Studienarbeit : Spracherkennung mit im Raum verteilten Mikrofonen

von Daniel Gärtner[1]

Betreuer : Florian Metze[2]

ILKD Professor Waibel, Universität Karlsruhe, ISL

27. April 2005

[1]gdaniel@ira.uka.de
[2]metze@ira.uka.de

Zusammenfassung

Diese Studienarbeit befasst sich mit der automatischen Verschriftung, der sogenannten Transkription, von Gesprächen zwischen mehreren Personen, den „Meetings". Es werden keine Nahbesprechungsmikrofone verwendet, sondern nur einige wenige Raummikrofone.

Die Mikrofone befinden sich auf einem Tisch, um den die Sprecher sitzen, und zeichnen somit zum einen alle Sprecher, zum anderen auch Störungen wie Hall und sämtliche Nebengeräusche wie Lüfter mit auf. Die Position der Mikrofone ist unbekannt und interessiert auch nicht weiter.

Delay and Sum ist ein Verfahren, bei dem die Aufnahmen aus den einzelnen Kanälen zuerst mit einem „Delay", also einer Verzögerung, versehen werden, danach werden sie addiert. Die Idee dahinter ist, dass Signale die von einer bestimmten Quelle ausgehen wegen der (endlichen) Schallgeschwindigkeit (343 $\frac{m}{s}$ in Luft) zu verschiedenen Zeitpunkten von im Raum verteilten Mikrofonen aufgezeichnet werden. Wenn man die einzelnen Kanäle nun geeignet verzögert, kann man erreichen, dass sich Signale aus einer bestimmten Richtung überlagern, also verstärken, während in der Summe Signalanteile aus anderen Richtungen gedämpft werden. Wir wollen mittels Delay and Sum die gegebenen Kanäle zu einem Kanal zusammenfassen und dann diesen Kanal dekodieren.

Schwerpunkt dieser Studienarbeit soll das Finden der für das Delay and Sum benötigten Informationen sein : Welche Kanäle sollen kombiniert werden und mit welchen Delays sollen die Kanäle aufeinander addiert werden.

Wir nutzen die Korrelation zweier Kanäle und stellen verschiedene, einfache Algorithmen vor, wie aus der Korrelation passende Delays gewonnen werden können und wie wir fehlerhafte Delays, die durch Störgeräusche hervorgerufen werden, herausfiltern können.

Wir arbeiten mit verschiedenen Filtern (bis zu 7,2% durchschnittliche Senkung der Wortfehlerrate) und einer Methode, mit der wir einen Teil der fehlerhaften Delays identifizieren können und einem daraus resultierenden Auswahlverfahren der Kanäle, die wir für Delay and Sum verwenden (durchschnittlich 4,2% bei manueller Segmentierung).

Die Wortfehlerrate kann durch Kombinationen dieser Methoden bei Verwendung von vier Mikrofonen um zwischen 0,1% und 14,1% gesenkt werden.

Ich versichere hiermit wahrheitsgemäß, die Arbeit selbständig angefertigt, alle benutzten Hilfsmittel vollständig und genau angegeben und alles kenntlich gemacht zu haben, was aus Arbeiten anderer unverändert oder mit Abänderung entnommen wurde.

Unterschrift: ..... , Mat.No. 1078162

Inhaltsverzeichnis

Kapitel 1

Einführung

1.1 Motivation

Automatische Spracherkennung wird der Vorgang genannt, bei dem gesprochene Sprache von einem Computer in Text, also Zeichenfolgen umgewandelt wird. Dafür gibt es verschiedenste Einsatzmöglichkeiten, die Transkription von diktierten Texten ist nur eine davon. Spracherkennung von spontaner Sprache mit Nahbesprechungmikrofonen und sprecherunabhängigen Erkennern ist heutzutage ein beherrschtes Gebiet. Jedoch ist es nicht immer möglich oder erwünscht, Nahbesprechungsmikrofone einzusetzen. In dieser Arbeit befassen wir uns mit Meeting Situationen. Eine typische Meeting Situation ist in Abbildung 1.1 dargestellt. Das Problem bei Meeting Tasks ist zum einen, dass auf den Aufnahmen jede Menge unerwünschte Nebengeräusche zu finden sind. Das Herausfiltern von Hall und Nebengeräuschen wie Prozessorlüftern und Festplattengeräusche ist keineswegs trivial. Zum anderen ist die Zuordnung von Gesagtem zu Sprechern zu bewerkstelligen,

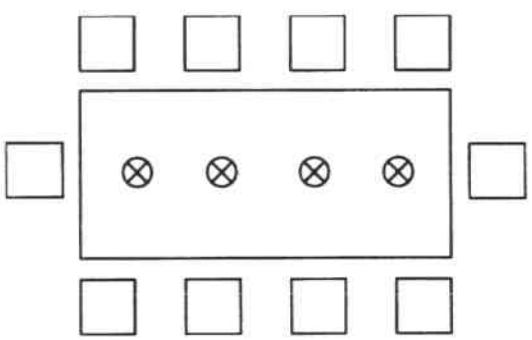


Abbildung 1.1: Meeting Situation - Ein Tisch, die Sprecher auf den Sitzplätzen um den Tisch angeordnet, die Mikrofone auf dem Tisch verteilt

eine Aufgabe, die es bei der Verwendung von Nahbesprechungsmikrofonen nicht in dem Umfang zu lösen gilt, ebenfalls wie die Behandlung von Cross-Talk-Regionen, also Zeitpunkte zu denen mehrere Personen sprechen und auf jeder Aufnahmespur mehrere Sprecher gleichzeitig zu hören sind. Eine Methode, Meetings zu erkennen besteht darin, jeden Kanal zu dekodieren und die Wordhypothesengraphen dann zu kombinieren. Nachteil dieser Methode ist, dass der Dekodiervorgang die meiste Zeit der Erkennung in Anspruch nimmt und bei dieser Methode mehrmals durchgeführt werden muss. Ziel dieser Arbeit ist es, durch eine numerisch nicht anspruchsvolle Vorverarbeitung der Daten nur einen Kanal dekodieren zu müssen. Diesen gewinnen wir über Delay and Sum mit geeigneten Parametern aus den gegebenen Kanälen. Ein weiteres Ziel ist das Erreichen niedrigerer Wortfehlerraten auf diesen gewonnenen Kanälen.

1.2 NIST Rich Transcription 2004 Spring Meeting Recognition Evaluation (RT-04S)

Diese Studienarbeit ist aus der NIST Rich Transcription 2004 Spring Meeting Recognition Evaluation [1] entstanden. Die Erkenner der beiden wichtigsten daran teilnehmenden Gruppen, ICSI und ISL, waren von der Erkennerleistung her ähnlich, sie unterschieden sich aber gravierend in der Zeitdauer des Erkennungsvorgangs voneinander (der ISL-Erkenner dekodierte jeden Kanal separat, was zu einer wesentlich längeren Zeitdauer führte).

1.2.1 Beschreibung

Die NIST Rich Transcription 2004 Spring Meeting Recognition Evaluation bestand aus zwei Tasks :

- Speech-to-Text Transcription tasks
  Bei dieser Aufgabe geht es darum, gesprochene Sprache in eine schriftlich fixierte Form zu übertragen.

- Diarization tasks
  Ziel dieser Aufgabe ist es, gesprochene Sprache Sprechern zuzuordnen.

In den für uns interessanten Speech-to-Text Transcription tasks wurde nochmals in Art der Mikrofone und Dauer des Erkennungsvorgangs gegliedert. Art der Mikrofone:

- mehrere Raummikrofone (Multiple distant microphones, MDM)

- ein einzelnes Raummikrofon (Single distant microphone, SDM)
- einzelne, sprecherzugeordnete Nahbesprechungsmikrofone (Individual head microphone, IHM)

Das Hauptaugenmerk lag auf dem MDM-Task.

Es gab drei verschiedene Datensätze. Mit den Trainingsdaten wurde der Erkenner trainiert, mit den Development-Testdaten wurde die Leistung während des Trainings getestet, der fertige Erkenner wurde dann mit dem Evaluations-Testdatensatz getestet.

Trainingsdaten

Für das Training könnten beliebige Daten verwendet werden, trotzdem stellte NIST einen Traininsdatensatz aus der entsprechenden Domäne zusammen, bestehend aus 11 Stunden CMU ISL Meeting Corpus, 72 Stunden ICSI Meeting Corpus und 13 Stunden NIST Pilot Meeting Corpus.

Development-Testdaten

Der Development-Testdatensatz [2] bestand je einem 10 Minuten Auszug von jeweils zwei Meetings der Datensammelgruppen die auch schon die Trainingsdaten bereitgestellt haben (CMU, ICSI, NIST), zusätzlich noch zwei 10 Minuten Auszügen von Meetings aus der LDC Datensammlung, zusammen 80 Minuten.

Evaluations-Testdaten

Der Evaluations-Testdatensatz [3] bestand aus acht 11 Minuten Auszügen, jeweils zwei von CMU, LDC, ICSI und NIST, zusammen 90 Minuten.

1.2.2 Ergebnisse

Der folgende Abschnitt gibt eine kleine Übersicht über die Ergebnisse der teilnehmenden Gruppen ICSI und CMU im MDM Speech to Transcription Task.

- ICSI-SRI-UW Spring 2004 Evaluation System [4]
  Beim trainierten ICSI-System durchläuft das Signal folgende Stufen:
  - Noise Reduction mittels Wiener filtering

- Segmentierung anhand des am zentralsten positionierten Mikrofons
- Delay and Sum, die Delays wurden mit der maximalen Kreuzkorrelation ermittelt, Basiskanal ist der zentralste Kanal.
- Dekodierung (inklusive Confusion Networks)
- Cross-Talk Unterdrückung, ermittelte Wörter aus Cross-Talk Regionen wurden aus den Hypothesen entfernt

Dieses System erreichte eine WER von 47 %, 6,54*RT (Mit RT (Realtime) wird die Länge der zu transkribierenden Audiodaten bezeichnet. Sollte es 30 Minuten dauern, ein 10 Minuten Audiodatensegment zu transkribieren würde das 3*RT entsprechen). Ein weiteres ICSI-System, welches einen zusätzlichen zweiten, anders konfigurierten Dekoder einsetzte erreichte eine WER von 44,9 %, 25*RT.

