chinesisch (Mandarin) ist so eine tonale Sprache: hier gibt es etwa 400 Aussprachesilben und 5 verschiedene Töne (Grundfrequenzverläufe) für die Aussprache einer Silbe. Durch deren Kombination ergibt sich durch eine theoretisch mögliche Anzahl von 5·400=2000 Silben, von diesen werden aber nur etwa 1300 benutzt. Zum Beispiel bekommt die Silbe ‚ma' durch die entsprechende Modulation des Grundfrequenzverlaufs vollkommen unterschiedliche Bedeutungen, wie etwa Pferd, Mutter oder schimpfen.
Es existieren zwei verschiedene Ansätze zur Erfassung der Tonalität in der Erkennung chinesischer Sprache. Der konventionelle Ansatz analysiert Silben- und Toninformation getrennt und kombiniert später die Ergebnisse beider Analysen. Der neuere Ansatz integriert die tonale Information in den Merkmalsvektor und kommt ohne eine spätere Zusammenführung von Silben- und Toninformation aus. Der konventionelle Ansatz ist in [Alf97] [Lyu95] [Wan95] beschrieben, der neuere Ansatz in [Ch97] [Zha98]. Leider konnten wir den angegebenen Quellen keine direkten Vergleiche in Bezug auf die Leistung (Wortfehlerrate) beider Ansätze entnehmen. Da die direkte Integration der tonalen Information in den Merkmalsvektor im Rahmen des JANUS-Systems einfach zu realisieren war, haben wir uns für diesen Ansatz entscheiden.

5.2.1 Das Basissystem

In der chinesischen Schrift werden alle Zeichen in einem Satz direkt aneinander gereiht, ohne daß wie in westlichen Sprachen Lücken zwischen einzelnen Bedeutungseinheiten gelassen werden. Chinesische Schriftzeichen sind für die akustische Repräsentation ungeeignet, da viele Zeichen mehrere Aussprachen haben, und umgekehrt eine Aussprachesilbe durch verscheiden Zeichen repräsentiert werden kann. Aus diesem Grund werden Sätze in Worte mit einer Länge von 1 bis 10 Zeichen segmentiert. Dabei werden Worte über eine Abbildung von einem englischen Wort auf eine Folge von chinesischen Zeichen über ein englisch - chinesisches Wörterbuch definiert. Für die akustische Repräsentation eines Wortes wird für jedes Schriftzeichen eine Aussprachesilbe in der Pinyinumschrift (eine Art Lautschrift) verwendet.
Der Merkmalsvektor unseres Ausgangssystems enthält 13 cepstrale Koeffizienten und die Kurzzeitenergie zusammen mit deren ersten und zweiten Ableitungen zur Modellierung des zeitlichen Verlaufs dieser Kurzzeitmerkmale, sowie die Nulldurchgangsrate. Damit ergibt sich ein Merkmalsvektor der Dimension 43, der durch eine lineare Diskriminanzanalyse (LDA) auf 24 Koeffizienten reduziert wird.
Das Basissystem unterscheidet 143 Phoneme. Dabei werden Vokale nach ihrer Tonalität unterschieden, so daß von jedem Vokal jede der 5 tonalen Varianten als ein

Phonem modelliert wird. Basierend auf diesen Phonemen werden Quintphone modelliert, die mittels Clusterung auf 1500 gaußsche Mischverteilungen mit je 16 Mischungskomponenten (diagonale Kovarianzmatrix) abgebildet werden. Das Ausgangssystem benutzt keine Methode zur Sprechernormierung. Weitere Informationen über das Basissystem finden sich in [Rei98].

5.2.2 Versuche und Ergebnisse

Mittels unseres Experimentes sollte überprüft werden, welchen Einfluß die Integration des F_0-Verlaufs in den Merkmalsvektor auf die Fehlerrate des Worterkenners hat. Hinweise auf die Grundfrequenz des Sprachsignals sind in den Parametern des Basissystems implizit bereits enthalten, da die Frequenz der Stimmbandanregung auch Auswirkungen auf die Nulldurchgangsrate und in geringerem Umfang auf die cepstralen Merkmale hat. Wir beziehen den Grundfrequenzverlauf explizit in die Erkennung ein, indem wir in den Merkmalsvektor eines Frames Grundfrequenzinformationen aus benachbarten Frames integrieren. Die absoluten Grundfrequenzwerte sind stark geschlechts- und damit sprecherabhängig, und deshalb nicht als Koeffizienten des Merkmalsvektors geeignet. Das Gleiche gilt für die Differenz der Grundfrequenzwerte benachbarter Frames, wie in Abbildung 4.4 gezeigt wurde. Da der in Oktaven gemessene Abstand benachbarter Grundfrequenzwerte laut dieser Abbildung weniger geschlechtsabhängig ist, ergänzen wir den Merkmalsvektor von Frame k um die Quotienten aus dem Grundfrequenzwert $F0_{k+d}$ in benachbarten Frames $k+d$ und dem Grundfrequenzwert $F0_k$ im Frame k. Die betrachtete Umgebung sollte groß genug sein um den Grundfrequenzverlauf innerhalb einer Silbe zu erfassen, aber nicht so groß, daß die Umgebung schon die für die Erkennung eines Phonems irrelevanten Grundfrequenzverläufe aus benachbarten Silben umfaßt. Wir haben die Größe der Umgebung empirisch zu $d \in D = \{-8,-4,-2,-1,1,2,4,8\}$ gewählt, bei unserem Frameabstand von 100 ms entspricht das einer Dauer von 160 ms. Zusätzlich zu den F_0-Merkmalen integrieren wir in den Merkmalsvektor noch ein Maß für den Grad der Stimmhaftigkeit des Sprachsignals innerhalb der Frames aus der Umgebung $D \cup \{0\}$. Als dieses Maß benutzen wir den Kreuzkorrelationskoeffizienten $\sigma_S(k)$ zweier aufeinanderfolgender Grundperioden in Frame k (siehe Gleichung 4.25 in Kapitel 4.11.3). Insgesamt haben wir damit für unsere ersten Experimente 8+9=17 neue Merkmale erhalten.

