Multimodale Suche I

Seminararbeit

vorgelegt am

Lehrstuhl für Praktische Informatik IV
Prof. Dr.-Ing. W. Effelsberg
Universität Mannheim

im
Juni 2006

im Rahmen des Seminars

Bridging the Semantic Gap
im SS 2006

Betreuer
Dr. Thomas Haenselmann

von

David Caccamo

Inhalt

1. Motivation

Gerade in der heutigen Zeit fällt es auf, dass der Informationsberg, der sich vor dem User des WorldWideWeb auftut, immer größer und immenser wird. Dabei steht der User ganz unten vor diesem Berg und versucht, ihn zu erklimmen. Nun scheint es so, dass zumindest der Informationsdurst des Users dabei doch so groß inzwischen ist, dass auch der Wille besteht, mit dieser gesamten Information zurecht zu kommen und sie auch für sich selbst nutzen zu wollen. Um dies tatsächlich bewerkstelligen zu können, ist es für den User nötig, sich geeigneter Mittel zu bedienen, da er ansonsten mit der ganzen Informationsmenge nicht zufrieden stellend umgehen könnte. Es ist nämlich heutzutage sehr schwierig geworden, auf Anhieb im Internet das zu finden, was man auch wirklich gesucht hat. Deswegen erlangt die Mensch-Computer-Interaktion immer größere Bedeutung. Nur durch Kombination menschlicher Fähigkeiten und wesentliche Verbesserung der Kollaboration von Seiten des Computers ist es möglich, wirklich verbesserte Suchanfragen und damit verbundene Ergebnisse zu generieren.
Der Computer soll dabei die Rolle eines „mitdenkenden“ Charakters bekommen, der nahezu selbständig auf die Anfragen und Bedürfnisse des Menschen reagiert. Natürlich muss dafür der Mensch selbst wieder genug Vorarbeit geleistet haben, um den Computer auch zu solchen Meisterleistungen zu bewegen.
Dennoch muss der Computer heute anders gesehen werden, als noch vor etwa 20 Jahren. Heute nämlich kann der Computer kommunizieren, früher wurde er „nur“ benutzt. Somit ist im Laufe der Zeit immer mehr ein Dialogsystem herangereift, das dem Endnutzer nur noch richtig zur Verfügung gestellt werden muss.
Dass dies noch nicht so geschehen ist, bemerkt man an der Popularität von Google beispielsweise. Obwohl die Suchergebnisse eher schlecht als recht sind – die Suche nach ‚Popcorn’ liefert mehr als 46 Mio. Ergebnisse – kann die Seite mehr als 10 Mio. Kicks / Woche verbuchen. Das mag aber natürlich an der absolut benutzerfreundlichen Oberfläche liegen und vielleicht auch daran, dass doch irgendwo gute Ergebnisse geliefert werden oder vielleicht einfach nur daran, dass Google zur richtigen Zeit kam und jeder es gut fand – es gab ja auch kaum Alternativen. Die Tatsache, dass nun bei Google darüber nachgedacht wird, soziale Annotationen hinzuzunehmen, die Bildersuche zu verbessern und allgemein die Suche nicht nur mehr von einem Algorithmus leiten zu lassen, zeigt doch, dass auch bei den altbewährten Systemen der Trend erkannt wurde. Der Nutzer will mehr Komfort bei gleich bleibend leichter Bedienbarkeit und ohne allzu große Umwege.

2. Multimodal – was bedeutet das?

2.1 Zum Begriff

Unter Multimodalität versteht man die Kombination mehrerer Modalitäten. Multimodal ist dabei in verschiedensten Richtungen verwendbar. Für unsere Zwecke stellen die menschlichen Sinne die Modalitäten dar. Die Verbindung mehrerer menschlicher Sinne als Input in ein System soll damit die Grundlage für eine bessere Suchanfrage liefern. *[4]*

2.2 Die menschlichen Sinne

Zunächst einmal ein Überblick über die wichtigen Sinne, die nutzbar und vor allem kombinierbar sind:

Modality	**Examples**
Visual	Gaze
	Image Capture
Auditory	Voice Input (as part of the language faculty)
	Nonspeech audio Input (as part if the music faculty)
Haptic/Kinesthetic	
Touch	Pressure
	Text
Hand movement	Sign language
	3D motions
	Writing motions
	Drawing or other nontextual gestures
Head movement	Lip reading
	Head movement for pointing
	Facial expressions
Other body movement	Movement captured in dance or sports

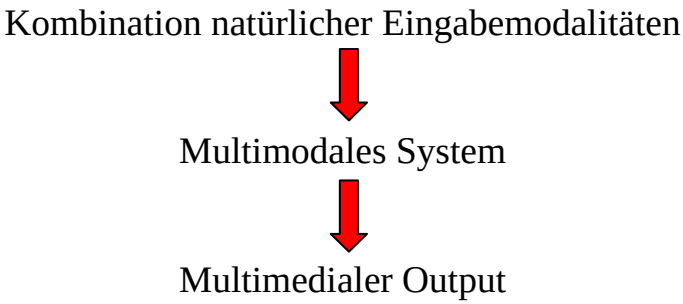
Tabelle 1, [6]