- ISL System
  Das trainierte ISL-System [5] setzt sich aus folgenden Stufen zusammen:
  - Segmentierung
  - Dekodierung mit adaptierten Modellen (ohne VTLN)
  - Confusion Network Combination
  - VTLN, erneute Adaption
  - Dekodierung mit neu-adaptierten Modellen
  - Confusion Network Combination

  Dieses System erreicht eine WER von 44,9 % bei einem Zeitfaktor von 259*RT. Dieser hohe Zeitfaktor ist darin begründet, dass bei diesem Erkenner jeder Kanal separat dekodiert wurde.

Kapitel 2

Der Erkenner und die Daten

Dieses Kapitel soll beschreiben, wie unser Erkenner trainiert wurde und mit welchen Daten wir arbeiten[6]. Für diese Arbeit konzentrieren wir uns auf den Speech-to-Text Transcription Task mit mehreren Raummikrofonen. Wir arbeiten mit dem ISL-Spracherkenner Janus und dem IBIS Single-Pass-Decoder auf 16 kHz ADCs, welche in einen auf MFCCs basierenden 42-dimensionalen Merkmalsraum transformiert werden.

2.1 Trainingsdaten

Unser Erkenner wurde mit Meeting-Daten der Datensammelgruppen ICSI, CMU und NIST trainiert. Diese Daten bestehen haupsächlich aus in professionellen oder Forschungsumgebungen aufgenommenen Meetings. Die Teilnehmer sitzen für gewöhnlich um einen Tisch. Wie für Meetings üblich enthalten Daten dieser Art spontane und unsaubere Sprache.

Korpus	Dauer	# Meetings	# Sprecher	# Raummikrofone
CMU	11h	21	93	0
ICSI	72h	75	455	4
NIST	13h	15	77	7

Tabelle 2.1: Umfang der für die Evaluation bereitgestellten Trainingsdaten

Zusätzlich zu den von ICSI bereitgestellten Trainingsdaten wurden für das Training unseres Erkenners noch 180 Stunden Broadcast News Daten von den Trainingsets 1996 und 1997 und für die Sprachmodellierung die Transkriptionen der Switchboard-Phasen "Cellphone" und "C-Tran" verwendet.

2.2 Testdaten

Als Testdaten verwendeten wir die 2 ICSI Datensätze [7] aus dem Development-Test-Datensatz. Alle Erkennerdurchläufe über die kompletten Daten laufen zu lassen hätte zuviel Zeit und Rechenleistung in Anspruch genommen.

2.2.1 Die ICSI - Datensätze

ICSI_20010208-1430 wurde am 08.02.2001 um 14.30 Uhr, ICSI_20010322-1450 am 22.03.2001 um 14.50 Uhr im ICSI Meeting Labor aufgezeichnet. An jedem Meeting sind mehrere Sprecher beteiligt.

Original Meeting ID	RT-02 Meeting ID	RT-04s Meeting ID
Bmr013	b013	ICSI_20010208-1430
Bmr018	b018	ICSI_20010322-1450

Tabelle 2.2: Mapping der ICSI Datensätze

Meeting	Sprecher	Geschlecht	Muttersprache
ICSI_20010208-1430	mn014	männlich	nicht englisch
ICSI_20010208-1430	fe008	weiblich	englisch
ICSI_20010208-1430	me013	männlich	englisch
ICSI_20010208-1430	me018	männlich	englisch
ICSI_20010208-1430	me001	männlich	englisch
ICSI_20010208-1430	me011	männlich	englisch
ICSI_20010208-1430	fe016	weiblich	englisch
ICSI_20010322-1450	fe008	weiblich	englisch
ICSI_20010322-1450	me018	männlich	englisch
ICSI_20010322-1450	me013	männlich	englisch
ICSI_20010322-1450	fe016	weiblich	englisch
ICSI_20010322-1450	me011	männlich	englisch
ICSI_20010322-1450	mn017	männlich	nicht englisch
ICSI_20010322-1450	me001	männlich	englisch

Tabelle 2.3: Übersicht der Sprecher

2.2.2 Segmentierung

Wir haben bei dieser Arbeit mit zwei Arten von Segmentierung gearbeitet.

Automatische Segmentierung

Die automatische Segmentierung [8] besteht aus 6 Schritten :

1. Initiale Unterteilung in Sprachsegmente und Nichtsprachsegmente mit einem modifizierten CMUseg_0.5 [9]
2. Vereinigung der Segmente mehrerer Kanäle
3. Auswahl des besten Kanals
4. Erkennung von Sprecherwechseln bei Segmenten die länger als 5s sind
5. Sprecher-Clustering
6. Glätten

Die automatisch segmentierten Daten enthalten Cross-Talk-Regionen, die allerdings aus dem Scoring ausgenommen sind. Dennoch kann ein falsch transkribiertes, von Scoring ausgenommenes Segment sich wegen des Sprachmodells negativ auf die folgenden Segmente, die wieder gescored werden, auswirken.

Manuelle Segmentierung

Die manuelle Segmentierung enthält nur Utterances, bei denen jeweils nur ein Sprecher spricht. Cross-Talk-Regionen sind in dieser Segmentierung ausgeklammert. Desweiteren ist zu jedem Zeitpunkt klar, welcher Sprecher gerade spricht. Diese Segmentierung wurde aus den Transkriptionen, die für das Scoring verwendet werden und somit keine Cross-Talk-Regionen enthalten, erstelllt. Der Unterschied der beiden Segmentierungen ist in Abbildung 2.1 dargestellt. Sprecher A beginnt zum Zeitpunkt t_{StartA}, zum Zeitpunkt t_{StartB} setzt Sprecher B ein (während Sprecher A noch spricht). Zum Zeitpunkt t_{EndeA} spricht nur noch Sprecher B. Die beiden automatischen Segmentierungen enthalten auch die Cross-Talk-Region zwischen t_{StartB} und t_{EndeA}. Segmentierung 2 ordnet Sprecher B zu zwei verschiedenen Sprechern zu. Die manuelle Segmentierung enthält nur die Region t_{StartA}-t_{StartB} und die Region t_{EndeA}-t_{EndeB}.

Beide Arten wurden untersucht, um den Einfluss der Segmentierung auf die Qualität der Kanalkombination zu erfassen.

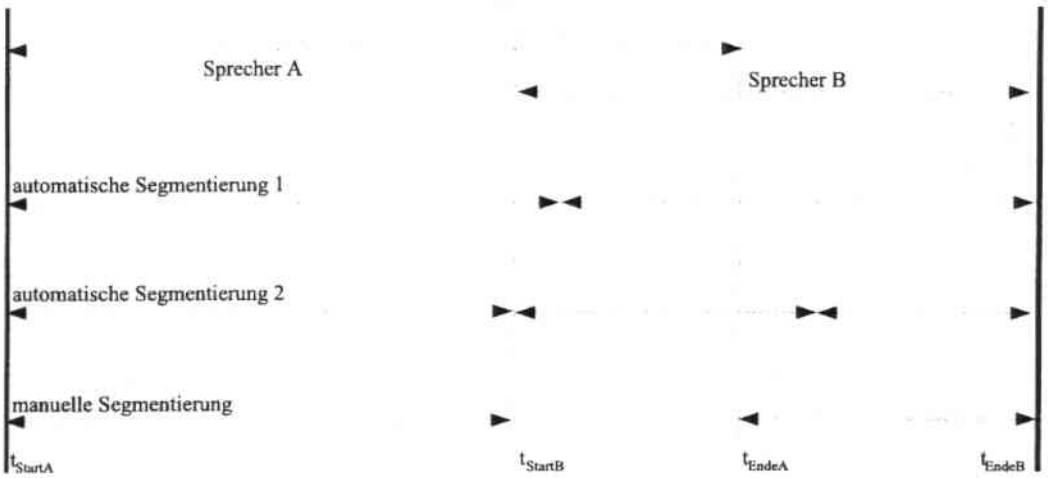


Abbildung 2.1: Unterschied automatische / manuelle Segmentierung

2.3 Akustische Modelle

Unser Erkenner arbeitet mit zwei Arten von akustischen Modellen. Das genaue Modell-Training kann in [6] nachgelesen werden.

2.3.1 Plain Modelle

Die plain Modelle wurden mit den Daten von ICSI, CMU, NIST und den Broadcast News Daten trainiert. Zwei zusätzliche Viterbi Iterationen wurden auf den jeweiligen Nahbesprechungsmikrofondaten gerechnet.

2.3.2 Adaptierte Modelle

Die adaptierten Modelle unterscheiden sich von den plain Modellen dadurch, dass sie auf die Raummikrofondaten von ICSI und NIST adaptiert wurden. Es wurde sowohl ein sprecher-adaptives Training (SAT) als auch ein Kanal-adaptives Training (CAT) durchgeführt. Desweiteren wird eine Vokaltraktlängennormierung (VTLN) angewendet.

Kapitel 3

Werkzeuge

3.1 Delay and Sum Beamforming

Normalerweise sind verschiedene Sprecher von verschiedenen Mikrofonen unterschiedlich weit entfernt. Das hat zur Folge dass ein Signal von Sprecher A von Mikrofon 1 zu einem anderen Zeitpunkt aufgezeichnet wird als von Mikrofon 2. Sind die Positionen der Sprecher und der Mikrofone bekannt, so kann man für jeden Sprecher den Laufzeitunterschied von zwei Mikrofonen berechnen. Verschiebt man nun das Signal eines der beiden Mikrofone um diesen Laufzeitunterschied und addiert dann die beiden Signale aufeinander, so wird alles, was aus der Richtung des Sprechers, zu dem der Laufzeitunterschied berechnet wurde im Vergleich zu den Signalen von anderen Orten verstärkt. Wir haben also die Möglichkeit, ein wenig in diese Richtung zu hören. Bei dieser Arbeit sind weder die Positionen der Mikrofone noch die Aufenthaltsorte der Sprecher bekannt. Wir wollen uns hauptsächlich damit beschäftigen, wie wir die für das Delay and Sum notwendigen Laufzeitunterschiede (Delays) herausfinden können, um dann die Audiodaten so zu bearbeiten, dass wir stets in die Richtung hören aus der der momentane Sprecher spricht um somit dem Decoder zu besseren Ergebnissen zu verhelfen.