Im ersten Experiment wollten wir feststellen, wie sich die Integration der Grundfrequenzwerte und der SH/SL-Maße in den Merkmalsvektor auf die Fehlerrate des Worterkenners auswirkt. Das Basissystem benutzt einen 43 dimensionalen Merkmalsvektor, der mittels einer LDA auf 24 Elemente reduziert wird. Eine naheliegende Möglichkeit zur Integration unserer 17 Grundfrequenzmerkmale besteht darin, den originalen 43 dimensionalen Merkmalsvektor um unsere 17 Grundfrequenzmerkmale zu erweitern, und daraufhin wieder eine Dimensionsreduktion mittels LDA durchzuführen. Mit dieser Vorgehensweise ist aber nicht sichergestellt, daß unsere Grundfrequenzmerkmale nach der Transformation des Merkmalsraums (LDA)

im resultierenden Merkmalsvektor enthalten sind. Da wir dies im ersten Experiment erzwingen wollten, haben wir uns zu einer zweistufigen Merkmalsraumtransformation über zwei LDAs entschlossen: die erste LDA liefert uns den gleichen 24-elementigen Merkmalsvektor der im Basissystem verwendet wird. Diesen Merkmalsvektor, ergänzt um unsere 17 Grundfrequenzmerkmale, reduzieren wir mittels einer zweiten LDA auf 30 Elemente. Die Ergebnisse dieses Versuches sind in der folgenden Tabelle aufgelistet.

System (mandarin)	Worterkennungsrate
Basissystem	76.8%
24+8 Nullen	76.7%
24+8 F_0-Merkmale	78.0%
24+8 F_0-Merkmale + 9 SH/SL-Merkmale	78.6%

Tabelle 5.6 Erkennungsrate nach Einbeziehung von Grundfrequenzinformationen

Die Wortfehlerrate konnte durch Integration der F_0- und SH/SL-Merkmale (siehe vierte Zeile in der Tabelle) um 7.8% reduziert werden, allerdings auf Kosten einer Vergrößerung des Merkmalsraumes von 24 auf 30 Dimensionen. Beim Versuch in der zweiten Zeile der Tabelle haben wir den ursprünglichen Merkmalsvektor statt um 8 F_0-Merkmale um 8 Nullen ergänzt um sicherzustellen, daß die Verbesserung der Wortfehlerrate wirklich einzig und allein den F_0-Merkmalen zuzuschreiben ist.
In einem weiteren Experiment wollten wir ermitteln, wie sich die Integration der Grundfrequenzmerkmale in unterschiedlich großen Merkmalsräumen auf die Fehlerrate auswirkt. Dazu wurde der originale 43-elementige Merkmalsvektor des Basissystems um die 17 F_0- und SH/SL-Merkmale ergänzt, und dieser Merkmalsraum mittels einer einzigen LDA auf n=24, 30 oder 36 Dimensionen reduziert. Die Ergebnisse dieser Versuche sind in der unteren Tabelle zusammengefaßt, das Ergebnis des Basissystems steht in der linken oberen Spalte.

Dimension n	Erkennungsrate ohne -	- mit Grundfrequenzmerkmalen
24	76.8%	77.0%
30	78.0%	78.3%
36	78.0%	78.9%

Tabelle 5.7 Erkennungsraten bei verschiedenen Dimensionen des Merkmalsraums

Durch Einbeziehung der Grundfrequenzmerkmale kann die Erkennungsleistung zwar gesteigert werden, die Fehlerrate verbessert sich aber nur wenig: sie fällt um 1 bis 4%. Dabei ist die Verbesserung um so größer, je höher die Dimension des Merkmalsraumes gewählt wird. Auffällig an den Ergebnissen ist, daß im Experiment mit der einstufigen LDA auf einem 30-dimensionalen Merkmalsraum nur 78.3% Erkennungsrate erreicht wurden, während das vorige Experiment mit der zweistufigen LDA mit 78.6% etwas besser Ergebnisse lieferte. Das führen wir darauf zurück, daß die einstufige LDA nicht die optimale Merkmalsraumtransformation durchgeführt

hat, denn die zweistufige Variante läßt sich zu einer einzigen (und besseren) Merkmalsraumtransformation zusammenfassen.