Man erkennt daran, dass es sich um die drei Sinne des Sehens, Hörens und Tastens handelt. Wobei gerade die Spracheingabe jedem bekannt vorkommen dürfte. Wichtig ist die Unterteilung in Sprach- und Nichtspracheingabe. Ebenso spielt die Körperbewegung eine sehr wichtige Rolle. *[6]*

Einzeln mögen diese Modalitäten als Eingabemöglichkeit schon ganz gut klingen, jedoch erst die Verbindung von zwei oder drei macht sie richtig gut und somit auch erst multimodal.
Heute bereits wird beispielsweise im Auto bei der Eingabe ins Navigationsgerät die Spracheingabe mit einem Touchscreen kombiniert, um den Nutzer mehr Komfort und leichter Bedienung zu bieten.
Die Kombination aus Nichtsprach-Eingabe (also Musikeingabe) und Text oder die Kombination aus Gesten und Spracheingabe werde ich später genauer betrachten, da darauf die zwei Systeme basieren, die ich erläutern möchte.

3. Die multimodale Suche

3.1 Definition



Die obere Grafik zeigt den Aufbau eines Systems auf, welches einen multimodalen Input verarbeiten kann, um daraus einen multimedialen Output zu generieren.
Der Input stellt sich aus kombinierten Modalitäten von *Tabelle 1 [6]* zusammen. Im multimodalen System kann diese Eingabe, aus sagen wir zwei Modalitäten, tatsächlich kombiniert genutzt werden, um die Suchanfrage und damit den multimedialen Output hervorzubringen.
Dabei ergänzen sich die Modalitäten, es entsteht also nicht etwa die doppelte Menge an Information, die dann getrennt analysiert wird und damit doppelt so viel Suchergebnisse liefert. Ganz im Gegenteil, die eine Eingabemodalität wirkt für die andere unterstützend, um eben die Suche einzuschränken und damit zu verfeinern. Ziel ist es, qualitativ bessere Ergebnisse zu liefern und nicht eine quantitative Steigerung zu erreichen.
Insgesamt betrachtet, gibt es zwei Varianten für Systeme, die multimodalen Input verarbeiten können. Es muss also eine Abgrenzung der Varianten voneinander durchgeführt werden.

3.2 Abgrenzung

Grob gesehen, könnte man sagen, dass die eine Variante eher eine tatsächliche Suche mithilfe multimodaler Interaktion darstellt, wohingegen bei der anderen Variante eine Eingabe auf ihre multimodalen Gegebenheiten hin untersucht wird, um daraus eine Reaktion zu generieren.
In der ersten Variante steht der User als aktiver Sucher im Vordergrund. Er gibt multimodal etwas in den Computer ein, um vom Computer die Erfüllung seiner Suchanfrage geliefert zu bekommen. Damit ist der Computer ausführende Hand des Nutzers.
Die zweite Variante stellt den User eher passiv dar, weil er nämlich mit dem Computer eine regelrechte Kommunikation eingeht. Der Computer versucht dann selbständig die Eingabe des Nutzers – also sein Handeln und Verhaltern – zu analysieren, um daraus genügend Information zu bekommen, die dem Computer zu automatisierten Reaktion verhilft. Der Computer soll dem User also als echter Dialogpartner zur Verfügung stehen.

Um einen besseren Einblick in die Funktionalität beider Varianten zu bekommen, werde ich nun zwei bereits existierende Systeme vorstellen, die jeweils auf einer der beiden Varianten aufbauen.

3.3 QBH – Query by humming

3.3.1 Einführung

Query by humming steht für Suche durch Summen.
Ich stelle dieses System repräsentativ für die erste erwähnte Variante vor. Das Fraunhofer Institut hat zusammen mit Audio Engineering Society AES dieses Projekt vorangetrieben und ist noch immer federführend in diesem Gebiet.
Die Idee bei QBH-Systemen besteht darin, dem User mehr als nur eine textbasierte Suche an die Hand zu legen, um damit Musikstücke zu finden. Schließlich sucht man meistens etwas, das man nicht kennt. So hat man meist bei einem Musikstück die Melodie im Kopf, weiß aber nicht immer den Interpret oder gar den Titel des Stücks. Das System macht sich dies gerade zu Nutze, indem es dem User ermöglicht, eine gesummte oder gepfiffene Eingabe mit Hilfe eines Mikrofons in den Computer einzugeben. Anhand verschiedener Umwandlungen der Eingabe, wird das Gesummte in eine Darstellung umgewandelt, die für einen Vergleich ausreichend und zufrieden stellend ist.
Das nun hier gezeigte System **Queryhammer** *[8]* wurde auf einer Konferenz der AES von einer Forschungsgruppe der TU Berlin vorgestellt.