3.2 Berechnung der Laufzeitunterschiede

Ein Verfahren zur Mustererkennung in Features ist die Korrelation. Sie liefert uns den Wert zurück, um den das eine Feature verschoben werden muss, um bestmöglich mit dem anderen Feature übereinzustimmen. Wir stellen ausserdem die Additionsmethode vor, die uns helfen soll, unter vielen potentiellen Delays den Richtigen auszuählen. Darüber hinaus haben wir auf unsere Daten einige Filter angewandt. Ziel ist es, die Daten so zu filtern, dass

die Korrelation den richtigen Delay liefert.

3.2.1 Korrelation

Mit *seg* bezeichnen wir ein Segment von Audiodaten.
Vom Segmentierer festgelegte Segmente bezeichnen wir als Utterances *utt*.
Die Menge aller *seg* nennen wir SEG, die Menge aller *utt* UTT.
Mit dx oder dy bezeichnen wir einen der 4 Kanäle $d01$, $d02$, $d05$ oder $d06$.
Das Korrelationsspektrum als Ergebnis einer Korrelation von einer Utterance *utt* zweier Kanäle dx und dy bezeichnen wir mit C oder $C(dx, dy, utt)$.

Die Korrelation zweier Features A und B im Intervall [from, to] erzeugt eine $1 \times (1 + (\text{from-to}))$ Matrix C, die als Koeffizienten

$$c_{0,i} = \frac{1}{m} \sum_{k=1}^{m} a_k b_{k-i}, m = \min(\dim(A), \dim(B))$$

enthält. Die Matrix zeigt an, wie sehr Feature A mit dem um i verschobenen Feature B übereinstimmt. Im Idealfall entspricht die Stelle des maximalen Wertes von C genau dem Wert, um den man Feature B für ein Delay and Sum mit Feature A verschieben müsste um in die Richtung zu hören aus der das Signal zu diesem Zeitpunkt kam. Leider tritt dieser Idealfall selten ein. Oft entspricht die Stelle des Maximums von C nicht dem gewünschten Delay. Das kann mehrere Ursachen haben. Das Signal ist durch Nebengeräusche wie Lüfter gestört, welche teilweise den Signalanteil des Sprechers übertreffen, was in diesem Fall zu einem Delay führen würde, mit dem in die Richtung der Störquelle gehört werden würde. Wir haben für diese Arbeit sowohl Korrelationen auf Utterances berechnet als auch Korrelationen auf Teilen von Utterances (pro Utterance $n = \lceil length_{utt} \rceil$ Segmente gleicher Länge $l = \frac{length_{utt}}{\lceil length_{utt} \rceil}$).

3.2.2 Additionsmethode

Mit $t(dx, dy, spk)$ bezeichnen wir einen Laufzeitunterschied, also die Zeit, die ein Signal, das von Sprecher *spk* ausgeht, benötigt, um vom Mikrofon das Kanal dx aufzeichnet zum Mikrofon das Kanal dy aufzeichnet zu gelangen.
$s(dx, dy, seg)$ ist die Stelle, an der das Korrelationsspektrum $C(dx, dy, seg)$ seinen maximalen Wert hat.
$spk(seg)$ gibt den Sprecher zurück, der in diesem Segment gesprochen hat. Wir betrachten nur Segmente bei denen wir davon ausgehen dass nur ein Sprecher gesprochen hat.

$s(dx, dy, seg)$ ist, was uns die Korrelation liefert, $t(dx, dy, spk)$ ist, was wir gerne hätten. Im Idealfall ist $s(dx, dy, seg) = t(dx, dy, spk(seg))$.

Da wir 4 Audiokanäle zu Verfügung haben gibt es insgesamt 6 verschiedene Kanalkombinationen, mit denen wir Korrelationen berechnen können. Zwischen jeweils 3 Kanälen (zB. d01, d02 und d05) und den dazugehörigen Laufzeitunterschieden ($t(d01, d02, seg)$, $t(d02, d05, seg)$ und $t(d01, d05, seg)$) besteht folgender Zusammenhang :

$$t(d01, d02, seg) + t(d02, d05, seg) = t(d01, d05, seg) \text{ oder allgemein}$$
$$t(dx, dy, seg) + t(dy, dz, seg) = t(dx, dz, seg)$$

Ist dieser Zusammenhang bei 3 Delays die über die Korrelation ermittelt wurden nicht gegeben so können wir uns sicher sein, dass mindestens einer der 3 Delays falsch ist. Umgekehrt können wir, wenn wir uns sicher sind dass

$$\begin{aligned} s(dx, dy, seg) &= t(dx, dy, spk(seg)) \text{ und} \\ s(dy, dz, seg) &= t(dy, dz, spk(seg)) \text{ gilt,} \\ s(dx, dy, seg) + s(dy, dz, seg) &= s(dx, dz, seg) \\ &= t(dx, dz, spk(seg)) \text{ berechnen.} \end{aligned}$$

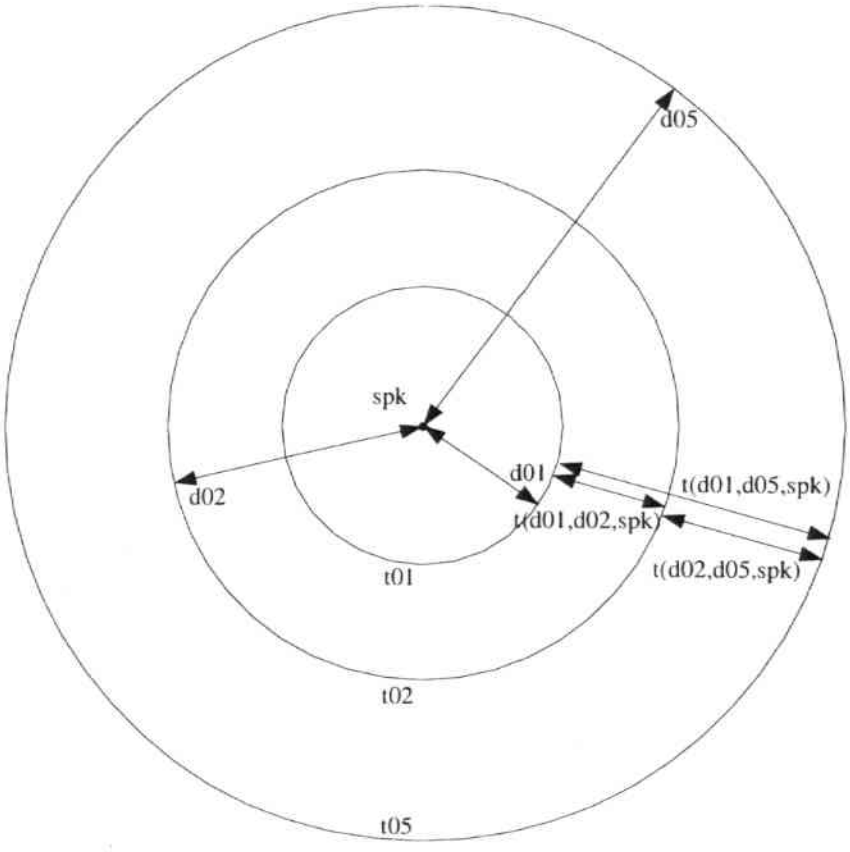


Abbildung 3.1: Die Additionsmethode : Das Signal das von der Quelle spk ausgeht mit seiner Position zu den Zeiten t01,t02 und t03. Die Laufzeitunterschiede t(d01,d02,seg) und t(d02,d05,seg) lassen sich auf t(d01,d05,seg) addieren

3.2.3 Filter

Mit F bezeichnen wir eine Matrix, mit der das Signal gefiltert wird.
Die Delays berechnen sich dann $s(dx, dy, seg) = \text{argmax}\,[C(dx, dy, seg) - k * F_x]$.
Mit k bezeichnen wir eine Skalierung die garantiert, dass C und F im richtigen Verhältnis in die Differenz eingehen.

Wir unterscheiden bei den Filtern Korrelationsspektren F und geglättete Korrelationsspektren $\tilde{F}$. Geglättet wird F, indem jeder Korrelationskoeffizient f_i durch $\tilde{f}_i = \frac{1}{7}\sum_{k=i-3}^{i+3} f_k$ ersetzt wird.

Da bei Meetings keine Nahbesprechungsmikrofone verwendet werden sind auf den Aufnahmen vermehrt Nebengeräusche wie Hall oder Rechnergeräusche vorhanden, die die Ermittlung der korrekten Laufzeitunterschiede erschweren. Korrelierte Nebengeräusche finden sich auch in den Korrelationsspektren wieder. Wir wenden verschiedene Filter F auf die lokalen Korrelationsspektren C an, um die Nebengeräusche zu reduzieren.

[10] schlägt vor, zu diesem Zweck das Korrelationsspektrum eines Segments ohne Sprecher von den lokalen Korrelationsspektren zu subtrahieren. Wir testen die Anwendung von Filtern, die aus Segmenten mit Sprechern ermittelt wurden.

Geglättete lokale Korrelationsspektren

Eine mögliche Art zu filtern ist die Subtraktion des geglätteten lokalen Korrelationsspektrums vom lokalen Korrelationsspektrum.

$$\tilde{F}_{lokal} = \tilde{C}(dx, dy, utt)$$

Da durch das Glätten des lokalen Korrelationsspektrums scharfe Peaks abgeschwächt werden sind genau diese in der Differenz wiederum besonders ausgeprägt. $\tilde{F}_{lokal}$ ist keine Summe aus Korrelationsspektren, somit kann $k = 1$ gesetzt werden. Die Wirkung eines $\tilde{F}_{lokal}$-Filters ist in Abbildung 3.2 dargestellt.