5.3 F_0-Merkmale für nicht-tonale Sprachen

Durch die Integration der SH/SL-Information in den Merkmalsvektor konnte die Erkennungsrate des chinesischen Worterkenners von 78.0% auf 78.6% gesteigert werden (siehe Tabelle 5.6). Wir nehmen an, daß durch dieses Merkmal dem Erkenner der Einsatz der Stimmbandanregung (‚voice onset') verfügbar gemacht wird, ein Merkmal, daß besonders zur besseren Unterscheidung von harten und weichen Plosiven herangezogen werden kann.
Dieses Merkmal könnte nicht nur in der chinesischen Sprache, sondern auch in nichttonalen Sprachen eine sinnvolle Ergänzung des Merkmalsvektors sein. Aus diesem Grund haben wir wie bei den Versuchen mit dem chinesischen Erkenner den Merkmalsvektor des GSST-Erkenners (Kapitel 7.2) um 8 SH/SL-Merkmale wie in Kapitel 5.2.2 ergänzt, und die resultierende Erkennungsrate gemessen. Die Ergebnisse dieses Versuches sind in der folgenden Tabelle angegeben, die Dimension des Merkmalsraumes ist bei dem Versuch die gleiche wie im Basissystem.

System (GSST)	Worterkennungsrate
Basissystem	83.7%
System mit SH/SL-Information	83.5%

Tabelle 5.8 Erkennungsrate nach Einbeziehung von Grundfrequenzinformationen

Leider konnte die Leistung des Erkenners durch Integration der Grundfrequenzinformation nicht verbessert werden. Eine mögliche Erklärung dafür ist, daß der Einsatz der Stimmbandschwingung bereits implizit in den cepstralen Merkmalen und deren Deltas enthalten ist, so daß unser SH/SL-Merkmal keine neuen Informationen liefern kann.

5.4 Weitere Anwendungen

Grundfrequenzinformationen können außer zur Sprechernormierung und als Merkmal in der Erkennung tonaler Sprachen auch noch auf andere Art und Weise zur Verbesserung der Sprach- beziehungsweise Sprechererkennung beitragen. Die folgende Aufzählung zeigt einige weitere Nutzungsmöglichkeiten mit zugehörigen Literaturverweisen.

1. Sprechererkennung
   In [Ch98] wird beschrieben, wie die Fehlerrate eines auf cepstralen Merkmalen basierenden Sprechererkennungssystems durch Integration von Grundfrequenzin-

formationen um 30-40% reduziert werden konnte. Ähnliche Experimente werden in [Mar98] beschrieben.

2. Prosodie
   In [Lop98] wird dargestellt, wie die Fehlerrate eines kontinuierlichen Zifferner-kenners für spanische Sprache durch Einbeziehung von prosodischen Informationen (Grundfrequenz) um 16% reduziert werden konnte.

3. Pitchsynchrone Analyse
   Dieses Verfahren wird in [Rab78] erwähnt. Leider konnten wir keine Erkennungsergebnisse von Experimenten im Zusammenhang mit pitchsynchroner Analyse finden.

4. Rauschreduktion
   In [Cos98] wird eine Methode zur Rauschunterdrückung namens ‚correlogram subtraction' beschrieben, die Informationen über der Grundfrequenz des Sprachsignals benutzt, wobei aber leider noch keine konkreten Erkennungsergebnisse angegeben sind.

5. Sprachkodierung
   Grundfrequenzinformation wird in einigen Verfahren zur Komprimierung von Sprachsignalen benötigt. Ein Vertreter dieser Verfahren wird in [Mer99] beschrieben.

6 Zusammenfassung der Ergebnisse

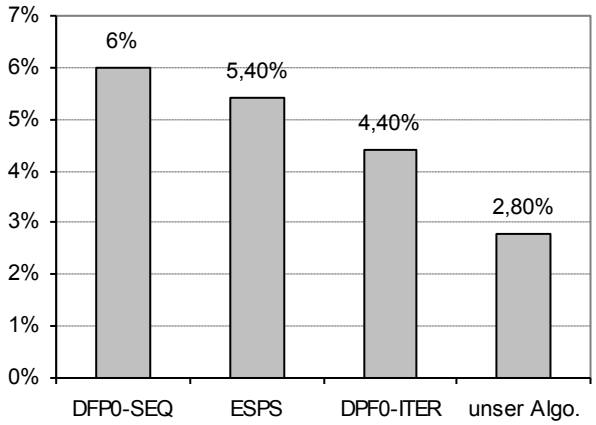

Abbildung 6.1 Grobfehlerrate mehrerer Grundfrequenzverfahren (SPONTAN)