Zum Vergleich der Musikstücke wird sich einer MPEG-7-Datenbank bedient, die dem System zugrunde liegt. In dieser Datenbank sind alle Referenzstücke bereits vollständig transkodiert und mit MPEG-7-Deskriptoren annotiert. Das eingegebene Musikfragment muss nun in die gleiche Darstellung gewandelt werden, um einen Vergleich durchführen zu können. Beim Forschen hat sich herausgestellt, dass sich die Konturendarstellung der Eingabe besonders gut eignet, um einen Vergleich vorzunehmen. Im Grunde wird die Qualität der Eingabe so sehr heruntergeschraubt, so dass überhaupt das ganze System einen Sinn ergibt und annähernd ein gutes Ergebnis liefert.
In *3.3.2* wird mit Hilfe von *Abb. 1* das System genauer erklärt. Auch werde ich auf die MPEG-7-Deskriptoren eingehen, die die Vergleichbarkeit der gesummten Eingabe erst möglich machen.

3.3.2 Die erste Umwandlung

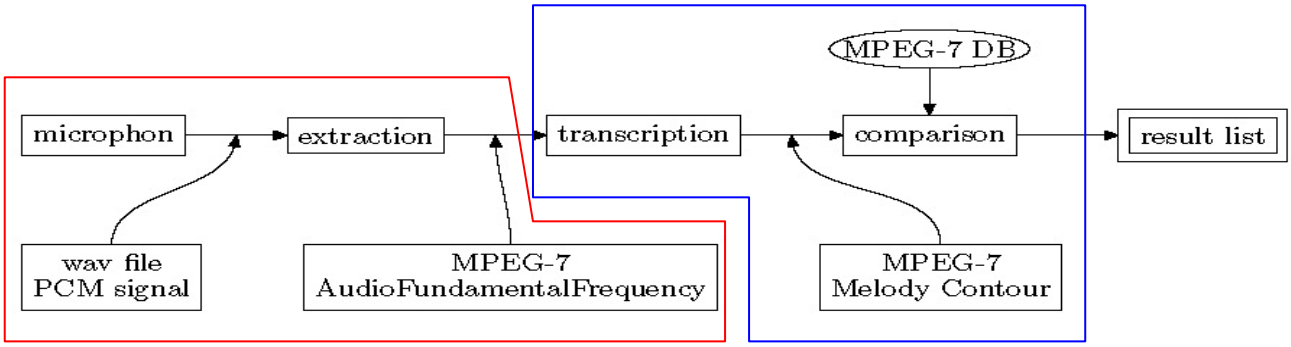


Abb. 1 [8]

Abb. 1 zeigt den gesamten Verlauf der Umwandlung von der Eingabe über das Mikrofon bis hin zur Ergebnisliste am Ende der beiden Umwandlungen und der Vergleiche.
In *Abb. 2* kann man das Eingabesignal in optimaler Qualität sehen. So wird es in Wirklichkeit nicht vorliegen – außer für die Stücke auf der Datenbank. Die Eingabe erfolgt über Java-Applets.

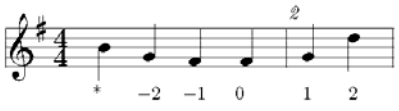


Abb. 2 [8]

Das Signal wird nun über einen Bandpass gefiltert, um am Ende ein Wav-Signal im Frequenzbereich von 80-800 HZ zu haben. Dadurch werden Brummgeräuche oder viel zu hohe Töne einfach abgeschnitten. Daraus ergibt sich dann *Abb. 3*, wo man die einzelnen Töne mit ihrer Amplitude erkennen kann.

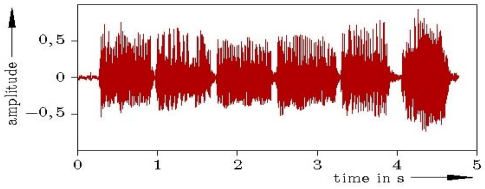


Abb. 3 [8]

Nach diesem preprocessing-Schritt ist es für den weiteren Verlauf notwendig, die einzelnen Noten zu erkennen. In dieser ersten Umwandlungs-stufe, beschränkt man sich darauf, die Anzahl der Töne und ihre Grundfrequenz zu erkennen, da der verfügbare MPEG-7-Deskriptor nicht in der Lage ist, mehr Informationen zu speichern.

