EIN SPRACHDIALOGSYSTEM MIT BEGRENZTEN HARDWARERESSOURCEN

Frank Duckhorn, Guntram Strecha, Matthias Wolff und Rüdiger Hoffmann
Technische Universität Dresden

Kurzfassung: Wir entwickeln ein Sprachdialogsystem, welches auch unter begrenzten Hardwareressourcen lauffähig sein soll. Deswegen verwenden wir für die Erkennung sowie für die Synthese die selben, sprecherunabhängigen Hidden-Markov-Modelle (HMM). Der Spracherkenner ist phonembasiert und kann beliebige reguläre Grammatiken verarbeiten. Die Synthese beruht auf der Verkettung von Syntheseeinheiten (Morpheme und Wörter), welche jeweils durch eine Zustandssequenz innerhalb des HMMs sowie dem Grundfrequenz- und Energieverlauf definiert werden. Für die Auswahl der Einheiten benutzen wir eine endliche Grammatik. Um mit einer bestimmten Stimme zu synthetisieren, werden die Merkmalvektoren der sprecherunabhängigen HMMs je nach gewünschtem Sprecher in Line-Cepstral-Frequency-Merkmale (LCQ) transformiert und geglättet. Das gesamte Sprachdialogsystem ist auf einem digitalen Signalprozessor (DSP) lauffähig. Ein Field Programmable Gate Array (FPGA) übernimmt dabei die rechenintensiven Algorithmenteile. Unser Ziel ist die Hardwaregröße und den Strombedarf soweit zu reduzieren, dass das Sprachdialogsystem in Form eines USB-Sticks an verschiedenen Geräten eingesetzt werden kann.

1 Einleitung

Ziel unserer Arbeit ist es, ein Sprachdialogsystem zu entwickeln, welches an verschiedene Geräte angekoppelt werden kann. Als Schnittstelle zum Gerät haben wir uns für USB entschieden, da dies weit verbreitet ist. Zur Sprachein- und -ausgabe ist das Dialogsystem über Bluetooth mit einem Headset verbunden (siehe Abbildung 1).

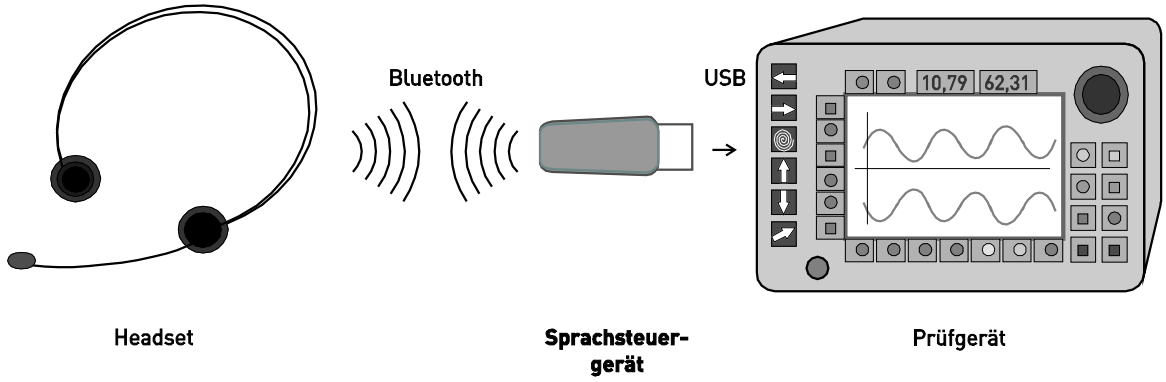


Abbildung 1 - Einsatz des Sprachdialogsystems

Da die über USB zur Verfügung gestellte elektrische Leistung begrenzt ist, muss das Dialogsystem, welches aus Spracherkennung und -synthese besteht, mit diesen begrenzten Hardwareressourcen auskommen. Um das zu gewährleisten, verfolgen wir zwei neue Ansätze:

1. Spracherkenner und -synthetisator verwenden die gleichen Hidden-Markov-Modelle. Dadurch kann der Speicherplatz für ein zweites Modell gespart werden.

