Ein neuer Systemansatz für die Integration multimodalen Inputs durch Late Semantic Fusion

A New Approach for Integrating Multimodal Input via Late Semantic Fusion

Dipl.-Math. Gregor McGlaun, Dipl.-Inform. Frank Althoff, Prof. Dr. rer. nat. Manfred Lang

Lehrstuhl für Mensch-Maschine-Kommunikation
Technische Universität München, 80290 München
Tel.: +49/89/289-28541, Fax: +49/89/289-28535
Email: {mcglaun | althoff | lang}@ei.tum.de

Kurzdarstellung
Thema dieses Beitrags ist ein innovatives Konzept für eine multimodale Systemarchitektur, in welcher die von den jeweiligen Erkennermodulen gelieferten Informationsanteile auf Basis einer Late Semantic Fusion in Echtzeit verarbeitet werden. Das Integrationssystem wurde mit einem rein regelbasierten sowie einem probabilistischen Ansatz umgesetzt und hinsichtlich Effizienz und Fehlerrobustheit evaluiert.

Abstract
In this contribution, we present an innovative concept of a multimodal system architecture that is based on the principle of Late Semantic Fusion. The inputs of the individual recognizer modules are processed in real-time. We have implemented the integration-system by means of a purely rule-based as well as a probabilistic approach, and evaluated it with regard to efficiency and error-robustness.

1. Einführung

Die heutigen technischen Systeme müssen ein immer breiteres Spektrum an individuellen Benutzern, Funktionalität und Umgebungssituationen abdecken. Dies erfordert hohe Ansprüche an das Design der Mensch-Maschine-Schnittstellen, welche einerseits intuitiv, flexibel und fehlerrobust sein sollen, gleichzeitig aber auch hohen technischen und applikationsspezifischen Ansprüchen gerecht werden müssen. Als vielversprechender Lösungsansatz erweisen sich multimodale Systeme, da sie den natürlichen Kommunikationsgewohnheiten des Menschen in besonderer Weise durch vielseitige und flexible Eingabeparadigmen gerecht werden. Ein spezieller Ansatz der multimodalen Integrationstheorie basiert auf der *paramodalen* Repräsentation von Informationen. Hierbei wird angenommen, dass die jeweiligen Informationseinheiten eine Anzahl von in sich abgeschlossenen Eigenschaften besitzen, die allesamt voneinander unabhängig betrachtet werden können. Man greift auf bereits ausgereifte unimodale Erkenner zurück und verfolgt das Konzept einer Kombination oft auch gleichartiger bzw. konkurrierender Systeme. Ziel ist somit nicht die weitere Verbesserung der jeweiligen Einzelerkenner, sondern vielmehr eine zentrale Verwaltung bzw. Evaluierung des kontinuierlich eintreffenden, bewusst auch mit redundanten Anteilen versehenen Informationsstroms außerhalb der Erkennerebene (Late Semantic Fusion [1]).

Quelle: VDI-Bericht 1678 "USEWARE 2002, VDI-Verlag, Düsseldorf

2. Methodik

2.1 Multimodale Integration

Die Grundlage für die Bewältigung des Fusionsproblems liegt im Entwurf einer entsprechend strukturierten Systemarchitektur, in welcher der multimodale Integrator die zentrale Instanz darstellt. Der Integrator hat prinzipiell die Aufgabe, den kontinuierlichen Ausgabestrom der einzelnen monomodalen Erkenner sinnvoll und effizient in einen ganzheitlichen Zusammenhang zu bringen und daraus für die einzelnen Applikationsmodule verwertbare Kommandos zu generieren. Von wesentlichem Interesse ist in diesem Zusammenhang die Synchronisation der Erkennerergebnisse der einzelnen Modalitäten vor allem unter der Prämisse, dass die jeweiligen Monomodalerkenner zumeist mit unterschiedlichen Laufzeiten ihre Ergebnisse an den Integrator liefern (*zeitliche Modalitätenkorrespondenz*). Des Weiteren muss unter Einbeziehung von applikationsspezifischen Kontextinformationen bei der Integration geprüft werden, ob die gewählte Interpretation der Befehle überhaupt plausibel ist, i.e. der Benutzerintention entspricht (*semantische Modalitätenkorrespondenz).*

Die Fehlerrobustheit eines technischen Systems stellt einen wesentlichen, von der Domäne unabhängigen Einflussfaktor für dessen Akzeptanz beim Benutzer dar. Dabei ist sowohl das *passive* (a priori Fehlervermeidung) wie auch das *aktive Fehlermanagement* (a posteriori Fehlerhandling) von Bedeutung. Ungewünschte Systemreaktionen durch Fehlbedienung sowie systemseitige Fehler, z. B. durch fehlerhafte Erkennung, sollen weitestgehend vermieden bzw. möglichst effizient abgefangen und nach Möglichkeit für den Benutzer transparent korrigiert werden.