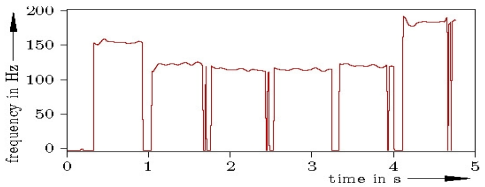


Abb. 4 [8]

Durch einen Abstandsmessung-Algorithmus werden die einzelnen Töne und die grundlegenden Frequenzen dazu erkannt (*Abb. 4*). Hier ist auch gut zu erkennen, dass ein Ton nicht aus einer einzelnen Frequenz besteht, sondern

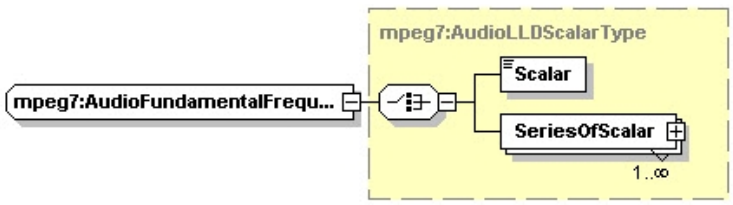


Abb. 5 [8]

die Frequenz schwankt innerhalb des Tones. Gerade deswegen wird die gesamte Bearbeitung der Suchanfrage in zwei
Teile getrennt. Der MPEG-7-Deskriptor AudioFundamentalFrequency (*Abb. 5*) ist nämlich in der Lage genau die nun gesammelten Informationen automatisch abzuspeichern. So werden zu jedem Ton Höchst- und die Tiefstfrequenzen als Bereich gespeichert, Zwischenwerte werden ausgelassen, da sie vernachlässigbar sind.
Außerdem kann dieser Deskriptor auch noch Rhythmus-Informationen speichern, zwar nur in sehr minimaler Form, aber das ist zumindest ein Anfang. Diese Rhythmus-Infos wurden im Zuge der Abstandsmessung ermittelt. Das System erkennt die Schläge (Beats) und hat dazu die Zeitachse. Daraus wird der Wert beats per minute (BPM) errechnet, indem hoch gezählt wird. Die errechnete Zahl wird im Deskriptor mit abgelegt. Sie liefert keine Aussage über Takt oder genaue Längen der Noten – der Deskriptor ist nicht so mächtig – aber über die BPM-Zahl kann zumindest die Ergebnisliste ein wenig minimiert werden. Schließlich soll diese Info unterstützend und nicht zusätzlich verwirrend wirken.
Am Ende dieser Umwandlungs- bzw. Extraktionsphase hat Queryhammer die Noten mit den Grundfrequenzen und einen MPEG-7-Deskriptor vorliegen. Beides wird nun an die zweite Stufe, die Umschreibungsphase (transcription) weitergeleitet, wo letztendlich die Darstellung entsteht, die zum Vergleich benötigt wird.

3.3.3 Die zweite Umwandlung

In dieser Phase müssen nun die einzelnen tatsächlichen Noten erkannt werden. Dafür werden nun die Infos aus dem AudioFundamentalFrequency-Deskriptor benutzt. Aus den übermittelten Frequenzbereichen werden Mittelwerte gebildet, die dann mit den Formeln aus *Abb. 8* den wirklichen Notenwerten zugeordnet werden. Dabei werden wohltemperierte Noten anhand eindeutiger Frequenzen sofort zugeordnet, abweichende Frequenzen werden durch die erste Formel auf der chromatischen Skala errechnet und schließlich alle davon abweichenden, mit der Cent-Berechnung (c(f)).
Wichtig ist hierbei, dass bisher nur die Anzahl der Noten bekannt ist, über die beiden

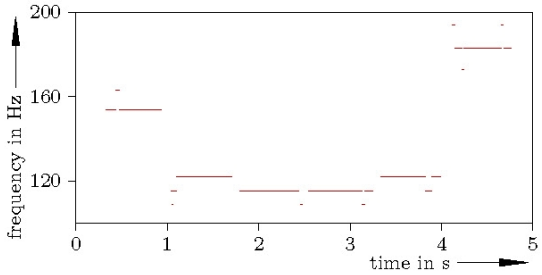


Abb. 6 [8]

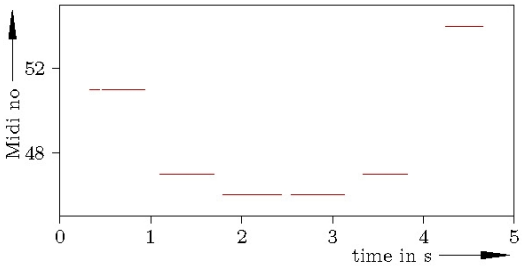


Abb. 7 [8]

Noten chrom. Skala: $f(n) = f_0 * 2^{n/12}$
Abweichende Töne: $c(f) = 1200 * \log_2 * (f/f1)$
$|c(f)| > 50$ c => neuer Ton
Event > 80 ms => neuer Ton

Abb. 8 [8]

Ungleichungen in *Abb. 8* werden die einzelnen Noten erst erkannt und wie oben beschrieben auch zugeordnet. Bedeutet, erst ab einer Abweichung von mehr als 50 Cent (im Betrag) oder aber wenn ab einer Pause von mehr als 80ms wird ein neuer Ton erkannt und die Notenwerterkennung durchgeführt. Das Ergebnis des ganzen Erkennens ist in *Abb. 6* gezeigt. Es sind dennoch aber kleine Ungenauigkeiten erkennbar, weshalb nun die Umwandlung des Fragments in Midi erfolgt (*Abb. 7*), damit werden auch die letzten (unnötigen) Informationen aus dem eingegeben Fragment entfernt; sie würden das Resultat nicht verbessern. Nun kommt der wichtigste Schritt; zwischen den einzelnen nun erkannten Noten wird der Frequenzunterschied in Cent-Beträgen errechnet (Contour Value).