2. Rechenintensive und zeitkritische Algorithmen werden nicht auf dem Signalprozessor ausgeführt, sondern auf ein FPGA ausgelagert.

2 Hardwareaufbau

Der Hardwareaufbau des Sprachdialogsystem teilt sich in zwei Blöcke (siehe Abbildung 2). Zum einen benutzen wir als Hauptprozessor einen digitalen Signalprozessor (TMS320C6727+). Dieser verfügt über Gleitkommarecheneinheiten sowie über 16 MB externen Speicher. Zum Import von Konfigurationsparametern und Modelldaten ist außerdem ein Flashspeicher angebunden. Der zweite Teil des Dialogsystems ist ein FPGA (Xilinx LCA Virtex4), welcher die Schnittstellen zum Headset (Bluetooth) und zum angekoppelten Gerät (USB) steuert. Dieser FPGA soll zudem rechenintensive Algorithmen übernehmen, um den Signalprozesser zu entlasten und durch die zusätzliche Rechenkapazität eine höhere Qualität von Erkennung und Synthese zu erreichen.

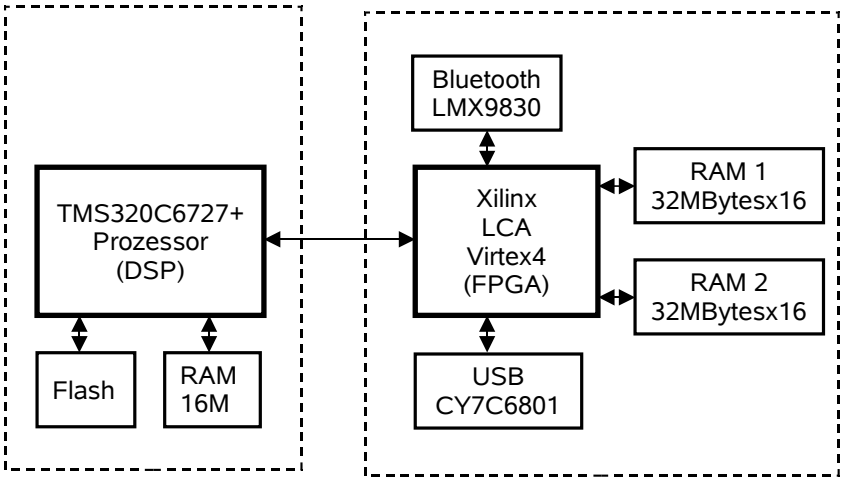


Abbildung 2 - Hardwareaufbau

3 Spracherkenner

Der Spracherkenner benutzt als Merkmale die logarithmierten Werte einer MEL-Filterbank, welche aus dem in Rahmen geteilten und fourier-transformierten Signal gewonnen werden. Diese primären Merkmale werden in einem zweiten Schritt der Merkmalextraktion um die dynamischen Merkmale erweitert und nach einer Hauptkomponentenanalyse zu einem 21-dimensionalen sekundären Merkmalvektor reduziert.
Die eigentliche Klassifikation basiert auf 43 Hidden-Markov-Modellen (eines pro Phonem). Jedes Modell hat drei Zustände, welchen eine Mischung von Normalverteilungen zugeordnet ist. Durch die Benutzung voller Kovarianzmatrizen war es möglich, sich auf nur acht Verteilungen pro Mixtur zu beschränken. Außerdem wird ein festes Lexikon und eine feste Grammatik verwendet. Alle diese Modelle werden in Form eines endlichen Zustandsautomaten dargestellt (siehe [1]). Diese Zustandsautomaten haben Ein- wie auch Ausgabesymbole und können somit als Übersetzer zwischen den einzelnen Ebenen verwendet werden (das HMM übersetzt

zwischen Normalverteilungsmixtur- und Phonemindex, das Lexikon zwischen Phonem- und Wortindex und die Grammatik zwischen Wort- und Kommandoindex). Dies ermöglicht die Komposition der einzelnen Automaten zu einem einheitlichen Erkennungsnetzwerk, welches zwischen Normalverteilungsmixtur- und Kommandoindex übersetzt). Außerdem kann das entstehende Erkennungsnetzwerk durch bekannte Algorithmen für endliche Zustandsautomaten (zum Beispiel Minimierung) deutlich in seiner Größe reduziert werden. Schließlich kann die Klassifikation in folgender Form ablaufen:

1. Berechnung der Verteilungsdichten für jede Mixtur und den entsprechenden Zeitschritt,
2. Synchrone Suche im Erkennungsnetzwerk zur Bestimmung der Pfade mit der höchsten und zweithöchsten Wahrscheinlichkeit,
3. Bestimmung der Erkennungsicherheit aus der Wahrscheinlichkeitsdifferenz der zwei Pfade.

Der gesamte Erkennungsablauf wie auch der Trainingsschritte zur Modellgenerierung können in der Abbildung 3 nachvollzogen werden.

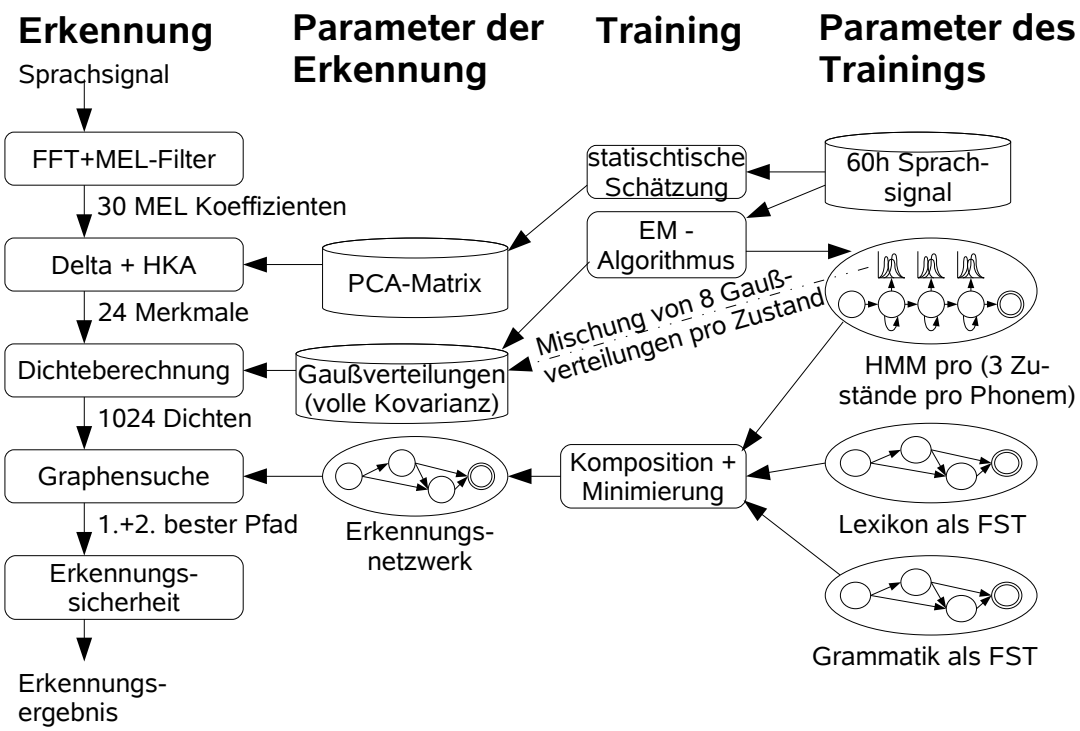


Abbildung 3 - Aufbau von Spracherkenner und Modelltraining

3.1 Implementierung

Unsere Vorbetrachtungen haben gezeigt, dass die Berechnung der Verteilungsdichten den größten Rechenaufwand im Ablauf darstellt. Aus diesem Grund und weil diese Berechnung sich gut parallelisieren lässt, haben wir uns entschieden sie auf den FPGA zu portieren. Im Weiteren wird die gesamte Merkmalextraktion ebenfalls auf dem FPGA durchgeführt. Somit berechnet der FPGA aus dem eingegebenen Sprachsignal zu jedem Rahmen alle Verteilungsdichten und