2.2 Systemarchitektur

Das System ist in Form einer Client-Server-Struktur organisiert. Hierbei stellt der Integrator den zentralen Server dar, an dem sich die Elemente der Eingabe- und Ausgabeschicht als Clients anmelden. Die Kommunikation zwischen Eingabe-, Evaluierungs- und Ausgabeschicht erfolgt durch einen bidirektionalen Austausch von String-Messages, welche über TCP/IP-Verbindungen (Socket Backports) gestreamt werden (vgl. Abb. 1).

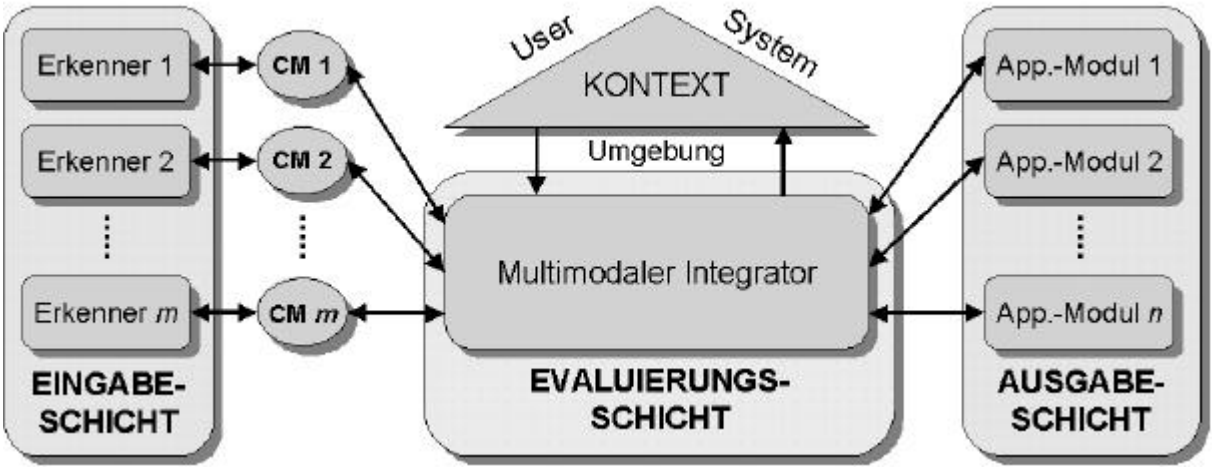

Abb. 1: Multimodale Systemarchitektur

Ein Meta-Device, der sogenannte Command Mapper (CM, vgl. Abb.1) konvertiert, basierend auf dem Formalismus einer kontextfreien Grammatik [2], mittels einer Look-Up Table sämtliche Nachrichten der Monomodalerkenner vor der Evaluierung durch den Integrator in ein erkennerunabhängiges Format. Diese starke Modularisierung ermöglicht eine rasche und unkomplizierte Ersetzung bzw. zusätzliche Einbindung einzelner Erkennermodule. Der aus verschiedenen miteinander kommunizierenden Komponenten (vgl. Abb. 2) bestehende Integrator interpretiert den von den einzelnen Erkennern eintreffenden multimodalen Informationsstream.