Contour value	Change of c(f) in cents
-2	$c \leq -250$
-1	$-50 \leq c < -250$
0	$-50 < c < 50$
1	$50 \leq c < 250$
2	$c \geq 250$

Abb. 9 [8]

Dafür wird dem ersten Ton der Nullwert zugeteilt und die folgenden dann abhängig von *Abb. 9* zugeordnet. Jedes Event hat somit einen Notenwert und das ganze Stück besitzt eine Kontur. Diese gesamte Information wird im MelodyContourType-MPEG-7-Deskriptor (in Contour und Beat, *Abb. 10*) gespeichert.
Die Datenbank enthält alle Musikstücke ebenfalls im Midi-Format mit den zugehörigen MPEG-7-Infos. Demnach kann nun der Vergleich erfolgen. Dabei werden die Konturen aus Eingabe und Datenbank miteinander verglichen und die Differenzen aus den Konturwerten summiert. Das Datenbankstück, welches am Ende die kleinste Summe an Kontorwerten besitzt, ist (hoffentlich) das vom User gesuchte und wird dann mit den schlechteren Ergebnissen mit ausgegeben.

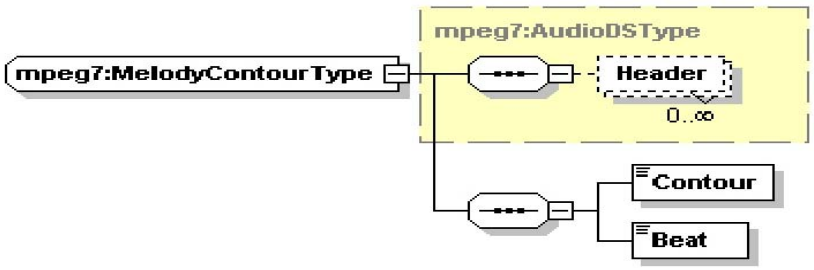


Abb. 10 [8]

3.3.4 Heutige Nutzung

Zwei gute und einfach zu bedienende Implementationen eines QBH-Systems sind im Internet unter *[1]* und *[2]* zu finden. Das erste System ist dabei die kommerzielle Variante und wurde vom Fraunhofer Institut entwickelt. Bei der zweiten Umsetzung handelt es sich um ein Privatprojekt. Die Ergebnisse sind ansehnlich und gerade das zweite System bietet sehr komfortable Suchoptionen.

3.4 Interactive Storytelling

3.4.1 Einführung

Das Gebiet des Interactive Storytellings umfasst einen sehr großen Bereich. Es geht hierbei nicht nur allein um das Geschichten erzählen. Das Ganze ist abstrahierter *[9]*. Der außenstehende User wird in eine virtuelle Welt hineinprojiziert und kommuniziert darin mit einer künstlichen Intelligenz bzw. einem Avatar. Beispiele für einen Avatar sind Robert T-Online oder Lara Croft. Somit (anscheinend) selbständig handelnde und denkenden Computerwesen. Der Avatar versteht es, die Aktionen des Users aufzuschlüsseln und daraus für sein eigenes Handeln eine Reaktion zu generieren. Somit steht der Nutzer nicht mehr als rein aktiver Sucher im Vordergrund, sondern liefert durch sein Handeln den multimodalen Input, der zum Dialog mit dem Computer führt.

In Deutschland ist vor allem das Zentrum für graphische Datenverarbeitung ZGDV *[3]* für die Realisierung vieler wichtiger und heute auch eingesetzter Projekte verantwortlich. Gerade im Bereich der virtuellen Museumsführung oder auch für die WM 2006 (Servingo) wurden erfolgreiche Projekte entwickelt, die effizient zum Einsatz kommen und von den Kunden gut akzeptiert werden.

In dem von mir vorgestellten System handelt es sich um ein Projekt *[5]* der Universität Teesside (UK), welches sich nun genau mit dem interaktiven Geschichteerzählen befasst. Es werden die Gesten und das Gesprochene des außenstehenden Users als multimodaler Input für das System genutzt. Daraus sucht sich das System die – vermeintlich – wichtigen Informationen, auf die es entsprechend reagieren kann. Natürlich wäre es utopisch, anzunehmen, dass das System dies problemlos und vor allem einwandfrei schafft.
Die Grenzen und Möglichkeiten gerade dieses Systems werde ich im Folgenden aufzeigen.