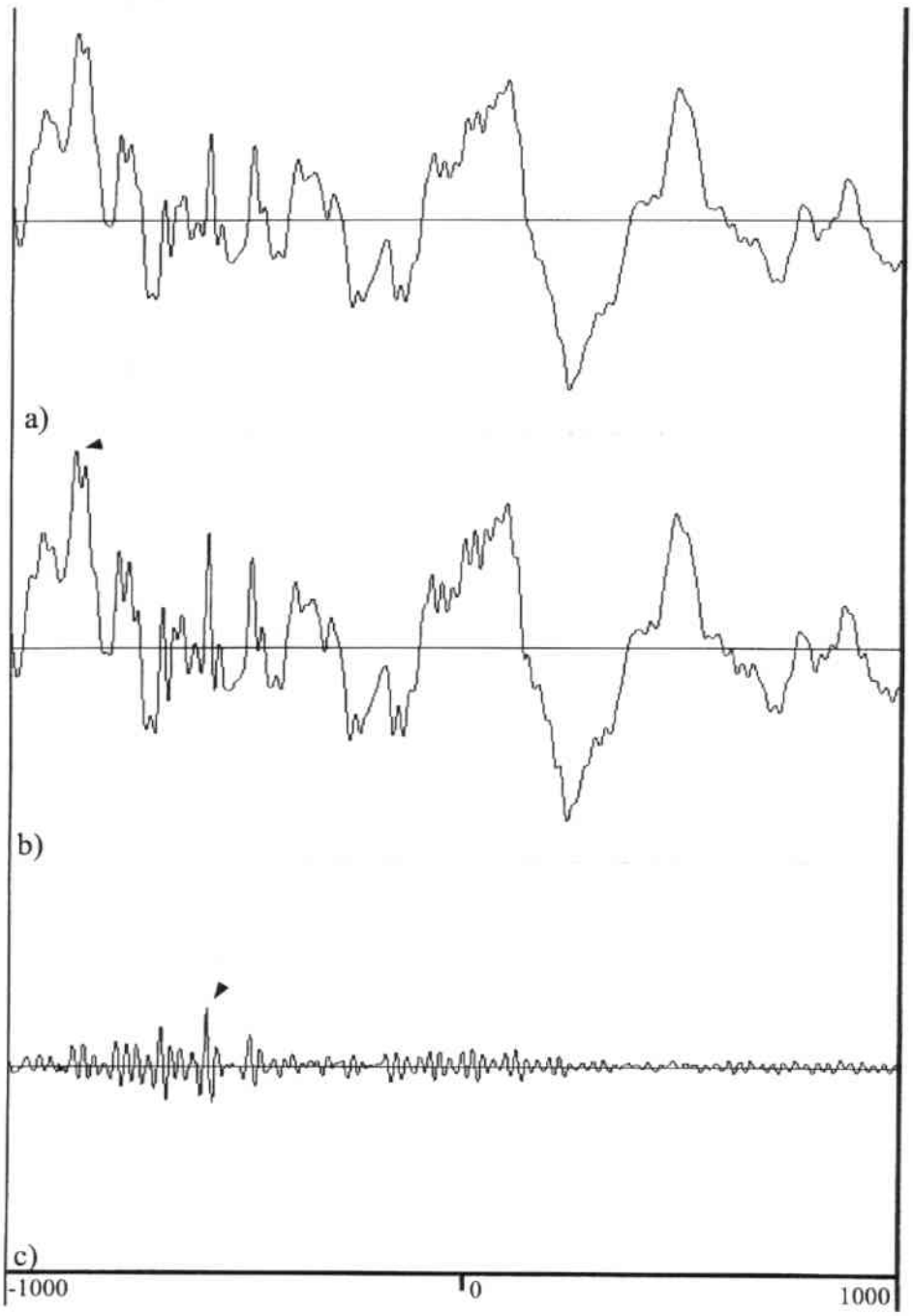


Abbildung 3.2: a) das lokale Korrelationsspektrum mit dem Maximum $s(dx, dy, utt)$, b) das geglättete Korrelationsspektrum, c) das gefilterte, lokale Korrelationsspektrum mit dem Maximum $s(dx, dy, utt) = t(dx, dy, spk(utt))$

Globale, sprecherabhängige Korrelationsspektren

Wie im Abschnitt Korrelation beschrieben betrachten wir zwei Fälle, zum einen ein Korrelationsspektrum pro Utterance, zum anderen mehrere Korrelationsspektren pro Utterance.

- Fall 1 ($F_{global\_spk\_utt}$): Wenn man alle Korrelationsspektren von Utterances eines Sprechers *spk* addiert, erhält man das globale Korrelationsspektrum des Sprechers spk :

$$F_{global\_spk\_utt}(dx, dy, spk) = \sum_{utt \in UTT(spk)} C(dx, dy, utt)$$

- Für Fall 2 ($F_{global\_spk\_seg}$) werden dementsprechend mehrere Korrelationsspektren pro Utterance addiert.

$F_{global\_spk\_utt}, F_{global\_spk\_seg}, \tilde{F}_{global\_spk\_utt}$ und $\tilde{F}_{global\_spk\_seg}$ sind Summen von Korrelationsspektren und müssen somit normiert werden :

$$k = \frac{\sum_{i=0}^{\dim C} c_i}{\sum_{i=0}^{\dim F} f_i}$$

bzw.

$$k = \frac{\sum_{i=0}^{\dim C} c_i}{\sum_{i=0}^{\dim \tilde{F}} \tilde{f}_i}$$

In Abbildung 3.3 ist die Anwendung eines $\tilde{F}_{global\_spk\_utt}$ - Filters dargestellt. Im ungefilterten Korrelationsspektrum liegt $s(dx, dy, utt)$ in der Nähe von $t(dx, dy, spk(utt))$. Durch die Anwendung des Filters wird $s(dx, dy, utt) = t(dx, dy, spk(utt))$ ermittelt.

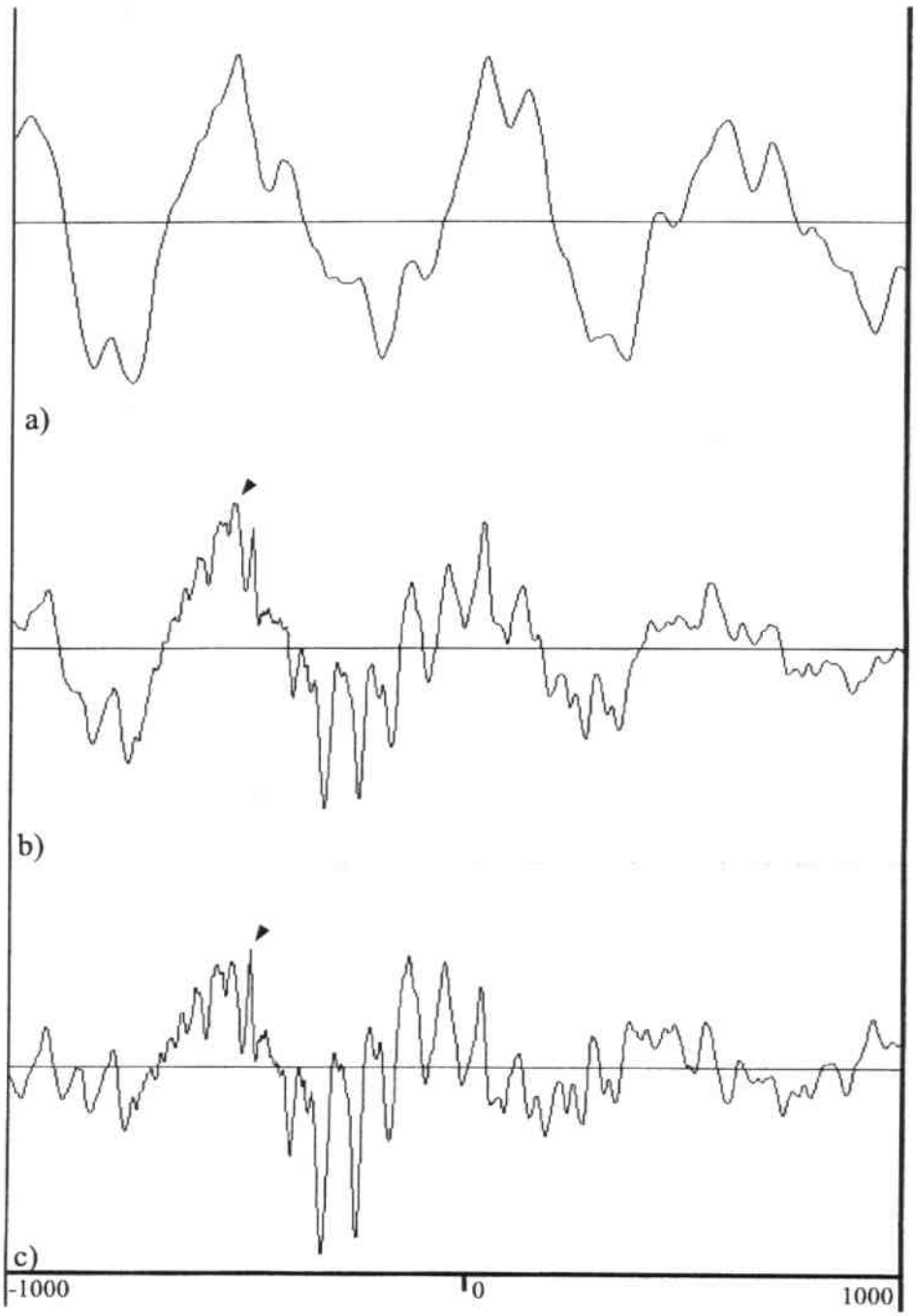


Abbildung 3.3: a) Ein globales, geglättetes Korrelationsspektrum eines Sprechers spk, b) das lokale Korrelationsspektrum, c) das gefilterte, lokale Korrelationsspektrum

Globales, sprecherunabhängiges Korrelationsspektrum

Dieser Filter unterscheidet sich vom davor beschriebenen dadurch, dass nicht nur die Korrelationsspektren von Utterances eines Sprechers addiert werden. Jedes lokale Korrelationsspektrum wird mit der Summe von Korrelationsspektren aller Utterances des Recordings gefiltert.

$$F_{global\_all\_utt} = \sum_{utt \in UTT} C(dx, dy, utt)$$

Für k gilt wie auch bei den globalen, sprecherabhängigen Korrelationsspektren :

$$k = \frac{\sum_{i=0}^{\dim C} c_i}{\sum_{i=0}^{\dim F} f_i}$$

bzw.

$$k = \frac{\sum_{i=0}^{\dim C} c_i}{\sum_{i=0}^{\dim \tilde{F}} \tilde{f}_i}$$

Abbildung 3.4 zeigt die Anwendung eines $F_{global\_all\_utt}$-Filters. Im ungefilterten Korrelationsspektrum liegt $s(dx, dy, utt)$ in der Nähe von $t(dx, dy, spk(utt))$. Durch die Anwendung des Filters wird $s(dx, dy, utt) = t(dx, dy, spk(utt))$ ermittelt.