liefert diese dann in Echtzeit zum Signalprozessor. Dort wird die synchrone Suche im Erkennungsnetzwerk, wie auch die Bestimmung der Erkennungssicherheit ausgeführt. Für die Dekodierung benutzen wir eine modifizierte A*-Suche, die aufgrund ihrer Zulässigkeit garantiert den besten Pfad liefert, aber deutlich weniger Knoten expandieren muss wie eine vergleichbare dynamische Programmierung.

3.2 Sprache-Pause-Detektion

Die Sprache-Pause-Detektion wird ebenfalls mittel statistischer Klassifikation umgesetzt. Allerdings beschränken wir uns hier auf ein Gaussian-Mixture-Modell. Es werden einzelnen Modelle für Pause, Sprache und Nebengeräusche trainiert. Ein Zustandsautomat passt anschließend die Entscheidung des GMMs so an, dass gewisse Parameter, wie minimale/maximale Sprach- und Pausenlänge sowie Vor- und Nachlauf, eingehalten werden. Der Vorteil in der statistischen Klassifikation liegt hier in der Möglichkeit, die Sprache-Pause-Detektion an verschiedene Umgebungsgeräusche oder Mikrofonstörungen zu adaptieren. Speziell das Pause-Modell kann gezielt auf die Bedingungen trainiert werden. Außerdem bietet dieses Verfahren die Möglichkeit, Algorithmen, die von der Spracherkennung benutzt werden, wiederzuverwenden. So ist die gesamte primäre Merkmalextraktion gleich. Danach werden für die Sprache-Pause-Detektion eine Hauptkomponentenanalyse durchgeführt sowie die Verteilungsdichten berechnet.

4 Synthese

Bei der HMM-Synthese unter Verwendung der Modelle des HMM-Erkenners ergeben sich drei prinzipielle Probleme, welche gelöst werden müssen.

1. Es ist eine geeignete Gaußfolge zu schätzen, so dass aus der Folge der synthetisierbaren Sprachparameter, welche der Gaußfolge zugeordnet ist, ein qualitativ hochwertiges Sprachsignal generiert werden kann.

2. Es ist eine Glättung der Sprachparameter notwendig, da die Gaußfolge aus sich wiederholenden Zuständen bestehen kann. Damit wiederholen sich die an den Zustandsübergängen emittierten Sprachparameter, was zu einer unnatürlichen Parameterfolge führt.

Diese zwei Probleme sind prinzipiell bei einer HMM-Synthese zu lösen.

3. Die Verwendung sprecherunabhängiger HM-Modelle des Erkenners, welche mit Sprachsignalen einer großen Anzahl an Sprechern trainiert wurden, erfordern eine Sprecheradaption. Algorithmen zur Sprecheradaption, wie sie aus der Spracherkennung bekannt sind, werden üblicherweise verwendet, um aus den sprecherunabhängigen HMM's Modelle eines Zielsprechers zu bilden [4]. Derartige Methoden erfordern zusätzliche Adaptionsparameter, da die HM-Modelle für die Erkennung bewahrt bleiben müssen. Die Speicherung der Adaptionsparameter und die Algorithmen zur Adaption während der Synthese erfordern zusätzlichen Speicher- und Rechenaufwand, welcher den Einsatz auf des kombinierten Erkennungs-/Synthesesystems auf der Zielplattform verhindern.

Unter Ausnutzung der Vorgaben für das Dialogsystem ist eine Lösung der Probleme mit sehr geringem zusätzlichen Speicher- und Rechenaufwand möglich [3]. Da die Anzahl der zu synthetisierenden Äußerungen begrenzt ist, kann man einen Satz von kleinen Einheiten definieren, aus denen das Synthesesignal generiert wird. Die Einheiten bestehen aus den vorberechneten Gaußfolgen, deren Indizes in einem Inventar einheitenweise abgelegt werden. Die Synthese der

Tabelle 1 - Parameteranzahl des Inventars und der Konvertierungsmatrix der drei getesteten Stimmen A, C und M.