3.4.2 Das System

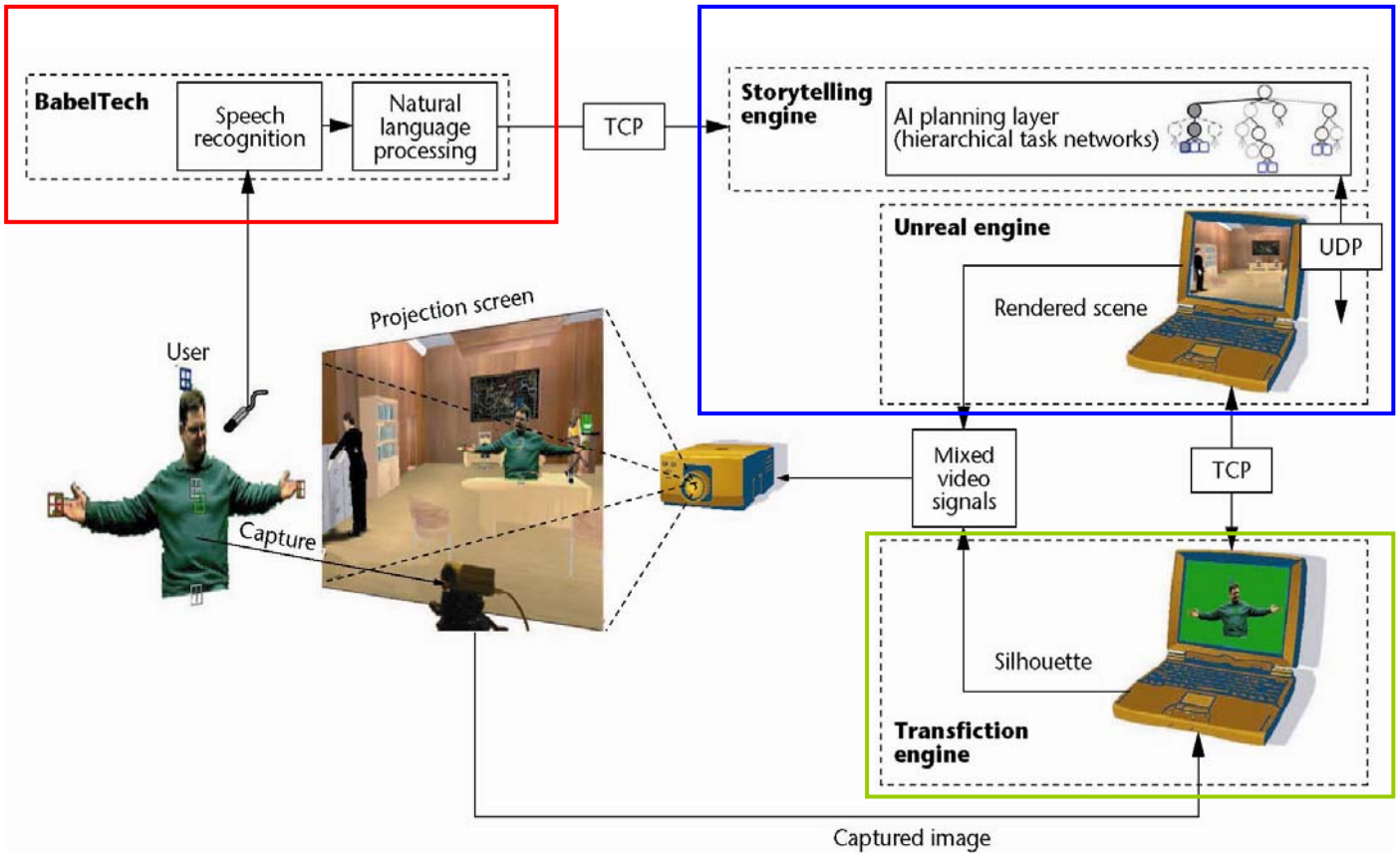


Abb. 11 [5]

Abb. 11 zeigt das komplette System, wie es von der Universität Teesside entwickelt wurde. Dabei haben sie aber einige bereits implementierte Bestandteile mit verwendet, die nicht unbedingt neu entwickelt werden mussten. Schließlich muss das Rad nicht neu erfunden werden und bisher bewährte Software muss daher nicht unbedingt selbst erstellt werden *[7]*.

3.4.3 Die Gesten- und Spracherkennung

Das System besteht grob gesehen aus drei Teilsystem (farblich unterlegt). Der rote und der grüne Part sind dabei für die multimodale Erkennung zuständig. Der außenstehende User wird von einer Kamera gefilmt und liefert per Mikrofon zudem den sprachlichen Input für das System.
Das rote Teilsystem nimmt die Info vom Mikrofon auf ist für die **Spracherkennung** zuständig. Es wurde nicht selbst entwickelt, es handelt sich um die Babeltech Software, die in der Lage ist, zunächst eine einfache Spracherkennung durchzuführen und anschließend versucht aus dem erkannten, die signifikanten Aussagen herauszufiltern.