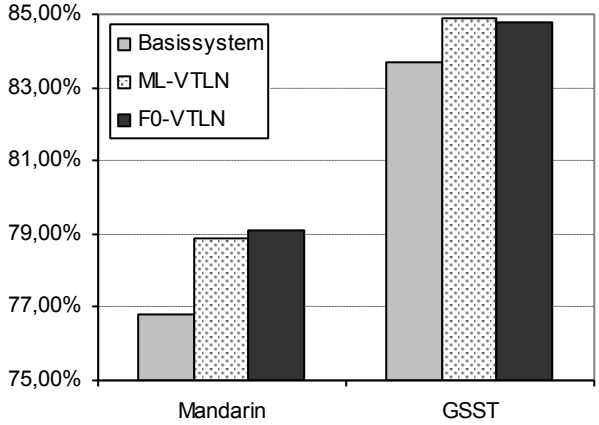

Abbildung 6.2 Vergleich der Erkennungsraten von F0-VTLN und ML-VTLN

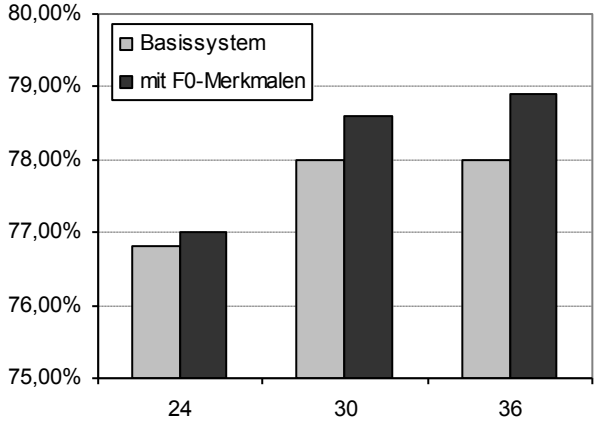

Abbildung 6.3 Erkennungsraten bei Einbeziehung von F0-Konturen (Mandarin)

7 Anhang

7.1 Die SPONTAN-Stichprobe

Die SPONTAN-Stichprobe wurde im Rahmen des DFG-Projekts Intonation-Register-Modus-Fokus [Bat92d] am Institut für Deutsche Philologie der Ludwig-Maximilians-Universität München aufgenommen, und im Zusammenhang mit dem Lehrstuhl für Mustererkennung (Informatik 5) der Universität Erlangen-Nürnberg in den nachfolgenden BMFT-Projekten ASL und Verbmobil weiter verarbeitet. Der Schwerpunkt der Untersuchungen lag auf der Untersuchung des Unterschiedes zwischen spontaner und gelesener Sprache. Die SPONTAN-Stichprobe besteht aus zwei Teilen: der eine Teil sind spontane Äußerungen, der andere Teil sind textuell identische Lesungen. Zur Erzeugung des spontansprachlichen Korpus saßen zwei ‚naive' Versuchspersonen an einem Tisch ohne einander zu sehen. Ihnen wurde die Aufgabe gestellt ein gemeinsames Problem in einer Blockweltumgebung zu lösen, wobei keine der Versuchspersonen wußte, daß es in dem Experiment eigentlich um die Aufnahme von Sprachdaten ging. Es wurden zwei Experimente mit insgesamt vier Sprechern durchgeführt: drei davon waren Studentinnen der Psychologie, einer war Physikstudent. Jeder Sprecher bekam ein eigenes Mikrofon mit Aufzeichnungsgerät. Die Versuchspersonen waren einander bekannt und sprachen eine leicht dialektal gefärbte Variante der süddeutschen, bayrischen Umgangssprache. Da es in dem Experiment um den Vergleich zwischen spontan gesprochenen und gelesenen Äußerungen ging, wurde ein Teil der Äußerungen den gleichen Sprechern nach 9 Monaten noch einmal zum Lesen vorgelegt. Die Aufnahmebedingungen im zweiten Experiment waren die gleichen wie schon beim ersten. Dabei mußte jeder Sprecher sowohl seine eigenen Äußerungen vorlesen, wie auch die des entsprechenden Dialogpartners. Somit besteht die SPONTAN-Stichprobe aus einem Drittel spontaner, und zu zwei Dritteln aus gelesener Sprache. Insgesamt wurden 1329 Äußerungen mit zusammen 28 Minuten Dauer aufgenommen. Die analogen Aufnahmen wurden mit 4,5 kHz tiefpaßgefiltert, mit 10 kHz abgetastet und 12 bit digitalisiert.
Die so erzeugten Aufnahmen wurden anschließend folgendermaßen etikettiert (eine detailliertere Darstellung findet sich in [Bat93b])

1) Automatische SH/SL-Entscheidung und F0-Berechnung mit unterschiedlichen Algorithmen; Berechnung des Energieverlaufs (Frameabstand jeweils 12.8 ms)
2) Automatische Lautzuordnung mit dem Spracherkennungssystem ISADORA ([ST95]) basierend auf einer Modifikation der SAMPA-Notation (Frameabstand wieder 12.8 ms)
3) Manuelle Korrektur von F0 und Lautsegmentierung, sowie Etikettierung der laryngalisierten Bereiche durch erfahrene Phonetiker. Die Laryngalisierungen wurden nach dem MÜSLI-System (Münchner Schema für Laryngalisierungs-Identifikation) etikettiert, das in [Bat93a] beschrieben wird.