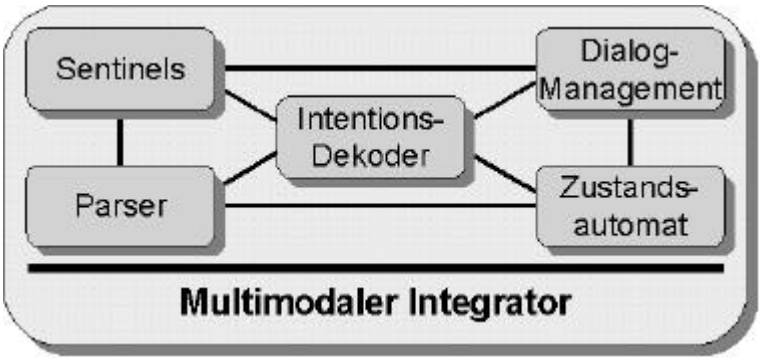

Abb. 2: Komponenten des Integrators bei Late Semantic Fusion

Ein String-Parser überprüft die eintreffenden Nachrichten auf Korrektheit der Syntax und Vollständigkeit. Die Datenbasis für den multimodalen Integrationsprozess wird von einem Zustandsautomaten zur Verfügung gestellt und verwaltet, wobei auf sekundäres Wissen aus der Applikation, den Erkennermodulen und dem Integrationszustand zurückgegriffen werden kann. Der Intentionsdekoder bildet die zentrale Komponente innerhalb des multimodalen Integrators. Unter Verwendung von Kontextinformation aus dem Dreieck Benutzer (z.B. emotionale Muster), System (interne Fehlermeldungen) und Umgebung (Lichtverhältnisse, Geräusche) werden hier die Nachrichten auf semantischer Ebene mittels eines semantischen Unifikationsprozesses evaluiert. Als Ergebnis werden Systemkommandos der Zielapplikation generiert, in Befehle der kontextfreien Grammatik umgesetzt und an die Applikation geschickt.

2.3 Integrationsprozess

Der Integrationsprozess wird von hochspezialisierten Einzelagenten, den sogenannten *Message Sentinels*, übernommen, die den multimodalen Datenstrom überwachen. Der Semantic Evaluation Sentinel (SES) hat die Aufgabe, die Informationsströme zu kategorisieren und nach einem speziellen, im Folgenden dargestellten Auswertungsschema zu gruppieren. Die Zuordnung geschieht auf Basis von globalen Objektstrukturen (typed feature structures). Via einer applikationsspezifischen Wissensbasis, die dem SES in Form von Konfigurationsfiles als Hash-Tables zur Verfügung gestellt wird, ist der Sentinel in der Lage, Kommandopräfixe, Parameter und Vollbefehle zu unterscheiden. Der SES besitzt ferner Wissen über die Art und die Anzahl von zugehörigen Parametern eines Befehlspräfixes. Die Verwaltung der jeweiligen Kommandos findet in mehreren FIFO-Queues statt. Um die Kommandos korrekt zu fusi-

onieren, müssen mehrere Aspekte berücksichtigt werden. Eine wesentliche Rolle spielt die dynamische, i.e. kontextabhängige Festlegung eines Integrationszeitfensters. Dieses wird u.a. durch die charakteristische Latenzzeit eines jeden Erkenners zwischen Benutzereingabe und dem Eintreffen des Befehls am Integrator bestimmt. So werden etwa haptische Befehle zeitlich sehr viel schneller (im Schnitt ca. 500ms) umgesetzt als Kommandos aus einer Sprach- bzw. Gestikeingabe des Benutzers, da sich eine reine Ereignisweiterleitung (z.B. ein Button-Event) i.a. weniger komplex darstellt als ein Mustererkennungsprozess. Eine schwer zu erfassende Größe, nach welcher sich die Dauer der Integrationsperiode richtet, ist die Abfolgegeschwindigkeit, in der der Benutzer einzelne Befehle eingibt. Sie wird von individuellen, situativen und domänenabhängigen Faktoren bestimmt, die sich auf die Nutzer-Workload auswirken. Das System verfügt hierzu über ein entsprechendes Benutzermodell, welches mit jeder neuen Eingabe geupdatet wird.
Stellt der SES nun in einem Zeitfenster des Integrationsprozesses mehrere Nachrichten unterschiedlicher Erkenner fest, die jedoch Kommandos mit identischer semantischer Bedeutung enthalten, so werden diese als *redundante* Eingabe interpretiert. Jede der beiden Nachrichten bringt gegenüber der anderen keine zusätzliche Information, sondern stützt lediglich die beabsichtigte Eingabe. Dies hat zur Folge, dass aus den beiden Kommandos ein einziges Applikationskommando erzeugt wird, um eine doppelte Ausführung, welche nicht der Benutzerintention entspräche, zu vermeiden.
Bezüglich der Verschmelzung der *komplementären*, i.e. sich ergänzenden Informationseinheiten (Kommandopräfix und Parameter) werden zunächst innerhalb eines Integrationsfensters alle zu einem Befehlspräfix passenden Parameter identifiziert. Im Falle mehrerer Alternativen wird unter Verwendung von Informationen aus dem Benutzermodell, der Befehlshistorie, Statusvariablen der Applikation und ggf. einem vom jeweiligen Erkenner mitgelieferten Konfidenzmaß der plausibelste Parameter selektiert und mit dem Befehlspräfix zu einem Vollbefehl kombiniert, der nun für die Applikation verwertbar ist.
Wird innerhalb einer Integrationsperiode für ein Befehlspräfix kein passender Parameter gefunden, so evaluiert der Integrator das Befehlspräfix im nächsten Integrationsintervall erneut nach obigem Schema. Dieser Fall tritt etwa dann ein, wenn der zeitliche Abstand zwischen komplementären Eingaben des Benutzers größer ist als die gerade vorliegende Integrationsperiode oder eine Fehlerkennung einer Benutzereingabe stattgefunden hat.
Wenn der Integrator innerhalb des Integrationsfensters zueinander *konkurrierende*, i.e. widersprüchliche Kommandos feststellt, werden zunächst keine Applikationskommandos generiert bzw. versendet. Da in diesem Fall die Benutzerintention nicht eindeutig dekodierbar ist, wird ein weiteres Integratormodul (Error-Sentinel) aktiviert. Letzterer initiiert innerhalb der Applikation einen entsprechenden Rückfragedialog und beseitigt so interaktiv mit dem Benutzer den Fehlerstatus.