Dies geschieht durch Kombination von zwei Analysen:

1. Hauptwortanalyse (W-Wörter bspw. dienen als Hinweis auf eine Frage)
2. Verbenanalyse (Verben und ihre unmittelbare Umgebung/Attribute werden als Richtung für den Sinn des Gesagten untersucht)

Diese Information wird direkt an das blaue Teilsystem per TCP-Socket weitergeleitet.
Das grüne Teilsystem (Transfiction Engine) bekommt die Daten der Kamera geliefert und hat somit den User vor einer Blankowand als Input vorliegen. Dadurch, dass der User vor einer Blankowand gefilmt wird, ist für das grüne System die Umriss-Erkennung des User nicht besonders schwer durchzuführen.
Das schwierige ist es nun, aus den Gesten eine mögliche Aussage des Users herauszufiltern. Dafür muss eine Silhouetten-Erkennung durchgeführt werden, indem der Videoinput in 4 x 4 Pixel-Regionen aufgeteilt wird und diese dann mit der Walsh-Hadamard Funktion analysiert werden. Hierbei werden in diesen kleinen Regionen Farbunterschiede erkannt bzw. Farbwerte ermittelt. Sind zwei benachbarte Regionen vom Farbunterschied her nicht über einem bestimmten Schwellwert, so werden sie als eine Region erkannt. Findet hingegen eine Überschreitung des Schwellwerts statt, so markiert die Transfiction Engine diesen Bereich und benutzt ihn im weiteren Verlauf als Bezugspunkt, um die Gestenerkennung durchzuführen.

Abb. 12 [5]

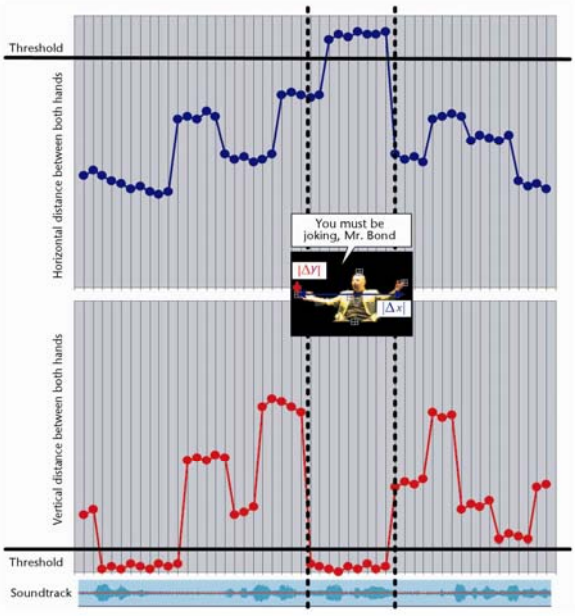


Abb. 13 [5]

Abb. 12 und *13* zeigen auf, wie die unterschiedlichen Regionen markiert werden und wie anschließend die ermittelten Koordinaten der Transfiction Engine dazu dienen die Armweite zu errechnen, um in diesem Fall dann herauszufinden, das der User mit weit geöffneten Armen auf seinem Stuhl

sitzt. Dieses Herausfinden läuft nicht selbstverständlich ab. Der Transfiction Engine liegt ein Gestenlexikon zu Grunde, welches mit vordefinierten Gesten und all ihren möglichen Interpretationen gefüllt ist. Natürlich erkennt man daran schon, dass einer Geste mehr als eine Interpretation zugeordnet werden kann, dies wird auch so gemacht.
Das aufgenommene Material mitsamt den erkannten Gesten wird auch per TCP-Socket an das blaue System sofort weitergeleitet.
Das blaue System bildet das Herzstück des gesamten Systems, da hier zum einen die erkannten Sprachaussagen mit den erkannten Gesten kombiniert werden und zum anderen die Überlagerung vom Gefilmten mit der Grafik-Engine vollzogen wird. Zunächst zur Überlagerung. Die Universität benutzt als Grafik-Engine die Unreal Tournament 2003 Engine, welche grafische sehr ausgefeilt ist und zudem eine Kollisionserkennung beinhaltet. Wird nämlich das Gefilmte mit der Unreal-Engine überlagert, so muss zum Beispiel der User hinter einem Tisch sitzen und nicht etwa in ihm. Dies geschieht automatisch. Das überlagerte Material wird per Projektor auf eine Leinwand projiziert.
Das Herausfiltern der Hauptaussagen aus der Gesten- und Sprachanalyse stellt die größte Herausforderung dar. Jedoch bietet die Kombination der Modalitäten erst die Möglichkeit, überhaupt die Aussagen zu filtern, da die erkannten Gesten mit Hilfe der erkannten Aussage aus der Spracherkennung dezimiert werden. Die Hauptaussagen werden nun im Hintergrund an die Storytelling Engine weitergeleitet.
Diese Engine stellt einen Entscheidungsbaum *[5]* dar, der die Grenzen der überhaupt möglichen Geschichte enthält. Erkennt nun die Engine ein bestimmtes Verhalten des Users, so schickt sie den dadurch resultierenden Pfad mit seinem Endknoten an den Avatar, der sich auch in der Unreal Engine befindet. Der Avatar ist nun in der Lage, „automatisch“ auf das Handeln des Nutzers zu reagieren und ist damit im Dialog mit dem Nutzer.