Die Filterung sowohl mit sprecherabhängigen als auch mit sprecherunabhängigen globalen Korrelationsspektren führt nur selten zum Erfolg, eine positive Wirkung der globalen Filter wie in den beiden Beispielen ist nur sehr vereinzelt zu beobachten (mehr dazu im Kapitel Experimente).

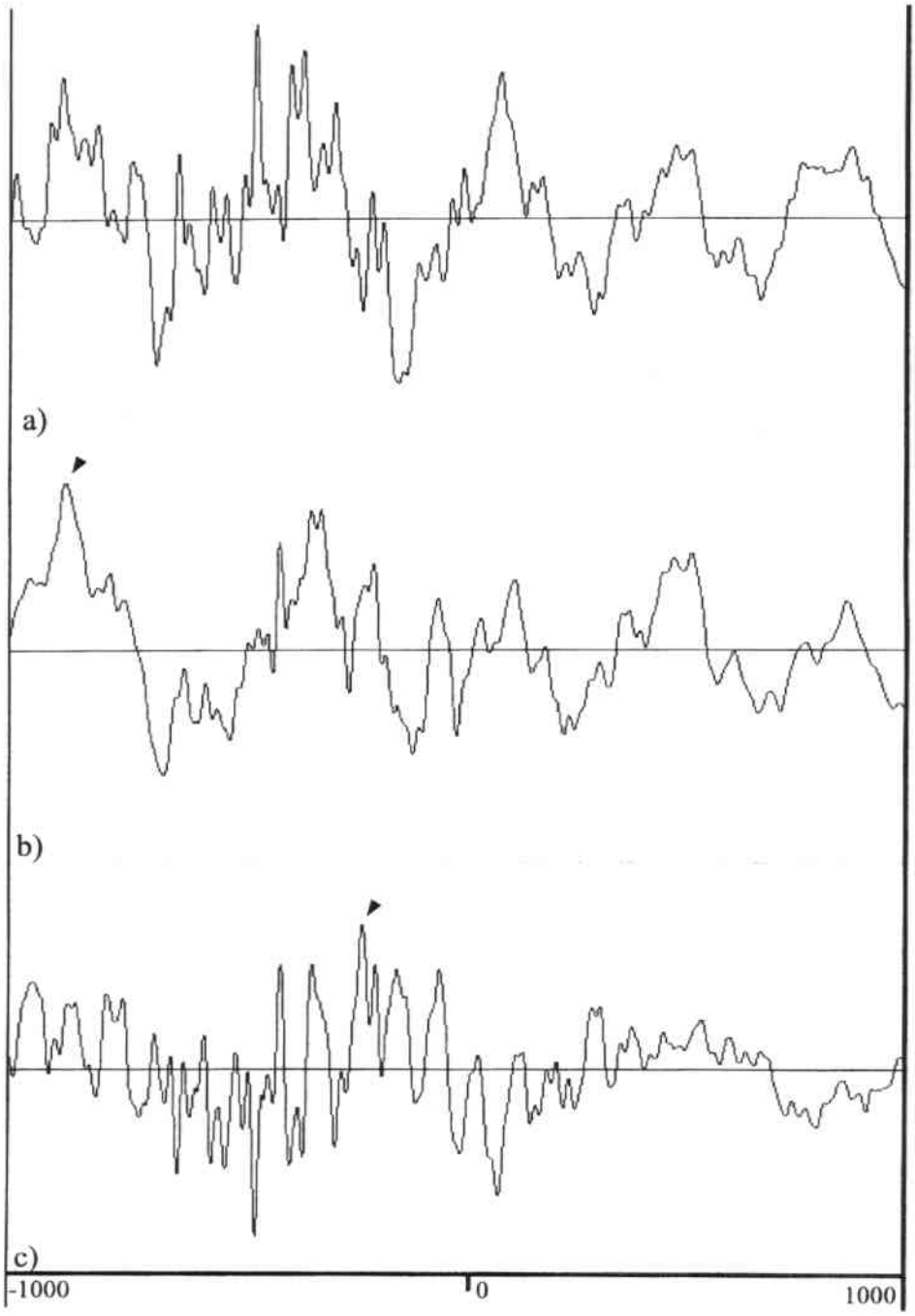


Abbildung 3.4: a) Das globale, sprecherunabhängige Korrelationsspektrum, b) das lokale Korrelationsspektrum, c) das gefilterte, lokale Korrelationsspektrum

Kapitel 4

Experimente

Dieses Kapitel gibt eine Übersicht über die Experimente, die wir durchgeführt haben.

- E1 : 4 Kanal Delay and Sum, Filter : Wir berechnen für jeden Kanal die Korrelation zum angegebenen Basiskanal und führen mit dem ermittelten $s(d_{Basis}, dy, utt)$ ein Delay and Sum durch. Desweiteren testen wir die Anwendung der verschiedenen Filter vor der Berechnung der Korrelation.
- E2.1 : Globale Maxima, lokale Korrektur : Wir bestimmen für jeden Sprecher globale Delays, die dann für jede einzelne Utterance lokal korrigiert werden.
- E2.2 : Globale Maxima, verfeinerte lokale Korrektur mit Additionsmethode : Wir nutzen die Additionsmethode, um die lokale Korrektur noch zu verfeinern.
- E3 : Delay and Sum ohne Angabe eines Basiskanals : Über die Additionsmethode wählen wir für jeden Sprecher die 3 Kanäle aus, die für das Delay and Sum verwendet werden sollen. Für dieses Experiment benötigen wir keinen Basiskanal.

4.1 Baseline

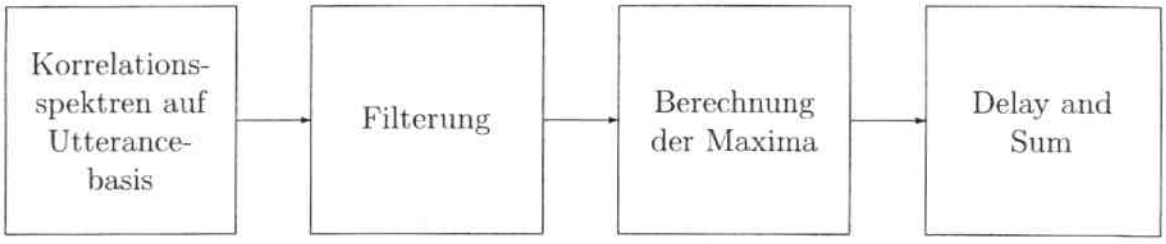
Für das Baselineexperiment wurde mit dem in Kapitel 2 beschriebenen Erkenner jeder der 4 Audiokanäle separat dekodiert. Es gab keine zusätzliche Vor- oder Nachbereitung der Daten.

Basiskanal	d01	d02	d05	d06	$\bar{d}$
automatische Segmentierung, plain Modelle	47,8	54,5	48,4	42,1	48,2
aut. Segmentierung, adaptierte Modelle	35,9	39,9	35,7	31,5	35,8
manuelle Segmentierung, plain Modelle	48,9	57,0	49,1	43,4	49,6
man. Segmentierung, adaptierte Modelle	33,0	36,1	33,5	31,5	33,5

Tabelle 4.1: Wortfehlerrate der Baselines, abhängig von Segmentierung und Art der akustischen Modelle, Spalte ganz rechts der Durchschnitt $\bar{d}$

4.2 4 Kanal Delay and Sum, Filter (E1)

Diese Reihe von Experimenten zeigt die Auswirkung vom Einsatz von Filtern auf die Korrelationsspektren.
Mit $S(dx, dy, utt)$ bezeichnen wir den Delay, mit dem wir den Kanal dy auf den Kanal dx addieren.



Basiskanal	d01	d02	d05	d06
DnS ohne Filter	36,5	40,9	43,4	38,0
$\tilde{F}_{lokal}$	33,3	35,6	34,6	35,5
$\tilde{F}_{global\_spk\_utt}$	43,1	49,6	49,5	44,0
$F_{global\_spk\_utt}$	43,2	49,8	49,4	44,0
$\tilde{F}_{global\_spk\_seg}$	44,9	53,9	51,9	46,6
$F_{global\_spk\_seg}$	45,9	54,6	53,1	48,0
$\tilde{F}_{global\_all\_utt}$	49,6	43,2	48,3	46,5
$F_{global\_all\_utt}$	49,5	43,2	48,6	46,4

Tabelle 4.2: Wortfehlerraten der Delay and Sum Experimente, abhängig von den eingesetzten Filtern auf manueller Segmentierung mit plain Modellen

Durch ein einfaches Delay and Sum können die Wortfehlerraten im Schnitt schon um 7,8% (auf manueller Segmentierung mit plain Modellen) gesenkt werden, mit dem Einsatz eines Filters, der von jedem lokalen Korrelationsspektrum das jeweils geglättete subtrahiert erhöht sich die Worterkennungsrate sogar um 15%. Am meisten wirkt sich dieser Filter bei der Arbeit mit

Kanal d05 aus, während sich die Wortfehlerrate bei einem Delay and Sum mit d05 als Basiskanal gar nicht verändert sinkt sie doch nach Abzug der jeweils geglätteten lokalen Spektren um 8,8%.

Filtern mit globalen Korrelationsspektren führt zu einer höheren Wortfehlerrate. Wir gehen davon aus, dass sich die Position eines Sprechers über die Dauer einer Aufnahme kaum ändert (sonst könnten wir nicht mit globalen Maxima arbeiten). Daher hat $\hat{C}$ seinen höchsten Peak in einem Bereich, in dem die meisten lokalen Korrelationsspektren ihren maximalen Peak haben. Da in vielen Fällen das aber genau die Stelle sein wird, die wir suchen führt eine Subtraktion dieses Filters in vielen Fällen zu einem unerwünschten Ergebnis (Abbildung 4.1).

An Tabelle 4.3 kann man jedoch sehen, dass es einzelne Sprecher gibt, für die einer der Filter bessere Ergebnisse bringt als ein Delay and Sum ohne vorherige Anwendung von Filtern.