4) Etikettierung der Grenzen unterschiedlicher Phrasentypen durch einen erfahrenen Linguisten
5) Ekikettierung des Satzmodus nach verschiedenen Gesichtspunkten, zum Beispiel nach dem Altmannschen Satzmodussystem

Außerdem wurden die Äußerungen verschiedenen Hörtests unterzogen, in denen Versuchspersonen die am betontesten wahrgenommene Silbe markieren sollten, und die Äußerungen als Frage, Aussage, Imperativ oder Exklamativ, sowie als spontan oder gelesen klassifizieren sollten.

7.2 Das Basissystem und der GSST

Die Vorverarbeitung des Basissystems P1 berechnet 13-dimensionale Melscale Cepstrum-Merkmale, die um ihre ersten und zweiten Ableitungen und einen Energiewert ergänzt einen 40-dimensionalen Merkmalsvektor ergeben. Dieser wird einer LDA unterzogen und auf 28 Dimensionen reduziert. Das Wörterbuch umfaßt etwa 6000 Wörter, 68 Phoneme werden unterschieden. Detailliertere Angaben zum Basissystem finden sich in [KA97].

Verbmobil ist die Bezeichnung für ein Langzeitforschungsprojekt mit der Zielsetzung der Entwicklung eines tragbaren maschinellen Sprachübersetzers, der es etwa Geschäftsleuten aus verschiedenen Sprachräumen ermöglichen soll, in ihrer jeweils eigenen Sprache miteinander zu kommunizieren. Im Rahmen dieses Projektes wurde eine Datenbasis mit etwa 32 Stunden transkribierter spontaner deutscher Sprache als Trainingsmaterial erstellt. Um eine repräsentative Mischung verschiedener deutscher Dialekte zu erlangen, wurden die Sprachdaten an vier verschiedenen Orten innerhalb Deutschlands gesammelt. Obwohl die Domäne eingeschränkt ist, wurden keinerlei Restriktionen bei der Auswahl der Sprecher oder Sprachstile angewendet. Typische Phänomene spontan gesprochener Äußerungen, wie etwa Hintergrundgeräusche, Stottern, unvollständige oder grammatikalisch falsche Sätze sind daher keine Seltenheit. Was die Sprachmodellierung angeht, ist der Verbmobil-Korpus eher klein. Die Äußerungen umfassen etwa 300000 Wörter, das Vokabular ist etwa 6000 Wörter groß.

Der German Spontaneous Scheduling Task (GSST) ist ein auf der Verbmobil-Datenbasis aufbauender Benchmark mit einer Trainingsmenge von 14009 Äußerungen und einer Testmenge von 343 Äußerungen.

Weitere Angaben zum Verbmobil-Korpus und zum GSST findet man etwa in [KA97]. Dort sind auch die Resultate der GSST-Evaluationen des Janus RTK und anderer Systeme aus den Jahren 1995 und 1996 aufgeführt.

7.3 Die chinesische Datensammlung

Die chinesischen Sprachdaten (Mandarin) wurden im Rahmen des „GlobalPhone“-Projekts [Sch97] gesammelt. Insgesamt wurden 132 Sprecher aufgenommen, etwa die Hälfte davon männlich, die Artikel aus der chinesischen Tageszeitung „People‘s Daily“ vorlasen. So wurden 10214 Sätze mit einer Gesamtlänge von 28,6 Stunden gesammelt. Da die Aufnahmen an sehr unterschiedlichen Orten erfolgten, unterscheiden sich die Hintergrundgeräusche erheblich. Zwar wurden die Sprecher angewiesen hochchinesisch zu sprechen, durch die verschiedenen Aufnahmeorte wird aber eine Beeinflussung durch eine Vielzahl von Dialekten gegeben. Im Chinesischen wirken sich die verschiedenen Dialekte auf das Vorlesen besonders stark aus, da keine Bindung der Aussprache an die Schrift besteht.
Die Aufnahmen wurden mit 16 kHz abgetastet und in einzelne Sätze segmentiert. Außerdem wurden die Transkriptionen manuell an die Äußerungen angepaßt, da die Sprecher teilweise von den Original-Zeitungstexten abwichen. Die häufigsten Abweichungen waren Wiederholungen oder die Auslassung von Textteilen und vom Sprecher erzeugte Nebengeräusche wie Husten und Räuspern. Die so aufbereiteten Texte wurden in das Pinyinsystem (eine Art Lautschrift) umgesetzt. Weitere Information zur Datensammlung finden sich in [Rei98], Details zum Pinyinumsetzer stehen in [Rei97].

7.4 Dokumentation der Schnittstelle zum PitchTracker

Der Grundfrequenzalgorithmus wurde als C++ Klasse ‚CPitchTracker‘ implementiert. Die Klasse stellt Funktionen zur Parametrierung des Algorithmus bereit, sowie Funktionen zur Übergabe des Eingabesignals und zur Übermittlung des Ergebnisses. Die Ein- und Ausgabefolgen werden als Datenstrom betrachtet, der aus Effizienzgründen blockweise übertragen wird.

1. **Übergabe des Eingabesignals**

   Da das Verfahren schritthaltend arbeiten soll, wird der Strom der Abtastwerte im Eingabesignal stückweise an den PitchTracker übergeben. Die Blockgröße ist dabei beliebig.
   Der Beginn eines neuen Eingabesignals wird durch Aufruf der Methode Reset() angezeigt. Eine Folge von Aufrufen der Methode ProcessSampleData(SampleData,SampleCount) übergibt die einzelnen zeitlich aufeinanderfolgenden Signalblöcke.