	Stimme A	Stimme C	Stimme M
Einheiteninventar	10575	11103	9279
Konvertierungsmatrix	1860	1860	1860

Gaußfolge der verketteten Einheiten umgeht die wiederholte Berechnung während des Syntheselaufs. Außerdem können bei der Erstellung des Inventars die Gaußfolgen optimal, hinsichtlich der Minimierung des globalen Fehlers zwischen den Sprachparametern eines gegebenen natürlichen Sprachsignals und den emittierten Parametern der optimalen Zustandsfolge des HMM, bestimmt werden. Diese Vorgehensweise löst Problem 1. Neben den Indizes der Gaußfolgen wird der Verlauf der Grundfrequenz für jede Einheit im Inventar gespeichert. Anhand dieser Grundfrequenzverläufe wird das Anregungssignal des Synthesefilters berechnet.
Für das Problem der sprecherabhängigen Synthese aus den sprecherunabhängigen Modellen wurde eine einfache Lösung gefunden, welche sich besonders für die Synthese mit begrenzter Anzahl an Einheiten eignet. Die lineare Transformation der Sprachparameter der HM-Modelle mit Hilfe einer sprecherabhängigen Transformationsmatrix erzeugt die Sprachparameter des Zielsprechers. Die Konvertierungsmatrix wird durch Minimierung der Summe des quadratischen Fehlers zwischen den Sprachparametern aller Einheiten des Zielsprechers und den der Gaußfolgen zugeordneten Sprachparametern der entsprechenden Einheiten bestimmt. Die Inventargrößen sowie die Größen der Transformationsmatrizen sind in Tabelle 1 für die drei für das Projekt getesteten Sprecher aufgelistet. Die Sprachparameter des Erkenners des vorgestellten Dialogsystems sind das Melfilter-Spektrum. Als Sprachparameter der Synthese wurden die Line Cepstral Quefrencies (LCQ, [2]) gewählt, da diese sehr gute Glättungseigenschaften besitzen (Problem 2).
Zur Evaluation des Synthesesystems wurden ein Degradation Mean Opinion Score (DMOS) und ein MOS-Hörtest durchgeführt. Beim DMOS-Hörtest wurde die Verschlechterung der Synthesesignale (erzeugt aus den unkonvertierten und den konvertierten Sprachparametern) gegenüber natürlicher Sprache gemessen. Beim MOS-Hörtest wurden Synthesesignale evaluiert, welche aus allen Kombinationen von Sprecherinventaren und Konvertierungsmatrizen erzeugt

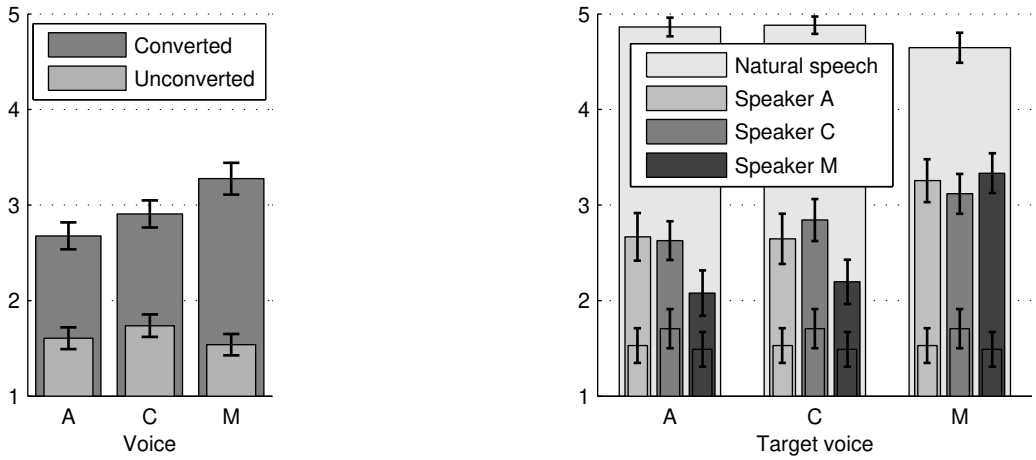
Tabelle 2 - Resultate des **(a)** DMOS- und **(b)** MOS-Hörtests. Beim DMOS-Hörtest wurden die Sprachsignale mit den Einheiten von Sprecher A, C und M einmal unkonvertiert und einmal konvertiert zu den Stimmen A, C bzw. M synthetisiert und evaluiert. Beim MOS-Hörtest wurden neben der natürlichen Sprache und den aus den unkonvertierten Sprachparametern erzeugten Synthesesignalen alle Kombinationen aus Basissprecher und Zielstimme evaluiert.