3.3.4 Heutige Nutzung

In *[3]* besteht die Möglichkeit auf die Projekte der ZGDV Einblick zu halten, um sich genauer den heutigen Stand des Interactive Storytellings anzuschauen.

4. Fazit

Insgesamt kann man sagen, dass der heutige User ein sehr großes Interesse für die hier vorgestellten Technologien und Systeme mitbringt. So wird heut auch schon im Radio damit geworben, dass die Zuhörer per MMS ein gesummtes Lied an den Sender schicken können, welches dann analysiert wird. Nach erfolgreicher Erkennung des Liedes, bekommt der Zuhörer eine Nachricht auf sein Handy mit Titel und Interpret. Auch die virtuellen Museumsführer genießen sehr große Beliebtheit, schließlich hat der Besucher nicht mehr das Gefühl, vielleicht eine „dumme“ Frage stellen zu können.
Auffallend ist vor allem, dass die Systeme immer besonders leicht zu bedienen, sehr komfortabel und zudem natürlich effizient sein müssen. Sonst ist der Nutzer nicht zu begeistern.
Zumindest der Grundstein ist also für weitere Forschung und Entwicklung gelegt. Jedoch hapert es noch entscheidend an der guten Umsetzung der Ideen. Meine Darstellung der Systeme scheint im ersten Moment ziemlich perfekt zu wirken, aber bleiben dem Nutzer doch sehr große Einbußen.

Gerade im Bereich der Musiksuche, hat der Nutzer zwar sehr viele Freiheiten (gut an den Internetbeispielen erkennbar), jedoch machen diese Freiheiten das tatsächliche Finden sehr schwierig. Es bestehen keine Vorgaben, wie schnell gesummt werden soll oder welche Passage gesummt wird. Auch die Länge der Eingabe ist nicht vorgegeben. So bleibt es zumindest heut noch eher Zufall, wenn das Gesummte tatsächlich gefunden wird, schließlich gibt es bei so wenig existierenden Noten doch schon eine gehörige Vielzahl an Liedern in den verschiedensten Kategorien. Ein weiteres fundamentales Problem stellen die MPEG-7-Deskriptoren selbst dar, die nicht ausreichend Möglichkeiten bieten, alle Informationen anzuspeichern.

Das Interactive Storytelling hat mit anderen Problemen zu kämpfen. Hier spielt die richtige Erkennung der Gesten vor allem eine entscheidende negative Rolle. Eine Verbesserung in diese Richtung wäre nicht nur wünschenswert, sondern zwingend. Sonst bleibt nämlich der mögliche Handlungsstrang doch sehr eingeschränkt.
Es gibt also noch viel zu entwickeln, was aber auch nicht unmöglich ist.

5. Literatur

[1]
http://www.musicline.de/de/melodiesuche/input

[2]
http://www.melodyhound.com/

[3]
Zentrum für graphische Datenverarbeitung e.V.
http://www.zgdv.de/zgdv/zgdv/departments/z5/Z5Projects

[4]
Zeljko Obrenovic, Dusan Starcevic, "Modeling Multimodal Human-Computer Interaction", IEEE Computer, Vol. 37, No. 9, September 2004, pp. 65-72

[5]
Marc Cavazza, Fred Charles, Steven J. Mead, Olivier Martin, Xavier Marichal, Alok Nandi: Multimodal Acting in Mixed Reality Interactive Storytelling. IEEE MultiMedia 11(3): 30-39 (2004)

[6]
M.M. Blattner, Ephraim P. Glinert: Multimodal Integration. In IEEE Multimedia, 3(4):14-24, Winter 1996.

[7]
Ed Kaiser, David Demirdjian, Alexander Gruenstein, Xiaoguang Li, John Niekrasz, Matt Wesson, and Sanjeev Kumar."Demo: A Multimodal Learning Interface for Sketch, Speak and Point Creation of a Schedule Chart," Proceedings of the Sixth International Conference on Multimodal Interfaces (ICMI 2004), State College, Pennsylvania, USA, October 14-15, 2004, pgs. 329-330.

[8]
Batke, Jan-Mark; Eisenberg, Gunnar; Weishaupt, Philipp; Sikora, Thomas:A Query by Humming system using MPEG-7 Descriptors. In: Proceedings of the 116th AES Convention, 2004

[9]
Bee-Wah Lee, Alvin W. Yeo: Integrating sketch and speech inputs using spatial information. ICMI 2005: 2-9