2. **Übernahme der Ergebnisfolge**

   Das Ergebnis besteht aus einer Folge von Grundperiodendauern, die Länge einer Grundperiode wird in Samples (also Anzahl der Abtastwerte im Eingabesignal)

angegeben. Der Empfänger der Ergebnisfolge muß die Übergabeschnittstelle CPitchTrackerCallbackInterface implementieren, also von dieser Empfängerklasse abgeleitetet sein. Das konkrete Empfängerobjekt wird dem PitchTracker durch Aufruf von CPitchTracker::SetCallback() mitgeteilt. Jeder Teilblock wird durch Aufruf der Methode ProcessSampleData(Folgenbeginn, Folgenlänge) an den Empfänger übergeben.
Aus Effizienzgründen werden Teilblöcke der Ergebnisfolge im PitchTracker-Objekt gespeichert und erst dann gesendet, wenn der interne Puffer voll ist. Das Leeren dieses Puffers kann durch Aufruf von Flush() vorzeitig erzwungen werden. Der PitchTracker liefert die Ausgabefolge mit einer einstellbaren zeitlichen Verzögerung, um so durch Ausnutzen einer Vorausschau die Leistung des Verfahrens zu verbessern (siehe Kapitel 4.8). Am Ende einer Äußerung gibt es jedoch keine Eingabedaten für eine Vorausschau, deshalb müssen die letzten Folgenelemente (Frames) mit verkürzter Vorausschau bestimmt werden. Das Ende einer Äußerung wird dem PitchTracker durch Aufruf von CompleteSampleData() angezeigt.

3. Parametrisierung

Die Bedeutung der folgenden Parameter wurden bereits ausführlich in Kapitel 4 beschrieben.

Funktion	*Parameterbeschreibung*
SetSampleFreq	Abtastrate des Eingabesignals (in Hz)
SetResultFrameShift	Abstand des Beginns zweier aufeinanderfolgender Kurzzeitanalysefenster in der Ausgabefolge (in Samples)
SetInitialFrameCenter	Zentrum des ersten Analysefensters (in Samples)
SetMinExpectedF0	Kleinste erwartete Grundfrequenz (in Hz)
SetMaxExpectedF0	Größte erwartete Grundfrequenz (in Hz)
SetLowpassFilterCutoffFreq	Grenzfrequenz des Tiefpaßfilters (in Hz)
SetLookAheadTime	Erlaubte Vorausschau (in Millisekunden)
SetUseKonturFilter	Ein-/Ausschalten der Konturfilterung (boolean)

Die folgenden Parameter dienen ausschließlich der Beschleunigung des Algorithmus.

SetSTAFrameRate	Rate der zu berechnenden Kurzzeitanalysefenster (in Hz)
SetDownsampleFreq	Neuabtastrate $f_{\text{neuabtast}}$ (in Hz)
SetUseFastAutocorr	Erlaube Benutzung der Autokorrelation statt des Rootcepstrums (ist schneller bei kleinen Fenstern)

Hier folgen die Signaturen der C++ Schnittstellen zu den Klassen CPitchTracker und CPitchTrackerCallbackInterface:

```
class CPitchTracker {
        public:
                CPitchTracker();

                ///// Ergebnisfolge
                void SetCallback(CPitchTrackerCallbackInterface& i);
                void RemoveCallback();

                ///// Konfigurieren
                void SetSampleFreq(int x);
                void SetResultFrameShift(int x)
                void SetInitialFrameCenter(int x)
                void SetMinExpectedF0(int x);
                void SetMaxExpectedF0(int x);
                void SetLowpassFilterCutoffFreq(int x);
                void SetLookAheadTime(int x);
                // Parameter fuer Beschleunigung
                void SetSTAFrameRate(int x)
                void SetDownsampleFreq(int x)  ;
                void SetUseFastAutocorr(bool x)
                void SetUseKonturFilter(bool x);

                ///// Ablaufsteuerung
                void Reset() { haveStartedNewSignal=true; };
                void ProcessSampleData(
                        std::vector<short>::const_iterator SampleData,
                        int SampleCount);
                void Flush();
                void CompleteSampleData();
};

class CPitchTrackerCallbackInterface {
        public:
                virtual void ProcessPitchData(
                        std::vector<int>::const_iterator T0, int SampleCount) = 0;
};
```

Der gesamte Code ist in den beiden C++ Dateien PitchTracker.cpp und PitchTracker.h untergebracht. Zusätzlich werden nur die C++ Standard Template Library

(STL) für dynamische Arrays und komplexe Zahlen benötigt, sowie eine Implementation der schnellen Fouriertransformation über die in Datei DFT.h angegebene Schnittstelle: void FFT(std::vector<Complex>::iterator begin, int size).