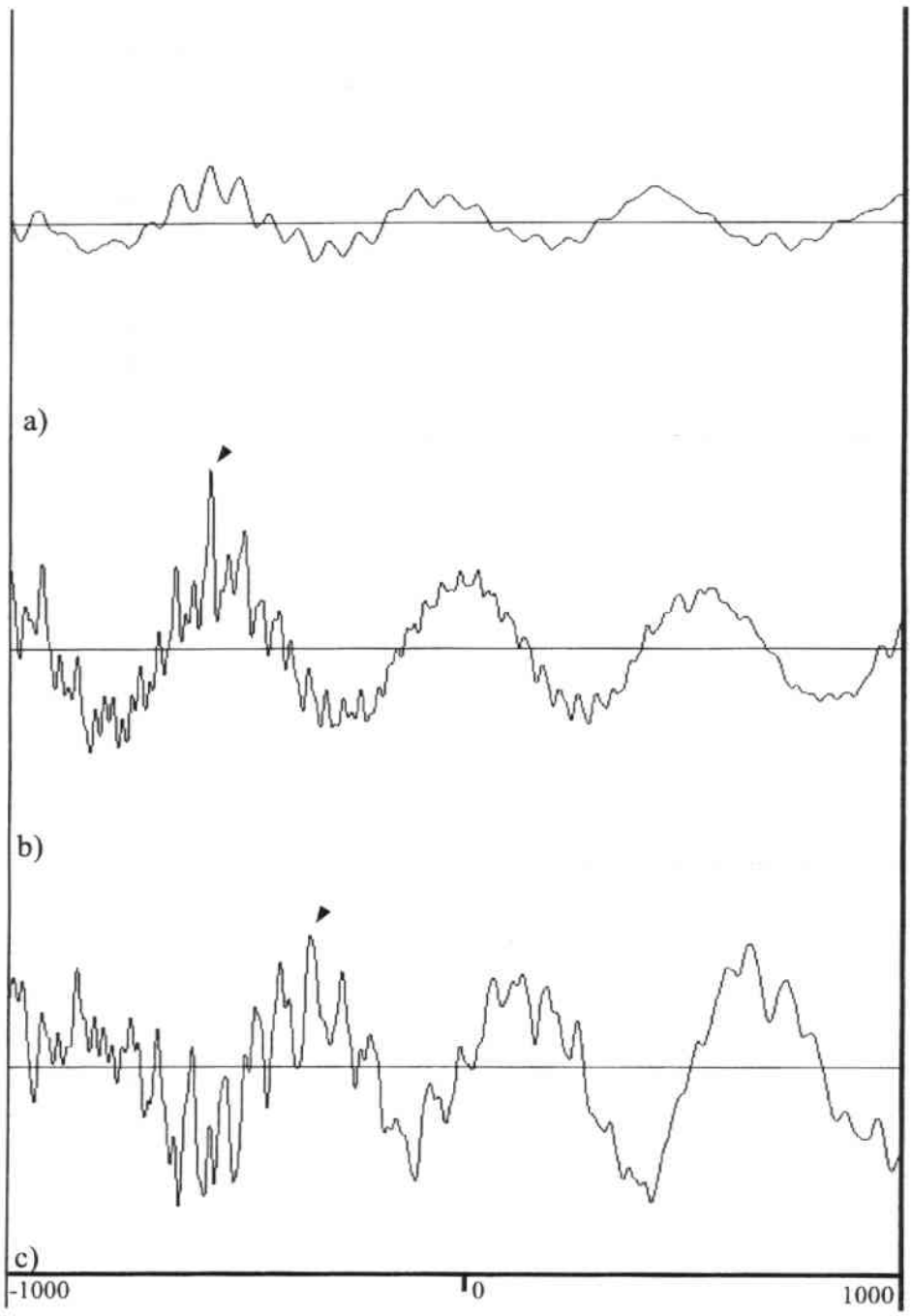


Abbildung 4.1: Das Filtern (a) führt vom ursprünglich richtig ermittelten Maximum (b) weg (c)

	Basisk.	spk1	spk2	spk3	spk4	spk5	spk6	spk7	spk8	spk9	spk10
Baseline	d01	38.8	53.3	52.9	29.0	30.0	32.1	52.8	34.8	40.9	**22.0**
	d02	43.9	52.0	65.9	32.6	30.0	38.1	53.7	32.6	47.0	17.1
	d05	58.1	40.0	57.6	27.1	30.0	36.7	55.6	30.4	61.1	**31.7**
	d06	45.3	41.3	57.6	27.9	30.0	33.0	47.2	37.0	46.3	**24.4**
$\tilde{F}_{global\_utt\_spk}$	d01	47.5	**52.0**	75.3	32.6	30.0	40.1	63.9	52.2	**37.6**	**22.0**
	d02	57.8	54.7	**64.7**	34.0	30.0	56.9	**51.9**	43.5	48.3	**24.4**
	d05	58.9	42.7	65.9	30.7	60.0	50.0	58.3	52.2	65.8	**43.9**
	d06	**44.7**	54.7	69.4	30.9	70.0	43.1	57.4	30.4	52.3	65.9
$\hat{F}_{global\_utt\_spk}$	d01	47.0	**52.0**	75.3	32.6	30.0	41.1	63.9	52.2	**37.6**	**22.0**
	d02	57.8	54.7	**64.7**	34.0	30.0	58.0	**51.9**	43.5	48.3	**24.4**
	d05	58.9	42.7	65.9	30.4	60.0	49.8	58.3	52.2	65.8	**43.9**
	d06	**44.7**	54.7	69.4	30.9	70.0	43.3	57.4	30.4	52.3	65.9
$\tilde{F}_{global\_seg\_spk}$	d01	47.5	**50.7**	75.3	33.2	40.0	45.6	63.0	41.3	41.6	39.0
	d02	57.8	57.3	**64.7**	41.0	50.0	56.4	**50.0**	47.8	71.8	53.7
	d05	58.9	42.7	65.9	33.6	40.0	51.1	**53.7**	65.2	72.5	85.4
	d06	**44.7**	49.3	69.4	36.2	30.0	47.0	53.7	54.3	53.7	82.9
$\hat{F}_{global\_seg\_spk}$	d01	47.0	**50.7**	75.3	33.6	60.0	47.2	63.0	54.3	43.6	39.0
	d02	57.8	58.7	**64.7**	41.9	60.0	57.3	**50.0**	56.5	71.1	53.7
	d05	58.9	49.3	65.9	34.7	60.0	52.5	**53.7**	69.6	73.8	87.8
	d06	**44.7**	57.3	69.4	37.4	70.0	47.9	52.8	63.0	56.4	82.9
$\tilde{F}_{global\_utt\_all}$	d01	55.1	54.7	67.1	37.8	40.0	49.8	61.1	39.1	56.4	34.1
	d02	51.5	57.3	**58.8**	33.2	50.0	**33.9**	57.4	**30.4**	54.4	39.0
	d05	**57.4**	45.3	63.5	32.6	30.0	46.1	62.0	39.1	63.8	46.3
	d06	58.9	56.0	57.6	29.6	30.0	44.3	53.7	39.1	58.4	31.7
$\hat{F}_{global\_utt\_all}$	d01	55.1	54.7	67.1	37.8	60.0	49.8	57.4	39.1	56.4	34.1
	d02	51.5	57.3	**58.8**	33.0	60.0	**33.9**	57.4	**30.4**	54.4	39.0
	d05	**57.6**	45.3	63.5	32.8	60.0	46.1	62.0	39.1	63.8	46.3
	d06	58.7	56.0	57.6	29.6	30.0	44.3	52.8	39.1	58.4	31.7

Tabelle 4.3: Fettgedruckt bedeutet Verbesserung des Sprechers durch Filter.

4.3 Globale Maxima und lokale Korrektur (E2)

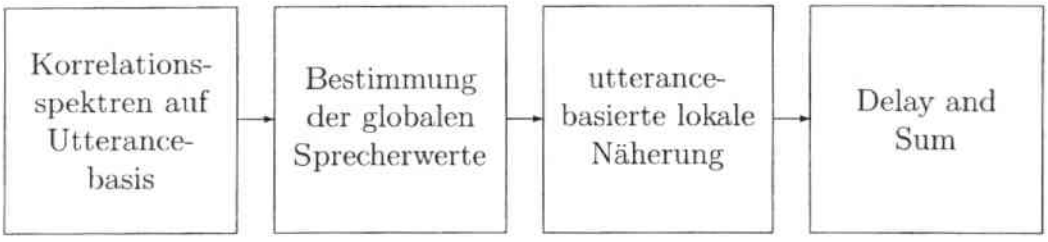
4.3.1 Lokale Maxima in Nähe der globalen Maxima als Delays für Delay and Sum (E2.1)

Bei diesem Experiment werden die 3 restlichen Kanäle auf einen vorher angegebenen Kanal per Delay and Sum aufaddiert. Als Delays werden lokale Maxima in Nähe der globalen Maxima verwendet.

Für dieses Experiment benötigen wir ebenfalls die Angabe des Basiskanals. Alle Korrelationen werden zwischen diesem Kanal und den anderen Kanälen errechnet. Sei der Basiskanal Kanal d02. Nun werden zuerst globale Sprecherinformationen ermittelt. Dazu werden für alle Utterances $C(d02, d01, utt)$, $C(d02, d05, utt)$ und $C(d02, d06, utt)$ berechnet und dann aufaddiert.

$$\hat{C}(dx, dy, spk) = \sum_{utt \in UTT(spk)} C(dx, dy, utt)$$

Resultat sind für jeden Sprecher $spk \in$ SPK 3 globale Korrelationsspektren $\hat{C}$. Die Stelle des maximalen Wertes jedes Korrelationsspektrums wird als globaler Delay der zwei Kanäle dx und dy von Sprecher spk, $\hat{s}(dx, dy, spk)$ bezeichnet. Zu beachten ist, dass $\hat{s}$ nicht wie s utt oder seg als Argument hat, sondern spk, da die globalen Sprecherinformationen vom Sprecher und nicht von einem Segment abhängen. Sind für alle Sprecher alle s berechnet wird für jede Utterance das am nächsten an $\hat{s}(dx, dy, spk)$ liegende Maximum im lokalen Korrelationsspektrum $C(dx, dy, utt)$ ermittelt. Mit den 3 auf diese Art und Weise berechneten Delays werden nun für den Dekodiervorgang dieser Utterance die 4 Kanäle mittels Delay and Sum addiert.