2.4 Implementierung der Integrationsalgorithmen

Bei der Late Semantic Fusion des multimodalen Inputs liegen im wesentlichen zwei Strategien zugrunde, die eine Integration in Echtzeit ermöglichen. In einem klassischen regelbasierten Ansatz wird die semantische Unifikation nach einem Satz von deterministischen Regeln mit Bezug auf die Grammatik durchgeführt. Das Regelwerk umfasst dabei ein domänenspezifisches Codebuch, mit dessen Hilfe die Regeln für die Länge des Integrationszeit-

fensters dynamisch zu verändern sind. In weiteren Regelblöcken werden sowohl Charakteristika der einzelnen Erkennermodule als auch der aktuelle zeitliche Kontext, in dem die Informationen am Integratormodul auflaufen, berücksichtigt. Die angewendeten logischen Beziehungen und Regeln resultieren aus einer detaillierten Analyse umfangreicher Usability-Untersuchungen zu multimodalen Interaktionen in verschiedenen Domänen (u.a. im automobilen Umfeld [3]).
Das zweite Integrationskonzept war stochastisch motiviert. Als Integrationsmethode wurde ein Ansatz aus dem Fuzzy-Controlling verwendet. Hierbei bestimmt der Integrator eine befehlsspezifische Wahrscheinlichkeit aus dem vom Monomodalerkenner übergebenen Konfidenzmaß (falls dieses existiert) und einem speziellen Score, der von der Verweildauer eines Befehls in der Verwaltungsqueue abhängt. Nach jeder Integrationsperiode wird diese Wahrscheinlichkeit des Befehls neu berechnet. So senkt etwa die Verweildauer eines Befehls in der Queue über mehrere Integrationszeitfenster dessen Wahrscheinlichkeit. Aus der Relation zu komplementären, redundanten bzw. konkurrierenden Befehlen wird schließlich eine bedingte Endwahrscheinlichkeit ermittelt. Eine Versendung des Vollbefehls bzw. fusionierter Befehlselemente findet nur dann statt, wenn die Endwahrscheinlichkeit oberhalb eines bestimmten Schwellwertes liegt. So schwächen etwa konkurrierende Informationen innerhalb einer Integrationsperiode ihre Endwahrscheinlichkeit gegenseitig ab. Bei redundanten Befehlen dagegen wird ein einziger Befehl generiert, wobei dessen Endwahrscheinlichkeit aus dem arithmetischen Mittel der Wahrscheinlichkeiten der beiden Einzelbefehle gebildet wird, um eine doppelte Versendung an die Applikation zu vermeiden.
Im Vergleich beider Algorithmen zeigte sich, dass der probabilistische Ansatz nach einer gewissen Anfangsphase aufgrund seiner nachtrainierten Änderungen der bedingten Wahrscheinlichkeiten für Kommandokombinationen bzw. –häufigkeiten flexibler ist. In Domänen, in denen die Kommandoabfolgen zeitlich weniger dicht sind sowie lediglich eine schwache Tendenz zu Modalitätenkombinationen besteht (wie etwa im Automobil), zeichnet sich hingegen der regelbasierte Ansatz durch sein zeitlich schnelleres Ansprechen aus.