Zum Abschluß folgt noch ein Codefragment, daß die Benutzung der Schnittstellen demonstriert:

```
class CPitchTrackerApplication : public CPitchTrackerCallbackInterface {
        public:
                void Run() {
                        myPitchTracker.Set...();                // initialisiere alle Parameter
                        myPitchTracker.SetCallback(*this);  // Empfänger festlegen
                        foreach Äußerung {
                                myPitchTracker.Reset();
                                foreach Teilblock_der_Äußerung
                                        myPitchTracker.ProcessSampleData(...);
                                myPitchTracker.CompleteSampleData();
                        }
                        myPitchTracker.RemoveCallback();
                } // Ende von Run()

        private:
                void ProcessPitchData(vector<int>::const_iterator T0, int Count) {
                        // wird vom PitchTracker aufgerufen (Callback)
                        // drucke übergebene T0-Werte
                        for (int i=0;i<Count;i++) cout << T0[i] << endl;
                }
                CPitchTracker myPitchTracker;
};
```

7.5 Dokumentation der JANUS FeatureSet-Methoden

Zur Evaluation unserer Versuche über die Ausnutzung von Grundfrequenzinformation in der automatischen Spracherkennung haben wir unser Grundfrequenzverfahren in das Karlsruher Spracherkennungssystem JANUS [KA97] integriert. Dazu wurden die JANUS-FeatureSet-Klasse um zwei neue Methoden ergänzt: die Methode ‚pitch' berechnet aus dem Sprachsignal eine Grundfrequenzkontur, die Methode ‚crosscorr' bestimmt aus dem Sprachsignal und seiner Grundfrequenzkontur die normierte Kreuzkorrelation zwischen zwei aufeinanderfolgenden Grundperioden. Die verwendeten Parameter wurden bereits in Abschnitt 7.4 beschrieben und sind zusätzlich noch im JANUS-System dokumentiert.

8 Literaturverzeichnis:

[Alf97] **Ying Pang Alfred, L.W. Chan, P.C. Ching:** *Automatic Recognition of Continuous Canto-nese Speech with Very Large Vocabulary*, EuroSpeech '97, Rhodos (Volume 3 pages 1551 - 1554)

[Bat92] **A.Batliner u.a.:** *Grammatisch-intonatorische Modus- und Fokusmarkierung in unterschiedlichen Registern gesprochener Sprache*, 1992, DFG-Abschlußbericht, München

[Bat93a] **A. Batliner u.a.:** *MÜSLI: A Classification Scheme for Laryngealizations*, in House, D.; Touati, P. (Hrsg.): Proc. of an ESCA Workshop an Prosody, Lund University, Department of Linguistics, Lund, 1993S. 176-179.

[Bat93b] **A. Batliner, A. Kießling, E. Nöth:** *Die prosodische Markierung des Satzmodus in der Spontansprache – Methodologie und erste Ergebnisse*, ASL-Süd-TR-14-93/LMU, 1993.

[Bel57] **R. Bellman:** *Dynamic programming*, Princeton University Press, 1957.

[Ch97] **C.J. Chen u.a.:** *New methods in Continuous Mandarin Speech Recognition*, EuroSpeech '97, Rhodos (Volume 3 pages 1543 - 1546)

[Ch98] **Cheng, Leung:** *Speaker Verification Using Fundamental Frequency*, ICSLP, 1998

[Cos98] **Cosi, Pasquin, Zovato:** *Auditory Modeling Techniques for Robust Pitch Extraction and Noise reduction*, ICSLP, 1998

[Dai97] *Spracherkennung bei Daimler-Benz*, Verbmobil Akustik Workshop, Herrenberg 9. und 10. Oktober 1997.

[Den92] **J. Denzler:** *Transformation von Sprachsignalen in Laryngosignale mittels künstlicher neuronaler Netze*, Diplomarbeit, Universität Erlangen-Nürnberg, 1992.

[Den93] **J. Denzler, R. Kompe, A. Kießling***: Going Back to the Source: Inverse Filtering of the Speech Signal with ANNs,* in Proc. European Conf. On Speech Communictaion and Technology, Bd. 1, Berlin, 1993, S. 111-114.

[Ei96] **Eide, Gish:** A Parametric Approach to Vocal Tract Length Normalization, ICASSP´96 S.346 ff.

[FA71] **Fourcin:** *First applications of a new laryngograph, Medical and Biological Illustration,* Bd. 21,1971.

[Gir98] **Girardi, Shikano, Nakamura***: Creating Speaker Independent HMM Models for Restricted Database using STRAIGHT-TEMPO Morphing*, ICSLP 1998

[Gon95] **Gong:** *Speech Recognition in noisy environments: A survey*, Speech Communication 16, 1995, p. 261-291

[Har94] **S. Harbeck:** *Entwicklung eines robusten Systems zur periodensynchronen Analyse der Grundfrequenz von Sprachsignalen*, Diplomarbeit, Lehrstuhl für Mustererkennung (Informatik 5), Universität Erlangen-Nürnberg, 1994

[Ind87] **H. Indefrey:** *Untersuchung von Algorithmen zur Grundfrequenzbestimmung von Sprachsignalen unter Verwendung eines Laryngographen als Referenzgröße*, Doktorarbeit, TU München, 1987.

[Jäh97] **B. Jähne:** *Digitale Bildverarbeitung*, 4. Auflage, Springer, 1997.