Ein grosser Nachteil der bereits vorgestellten Experimente ist die Angabe des Basiskanals. Versuche, mittels maximaler Autokorrelation den besten Basiskanal zu bestimmen schlugen fehl. Wie man sehen kann hängt die WER sehr von der Wahl des Basiskanals ab (Es gibt Unterschiede in der WER von bis zu 10% bei automatischer Segmentierung und plain Modellen). Das hängt damit zusammen dass die Korrelationen immer zum Basiskanal berechnet werden.

Basiskanal	d01	d02	d05	d06
automatische Segmentierung, plain Modelle	42,5	52,1	46,2	42,4
automatische Segmentierung, adaptierte Modelle	33,9	37,6	35,6	33,7
manuelle Segmentierung, plain Modelle	35,4	40,3	40,4	37,1
manuelle Segmentierung, adaptierte Modelle	27,3	30.2	30.1	28.8

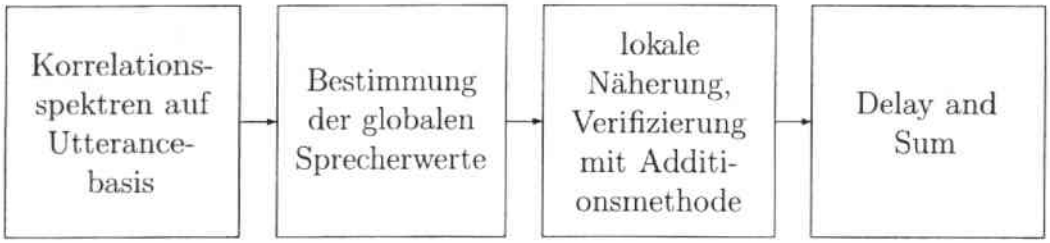
Tabelle 4.4: Wortfehlerrate in Abhängigkeit von Basiskanal, Segmentierung und Art der akustischen Modelle

Die schlechten Ergebnisse des Versuchs mit Basiskanal d02 lassen darauf schliessen dass Kanal d02 ziemlich gestört sein muss. Die Korrelationen von d02 mit den anderen Kanälen liefern alle falsche Delays. Nehmen wir nun d02 als Basiskanal dann werden $\hat{s}(d02, d01, spk)$, $\hat{s}(d02, d05, spk)$, $\hat{s}(d02, d06, spk)$ und die daraus resultierenden $S(d02, d01, utt)$, $S(d02, d05, utt)$, $S(d02, d06, utt)$ falsch berechnet. Wenn wir annehmen dass die anderen Kanäle nicht gestört sind und damit $S(d01, d05, utt)$, $S(d01, d06, utt)$ und $S(d05, d06, utt)$ richtige Delays sind, dann ist es leicht nachzuvollziehen, dass ein Versuch mit Basiskanal d02, der die Delays $S(d02, d01, utt)$, $S(d02, d05, utt)$ und $S(d02, d06, utt)$ (und somit drei falsche) verwendet schlechtere Ergebnisse liefert als z.B. ein Versuch mit Basiskanal d01, der die Delays $S(d01, d02, utt)$, $S(d01, d05, utt)$ und $S(d01, d06, utt)$ (also nur einen falschen) verwendet.

In diesen Experimenten werden die ermittelten Delays nicht weiter überprüft. Fehlerhafte Delays, also ermittelte Delays, die nicht den tatsächlichen Signallaufzeitunterschieden entsprechen werden nicht korrigiert.

4.3.2 Lokale Näherung mit Additionsmethode (E2.2)

Die Additionsmethode hilft uns, Ausreisser im lokalen Korrelationsspektrum teilweise zu entdecken und zu korrigieren.



Wir errechnen für jede Kanalkombination den Betrag der Abweichung von lokal nächstem Maximum und globalem Maximum.

$$a(dx, dy, utt) = \left|\hat{C}(dx, dy, spk) - S(dx, dy, utt)\right|$$

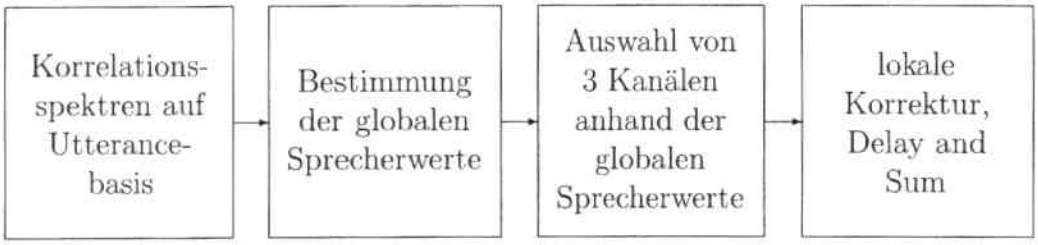
Nun kombinieren wir über die Additionsmethode die lokal nächsten Maxima zu einem Delay. Liegt dieser näher am globalen Maximum als das direkte lokale Maximum wählen wir selbigen als Delay dieser Kanalkombination für diese Utterance. Einen Fall behandeln wir gesondert : Liegt das lokale Maximum näher als 10 am globalen Maximum, so wird diese Delaysuche nicht durchgeführt, da wir davon ausgehen dass es sich bei einem so nahe am globalen Maximum liegenden lokalen Maximum nicht um einen Ausrutscher handelt (dies hat an der Senkung der WER einen Anteil von 0,3%).

Basiskanal	lokale Näherung ohne Additionsmethode	lokale Näherung mit Additionsmethode	Gewinn
d01	35,4	33,6	1,8
d02	40,3	38,7	1,6
d05	40,4	39,9	0,5
d06	37,1	36,0	1,1

Tabelle 4.5: lokale Korrektur des globalen Maximums mit Anwendung der Additionsmethode, manuelle Segmentierung, plain Modelle

Der Erfolg dieser Methode hängt natürlich stark von der Qualität unseres globalen Maximums ab. Sind die globalen Delays der über die Additionsmethode mit der betrachteten Kanalkombination verwandten Kanalkombinationen grob fehlerhaft, so ist durch deren Addition kein sinnvoller Delay zu erwarten.

4.4 Delay and Sum ohne Angabe eines Basiskanals (E3)



Das nun beschriebene Experiment benötigt keine Angabe eines Basiskanals. Das Experiment muss selbst entscheiden, welche Kanäle es für das Delay and Sum Beamforming auswählt und welche Delays es verwendet. Es kann durchaus ein Unterschied sein, ob man wenn man die Kanäle d01, d02 und d05 aufeinanderaddieren will d02 und d05 mit $s(d02, d05, utt)$ und dann die Summe mit $s(d01, d02, utt)$ auf d01 addiert oder ob man d05 mit $s(d01, d05, utt)$ und d02 mit $s(d01, d02, utt)$ auf d01 addiert. Entspricht $s(d01, d02, utt)$ nicht dem Laufzeitunterschied des Signals von d01 nach d02 so besteht im zweiten Fall immernoch die Möglichkeit, dass $s(d01, d05, utt)$ dem Laufzeitunterschied des Signals von d01 nach d05 entspricht und zumindest d01 und d05 korrekt geshiftet werden.

Für dieses Experiment wird eine Utterance die im ADC File zum Zeitpunkt *from* startet und bis zum Zeitpunkt *to* geht in $\lceil (to) \rceil - \lfloor (from) \rfloor$ Segmente seg der Länge 1s unterteilt. Für jedes dieser Segmente wird nun ein 6-Tupel errechnet :

$$t(seg) = (\text{argmax}_\tau(C(d01, d02, seg)), ..., \text{argmax}_\tau(C(d05, d06, seg)))$$

Alle Tupel einer Utterance werden nun darauf überprüft, ob Elemente zweier zeitlich benachbarter Vektoren nur geringfügig (dh. um weniger als 4) voneinander abweichen. Ziel dieser Analyse ist es, für die weitere Betrachtung nur die Delays zu wählen, die über mindestens 2 Sekunden konstant sind, da wir ja für den Laufzeitunterschied auch Konstanz über die ganze Utterance vorraussetzen. Resultat ist ein Merkmal $m(utt)$ welches genau die Delays für jede Kanalkombination enthält, die durch den vorhergehenden Schritt nicht herausgefiltert wurden. Darüberhinaus wird in dem Merkmal auch der zugehörige Sprecher gespeichert.

Sei eine Utterance 3.5 Sekunden lang, so können wir 4 6-Tupel t berechnen :

$$t_1 = (40, 480, 540, 440, 500, 60)\ t_2 = (91, 482, 540, 440, 503, 57)$$

$$t_3 = (43, 479, 380, 440, 506, 63)\ t_4 = (44, 580, 541, 440, 509, 59)$$

Das zugehörige Merkmal m hätte folgenden Inhalt :

{1 43} {1 44} {2 480} {2 479} {2 482} {3 540} {3 541} {4 440} {4 440} {4 440} {4 440} {5 500} {5 503} {5 506} {5 509}

Wie man sehen kann spielt die Reihenfolge in der die Delays aufgetaucht sind eine wichtige Rolle. So kommt es zustande, dass von der Kanalkombination 5 (d02d06) 4 Werte mit ins Merkmal aufgenommen werden, von der Kanalkombination 6 (d05d06) jedoch keines, obwohl sie eigentlich näher zusammen liegen.

Nun werden alle Merkmale eines Sprechers zusammengefasst. Dazu wird gezählt wie oft welcher Delay in den Merkmalen des Sprechers vorhanden ist.

{1 {43 1} {44 1}} {2 {480 1} {479 1} {482 1}} {3 {540 1} {541 1}} {4 {440 4}} {5 {500 1} {503 1} {506 1} {509 1}

Dann wird die Additionsmethode angewendet, um eine Delay-Kombination auszuwählen. Dazu wird für jede der 4 möglichen Additionen so vorgegangen :

Sei die zu testende Additionskombination d02d05 + d05d06 = d02d06 mit den zugehörigen Nummern 4,6 und 5. Man wählt nun alle Kombinationen von je einem Element aus den Mengen el_4, el_6 und el_5 mit den zugehörigen Häufigkeiten el_h_4, el_h_6 und el_h_5 und sortiert sie nach

$$p = \frac{(el\_h_4 + el\_h_6 + el\_h_5)^2}{|(el\_h_4 - el\_h_5)(el\_h_4 - el\_h_6)(el\_h_5 - el\_h_6)|}$$

Danach wird nacheinander für jedes Tripel der absteigend sortierten Liste geprüft, ob $|el_4 + el_6 - el_5| \leq 4$ ist. Sobald das erste Tripel gefunden ist bei dem das der Fall ist wird die Suche nach weiteren Tripeln abgebrochen.