(a) DMOS-Hörtest.

Sprecher/ Stimme	unkonvertiert	konvertiert
A	1.61 ± 0.11	2.68 ± 0.14
C	$1,74 \pm 0,12$	$2,91 \pm 0,14$
M	$1,54 \pm 0,11$	$3,28 \pm 0,17$

(b) MOS-Hörtest.

		Einheiten A	Einheiten C	Einheiten M
nat. Sprache		$4,86 \pm 0,10$	$4,88 \pm 0,09$	$4,65 \pm 0,16$
unkonvertiert		$1,53 \pm 0,18$	$1,71 \pm 0,21$	$1,49 \pm 0,18$
konvertiert zu Ziel- stimme	A	$2,67 \pm 0,25$	$2,63 \pm 0,20$	$2,08 \pm 0,24$
	C	$2,65 \pm 0,26$	$2,84 \pm 0,22$	$2,20 \pm 0,23$
	M	$3,25 \pm 0,22$	$3,12 \pm 0,21$	$3,33 \pm 0,21$



(a) DMOS-Hörtest.

(b) MOS-Hörtest.

Abbildung 4 - Grafische Darstellung der Resultate des DMOS- (siehe Tabelle 2(a)) und MOS-Hörtests (siehe Tabelle 2(b)).

wurden. Die Ergebnisse der Hörtests mit den Konfidenzintervallen sind in den Tabellen 2(a) und 2(b) sowie in den Abbildungen 4(a) und ¸4(b) dargestellt. Die Ergebnisse der Hörtests zeigen eine signifikante Verbesserung der Synthesequalität durch die Konvertierung der Parameter. Darüber hinaus zeigen die Ergebnisse des MOS-Hörtest, dass sich die Synthesequalität auch bei der Kombination von Sprecherinventaren mit Konvertierungsmatrizen unterschiedlicher Sprecher, im Vergleich zu der Synthese aus den unkonvertierten (Melfilter-Spektrum) Parametern signifikant, erhöht.

Literatur

[1] PEREIRA, F. C. N. und M. D. RILEY: *Speech Recognition by Composition of Weighted Finite Automata*. In: *Finite-State Language Processing*, S. 431–453. MIT Press, 1996.

[2] STRECHA, G., M. EICHNER und R. HOFFMANN: *Line Cepstral Quefrencies and Their Use for Acoustic Inventory Coding*. In: *Proc. of Interspeech*, S. 2873–2876, Antwerpen, Aug. 2007.

[3] STRECHA, G., M. WOLFF, F. DUCKHORN, S. WITTENBERG und C. TSCHÖPE: *The HMM Synthesis Algorithm of an Embedded Unified Speech Recognizer and Synthesizer*. In: *Proc. of Interspeech*, Sep. 2009. in Druck.

[4] YAMAGISHI, J., K. OGATA, Y. NAKANO, J. ISOGAI und T. KOBAYASHI: *HSMM-Based Model Adaptation Algorithms for Average-Voice-Based Speech Synthesis*. In: *Proc. of ICASSP*, Bd. 1, S. 77–80, Mai 2006.