[KA97] **Michael Finke, Petra Geutner, Hermann Hild, Thomas Kemp, Klaus Ries, Martin Westphal**: *The Karlsruhe Verbmobil Speech Recogition Engine*, In IEEE Conference on Acoustics, Speech and Signal Processing, 1997.

[Kah93] **B. Kahles:** *Detektion von Laryngalisierungen mittels Neuronaler Netze im invers gefilterten Sprachsignal,* Studienarbeit, Lehrstuhl für Mustererkennung , Universität Erlangen-Nürnberg, 1993.

[Kie92] **A. Kießling:** *DP-Based Determination of F0 Contours From Speech Signals, in Proc. Int. Conf. on Accoustics, Speech and Signal Processing*, Bd. 2, San Francisco, CA, 1992, S. II-17-II-20.

[Kie96] **A. Kießling***: Extraktion und Klassifikation prosodischer Merkmale in der automatischen Sprachverarbeitung,* Universität Erlangen-Nürnberg, 1996.

[Koh77] **K. J. Kohler:** *Einführung in die Phonetik des Deutschen, Grundlagen der Germanistik*, Erich Schmidt Verlag, Berlin, 1977

[Kom89] **R. Kompe:** *Prosody takes over: A prosodically guided dialog system, in Proc. European Conf. on Speech Communication and Technology*, Bd. 3, Berlin, 1993, S. 2003-2006.

[Kom96] **R. Kompe:** *Prosody in Speech Undestanding Systems*. Dissertation. Technische Fakultät der Universität Erlangen-Nürnberg, 1996.

[Lee96] **Lee, Rose:** *Speaker Normalization Using Efficient Frequency Warping Procedures*, ICASSP´96 S.353 ff.

[Lop98] **Lopez, Caminero, Cortazar, Hernadez:** *Improvement on Connected Number Recognition Using Prosodic Information*, ICSLP, 1998

[Lyu95] **Ren-Yuan Lyu u.a.:** *Golden Mandarin (III) - A User-Adaptive Prosodic-Segment-Based Mandarin Dictation Machine for Chinese Language with Very Large Vocabulary*, ICASSP 95, Detroit (Volume 1, Page 57)

[Mar98] **Markov,Nakagawa:** *Text-Independent Speaker Recognition Using Multiple Information Sources*, ICSLP, 1998

[Med91] **Y.Medan:** *Super Pitch Resolution Pitch Determination of Speech Signals,* IEEE Transactions on Signal Processing, Vol. 39, No. 1, January 1991.

[Mer99] **Mermelstein, Qian:** *Analysis by Synthesis Speech Coding with Generalized Pitch Prediction*, ICASSP, 1999

[Nie94] **H. Niemann u.a.:** *Modern modes of Man-Machine Communication,* pages 12-1-12-12. University of Maribor, Slovenia, 1994

[Rab78] **L. Rabiner, R. Schafer:** *Digital Processing of Speech Signals*, Prentice-Hall Inc., Englewood Cliffs, New Jersey, 1978.

[Rei97] **Jürgen Reichert:** *Lautschriftumsetzung und Wortrennung der chinesischen Schriftsprache*, Studienarbeit, Universität Karlsruhe 1997

[Rei98] **Jürgen Reichert:** *Spracherkennung im Chinesischen*, Diplomarbeit, Universität Karlsruhe, 1998

[Ros74] **M. Ross:** *Average magnitude difference function pitch extractor*, IEEE Trans. Acoustics, Speech, and Signal Processing, Vol. ASSP-22, Nr. 5, 1974, S. 353-362

[Sch97] **Tanja Schultz, Michael Westphal, Alex Waibel:** *The GlobalPhone Project: Multilingual LVCSR with Janus3*, 2nd SQEL-Workshop, Plzen, Tschechien 1997.

[Sen78] **S. Seneff:** *Real-Time harmonic pitch detector*, IEEE Trans. on Acoustics, Speech and Signal Processing, Bd. ASSP-26, Nr. 4, 1978, S. 358-365.

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

[Wan95] **Hsin-min Wang u.a.:** *Complete recognition of continous mandarin speech for chinese language with very large vocabulary but limited training data*, ICASSP 95, Detroit (Volume 1, Page 61)

[Weg96] **Wegmann,Allaster,Orloff,Peskin:** *Speaker Normalization on Conversational Telephone Speech*, ICASSP´96 S.339 ff.

[Wie66] **N. Wiener***: Extrapolation, Interpolation and Smoothing of Stationary Time Series*, MIT Press, Cambridge, MA, 1966.

[Zha98] **Puming Zhan u.a.:** *Dragon Systems' 1997 Mandarin Broadcast news System*, Darpa (Broadcast News Transcription and Understanding Work-shop) 98, Lansdowne, Virginia

[Zot95] **A. Zottmann:** *Weiterentwicklung des sequentiellen Grundfrequenzberechnungsverfahrens DPGF-SEQ,* Manuskript, Lehrstuhl für Mustererkennung (Informatik 5), Universität Erlangen-Nürnberg, 1995.

[ZW97] **Zhan, Westphal:** *Speaker Normalization Based On Frequency Warping*, ICASSP´97 S.1039 ff.