Aus den aus dieser Suche hervorgehenden maximal 4 Tripeln wird nun das mit dem maximalen p ausgewählt.Das ausgewählte Triple enthält 3 $\hat{s}(dx, dy, spk)$, also zB. $\hat{s}(d02, d05, spk)$, $\hat{s}(d05, d06, spk)$ und $\hat{s}(d02, d06, spk)$, die nun noch wie auch schon im vorhergehenden Experiment für jede Utterance nach $S(dx, dy, utt)$ verschoben werden.

Die Verfeinerung mit der Additionsmethode lässt sich hier nicht anwenden, da nur 3 der 6 möglichen Kanalkombinationen global bestimmt werden.

automatische Segmentierung, plain Modelle	41,5
automatische Segmentierung, adaptierte Modelle	32,2
manuelle Segmentierung, plain Modelle	35,5
manuelle Segmentierung, adaptierte Modelle	26,7

Tabelle 4.6: Wortfehlerrate in Abhängigkeit von Segmentierung und Art der akustischen Modelle, manuelle Segmentierung, plain Modelle

Im Vergleich zum vorhergehenden Experiment werden hier also nur 3 Kanäle via Delay and Sum aufeinanderaddiert. Falls eine Möglichkeit bestände, dieses Experiment auf 4 Kanäle zu erweitern könnte die Erkennerleistung noch gesteigert werden - diese Vermutung geht daraus hervor dass das bereits beschriebene Experiment E2.1 (lokales Maximum in Nähe des globalen Maximums) mit Basiskanal d01 nur mit 3 Kanälen (der schlechteste, d02 mit 57% WER auf den manuell segmentierten Daten mit plain Modellen wurde weggelassen) bei der WER von 35,4% auf 36,7% steigt.

```
/* Experiment E3 in Pseudocode                                          */
foreach Kanalkombination dx,dy do
    foreach Utterance utt do
        spk ← spk(utt)
        foreach Segment seg ∈ utt do
            berechne s(dx, dy, seg)
            if s und benachbartes s (nicht über utt-Grenzen) liegen nur
            höchstens 3 auseinander then
                füge s zur listofcandidates(dx,dy,spk) hinzu
            end
        end
    end
end
foreach element von listofcandidates do
    versehe Elemente von listofcandidates(element) mit Häufigkeit h
    und lösche doppelte Einträge
end
foreach spk do
    foreach Tripel (a,b,c)
    ((d01,d02,d05),(d01,d05,d06),(d01,d02,d06),(d02,d05,d06)) do
        wähle je ein Element aus listofcandidates(a,b,spk) loc_a,
        listofcandidates(b,c,spk) loc_b und listofcandidates(c,a,spk) loc_c,
        so dass gilt:
        Wähle unter allen Tripeln (loc_a, loc_b, loc_c), die
        |loc_a + loc_b + loc_c| ≤ 4 erfüllen das aus, das
        p = (h(loc_a)+h(loc_b)+h(loc_c))^2 / |(h(loc_a)-h(loc_b))(h(loc_a)-h(loc_c))(h(loc_b)-h(loc_c))| maximiert.
        potDelay(spk,a,b,c) ← (loc_a,loc_b,loc_c,p)
    end
    Wähle für Delay and Sum von Utterances von Sprecher spk die 3
    Kanäle a,b und c mit dem grössten p. Addiere b mit Delay loc_b
    und c mit Delay loc_c auf a.
end
```

Kapitel 5

Abschliessende Bemerkungen

5.1 Zusammenfassung

Das Ziel, nur noch einen Kanal dekodieren zu müssen und dabei bessere Ergebnisse zu erreichen als beim Dekodieren eines jeden einzelnen Kanals wurde erreicht.

Segmentierung Modelle	automatisch plain	automatisch adaptiert	manuell plain	manuell adaptierte
E2.1, Basiskanal d01	-5,3	-2,0	-13,5	-5,7
E2.2, Basiskanal d01			-15,4	
E2.1, Basiskanal d02	-2,4	-2,3	-16,7	-5,9
E2.2, Basiskanal d02			-18,3	
E2.1, Basiskanal d05	-2,2	-0,1	-8,7	-2,4
E2.2, Basiskanal d05			-9,2	
E2.1, Basiskanal d06	-5,7	-5,9	-3,4	-2,7
E2.2, Basiskanal d06			-4,5	
E3	-6,7	-3,6	-14,1	-6,8

Tabelle 5.1: Absolute WER-Änderung in Abhängigkeit des Experiments

Durch die Vorverarbeitung konnten wir die Wortfehlerrate bei automatischer Segmentierung um 4,5% / 2,8% (plain / adaptiert) und bei manueller Segmentierung um 11,3% / 4,7% (plain / adaptiert) senken. Die Ergebnisse zeigen, dass die Ansätze, mit denen wir versucht haben, globale Maxima zu bestimmen und falsche Kandidaten zu eliminieren durchaus funktionieren. Es ist nicht weiter erstaunlich, dass der Gewinn bei der manuellen Segmentierung höher ist - wir haben bei allen Experimenten mit globalen Maxima gearbeitet, deren Position durch die fehlerbehaftete Segmen-

tierung und der daraus resultierenden falschen Zuordnung von Segmenten zu Sprechern verfälscht wird. Der hohe Zugewinn bei den Ergebnissen der Versuche mit plain Modellen im Vergleich zu adaptierten Modellen liegt zum Teil darin, dass bei der Adaption der Erkenner auf einzelne Raum-Mikrofon-Aufnahmen adaptiert wird. Interessant wäre sicherlich auszuprobieren, was passieren würde wenn man die Adaption auf den bereits vorverarbeiteten Daten durchführen würde. Eine zweite Ursache liegt darin begründet, dass die Baselines mit adaptierten Modellen von der WER her viel niedriger liegen als die der Baselines mit plain Modellen.

5.2 Ausblick

Die Erweiterung des Experiments E3 auf 4 Kanäle verspricht eine weitere Senkung der Wortfehlerrate. Die Erweiterung wäre relativ einfach umzusetzen wenn es Methoden geben würde, mit denen man den besten Basiskanal, der bei den meisten Experimenten benötigt wird, bestimmen könnte.

Möglicherweise können die Ergebnisse der Erkenner mit adaptierten Modellen darüberhinaus dadurch verbessert werden, dass man erst auf den vorverarbeiteten Daten adaptiert.

Anhang A

Bezeichnungsübersicht

dx, dy	Kanal (d01,d02,d05 oder d06)
$C(dx, dy, utt)$	Korrelationsspektrum der Kanäle dx und dy der Utterance utt
$\tilde{C}$	geglättetes Korrelationsspektrum
SPK	Menge aller Sprecher
spk	bestimmter Sprecher
UTT	Menge aller Utterances eines Recordings
utt	bestimmte Utterance
seg	Segment, Teil einer Utterance
$spk(utt)$	Sprecher der Utterance utt
$s(dx, dy, seg)$	die Stelle des Maximums von $C(dx, dy, seg)$, potentieller Delay
$t(dx, dy, spk)$	Laufzeitunterschied der Signale von Sprecher spk zwischen dx und dy
$\hat{C}(dx, dy, spk)$	globales Korrelationsspektrum von Sprecher spk
$\hat{s}(dx, dy, spk)$	globaler Delay
$S(dx, dy, utt)$	finaler Delay
$a(dx, dy, utt)$	Differenz von $\hat{C}$ und C
$t(seg)$	Merkmalsvektor von E3
$\tilde{F}_{lokal}$	geglättetes, lokales Korrelationsspektrum
$F_{global\_spk\_utt}$	sprecherabhängiges, globales Korrelationsspektrum auf Utterance-Basis
$F_{global\_spk\_seg}$	sprecherabhängiges, globales Korrelationsspektrum auf Segment-Basis
$F_{global\_all\_utt}$	sprecherunabhängiges, globales Korrelationsspektrum

Literaturverzeichnis

[1] "Spring 2004 (RT-04S) Rich Transcription Meeting Recognition Evaluation Plan", *http://www.nist.gov/speech/tests/rt/rt2004/spring/documents/rt04s-meeting-eval-plan-v1.pdf*

[2] "RT-04S Development Test Data Documentation",*http://www.nist.gov/speech/tests/rt/rt2004/spring/devset/index.html*

[3] "RT-04S Evaluation Data Documentation",*http://www.nist.gov/speech/tests/rt/rt2004/spring/eval/docs.html*

[4] A.Stolcke, C.Wooters, N.Mirghafori, T.Pirinen, I.Bulyko, D.Gelbart, M.Graciarena, S.Otterson, B.Peskin, M.Ostendorf "Progress in Meeting Recognition: The ICSI-SRI-UW Spring 2004 Evaluation System", *Proc. ICASSP-2004 Meeting Recognition Workshop*;Montreal, May 2004

[5] F.Metze, C.Fügen, Y.Pan, T.Schultz, H.Yu "The ISL RT-04S Meeting Transcription System", *Proc. ICASSP-2004 Meeting Recognition Workshop*;Montreal, May 2004

[6] F.Metze, C.Fügen, Y.Pan, A.Waibel "Automatically transcribing meeting using distant microphones", *Proc. ICASSP-2004 Meeting Recognition Workshop*;Montreal, May 2004

[7] A.Janin, D.Baron, J.Edwards, D.Ellis, D.Gelbart, N.Morgan, B.Peskin, T.Pfau, E.Shriberg, A.Stolcke und C.Wooters, "The ICSI meeting corpus" in *Proceedings IEEE Int'l Conference on Acoustics, Speech & Signal Processing (ICASSP-2003)*;Hong Kong, April 2003

[8] Qin Jin, Kornel Laskowski, Tanja Schultz and Alex Waibel "Speaker Segmentation and Clustering in Meetings" in *Proc. ICASSP-2004 Meeting Recognition Workshop*; Montreal, May 2004

[9] M. Siegler, U. Jain, B. Raj and R. Stern "Automatig Segmentation, Classification and Clustering of Broadcast News Audio" in *DARPA Speech Recognition Workshop*, Chantilly, Virginia, 1997

[10] Y.Rui und D.Florencio "Time delay estimation in the presence of correlated noise and reverberation" *Proc. ICASSP 2004*, Montreal, May 2004

Abbildungsverzeichnis

Tabellenverzeichnis