2.5 Tools

Im Rahmen der Evaluierung der Integrationsalgorithmen wurde u.a. ein multimodaler Signalsimulator entwickelt. Hiermit können verschiedene Eingabesituationen dargestellt, Events auf Millisekunden genau geloggt und beliebig oft identisch reproduziert werden. So ist es möglich, sehr hohe Integratorlasten durch Paralleleingaben darzustellen oder die Leistungsfähigkeit des Kommunikationskanals bei der Übertragung komplexer Datenmengen zu untersuchen, ohne direkt die Erkenner selbst anschließen zu müssen. Des Weiteren können Szenarien nachgestellt werden, in denen Befehle falsch oder überhaupt nicht erkannt werden, was für die Untersuchung der Integration komplementärer Informationen sowie die Evaluierung des Fehlermanagements von besonderer Bedeutung ist. Außerdem ist es softwaretechnisch möglich, die Latenzzeiten zwischen Erkennung und Versenden der Nachricht für jeden Erkenner individuell zu variieren.

2.6 Domänenspezifische Anwendungen

Die Systemarchitektur wurde im Rahmen von Projekten in mehreren Domänen evaluiert und getestet. Im Projekt FERMUS (Fehlerrobuste Multimodale Sprachdialoge), in dem die

BMWGroup, DaimlerChrysler, SiemensVDO und der Lehrstuhl für Mensch-Maschine Kommunikation der Technischen Universität München miteinander kooperieren, stehen Ansätze zur Fehlervermeidung bzw. –behandlung im Fokus des Interesses. Zielapplikation ist ein per Sprache, Gestik und Haptik (Touchscreen, Hardkeys, Drehdrücksteller) zu bedienendes Informations- und Kommunikationssystem im Fahrzeug. Ebenso wird die Plattform im Projekt MIVIS (Multimodale Interaktion in virtuellen Szenarios) [2] verwendet. Hierbei handelt es sich um ein Internprojekt am Lehrstuhl für Mensch-Maschine-Kommunikation der TU München. Der Benutzer kann unter Verwendung natürlicher Sprache, Gestik sowie klassischer haptischer Eingabeparadigmen in virtuellen Welten navigieren und interagieren. In einer Kooperation mit dem Nuklearmedzinischen Institut am Klinikum Rechts der Isar in München wird die beschriebene Architektur in einem System verwendet, in dem Mediziner Tumore an einem virtuellen Avatar visualisieren und in einem entsprechenden multimodalen (sprachlich-haptischen) Interface weitere Informationen abrufen können. Im Projekt SOMMIA [3] (Kooperation zwischen SiemensVDO und dem Lehrstuhl Mensch-Maschine-Kommunikation der TU München) wurde das System als Testplattform für die Entwicklung eines intuitiven Bedienkonzepts für einen per Sprache und Haptik bedienbaren MP3-Player verwendet, bei dem starke geometrische (Zweizeilen-Display, 16 Zeichen pro Zeile) und ökonomische (LowCost-Ganzworterkenner) Constraints zu berücksichtigen waren.

3. Ausblick

In weiteren Arbeiten werden neue Formen der Late Semantic Fusion auf Basis genetischer Algorithmen [4] sowie einer effizienten Kombination wahrscheinlichkeits- bzw. regelbasierter Fusionsansätze evaluiert. Besonderes Augenmerk gilt des Weiteren einer verfeinerten Dynamisierung der Integrationszeitfenster sowie einer zusätzlichen multimodalen Integration emotionaler Muster [5] als Basis für eine systemseitige Adaption an den Benutzer.

4. Vorveröffentlichungen

[1] S. Oviatt, A. DeAngeli, and K. Kuhn*: Integration and Synchronization of Input Modes during Multimodal Human-Computer Interaction,* in Proceedings of Conference on Human Factors in Computing Systems: CHI'97, New York, ACM Press, 415-422

[2] F. Althoff, G. McGlaun, B. Schuller, and M. Lang: *Using Multimodal Interaction to Navigate in Arbitrary Virtual VRML worlds*. In Workshop on Perceptive User Interfaces (PUI), Nov. 2001, Orlando, USA

[3] G. McGlaun, F. Althoff, H.-W. Rühl, M. Alger, and M. Lang: *A Generic Operation Concept for an Ergonomic Speech MMI under Fixed Constraints in the Automotive Environment*, HCI 2001, New Orleans, LA, USA

[4] F. Althoff, M. Al-Hames, G. McGlaun, and M. Lang: *Towards a new Approach for Integrating Multimodal User Input Based on Evolutionary Computation*, ICASSP 2002, Orlando, FL, USA

[5] G. McGlaun, O. Sayan, F. Althoff, and M. Lang: *Towards Multimodal Detection and Classification of Emotional Patterns in Human-Machine-Interaction – Results of a Baseline Study*, SCI 2002, Orlando, FL, USA