Bauhaus-Universität Weimar · Fakultät Medien
Studiengang Mediensysteme

Diplomarbeit

Multimodale Interaktion mit Spracheingabe und Zeigeoperation für Virtuelle Umgebungen

Petros Kapakos
Matrikelnummer 11277
geb. am: 16.01.1977 in Ehingen (Donau)

18. September 2008

Betreuer: Prof. Dr. Bernd Fröhlich
Externer Betreuer: Dipl.-Phys. Thorsten Holtkämper

Danksagung

Ich möchte mich an dieser Stelle bei Prof. Dr. Bernd Fröhlich für seine Bereitschaft bedanken, diese Diplomarbeit anzunehmen und sie über die Distanz hinweg zu betreuen. Ebenfalls geht mein Dank an Dr. Manfred Bogen, Leiter der Abteilung Virtual Environments am Fraunhofer Institut IAIS, der mir diese Arbeit ermöglicht hat. Weiterer Dank geht an meinen externen Betreuer Dipl.-Phys. Thorsten Holtkämper und an weitere ehemalige und aktuelle Mitarbeiter der Abteilung wie Jonas Schild, Kai Riege, Sascha Scholz, Armin Dressler, Andreas Bernstein und Jürgen Wind. Ein besonderer Dank geht natürlich an meine Familie, die mich unterstützt hat.

Inhaltsverzeichnis

Abbildungsverzeichnis

Tabellenverzeichnis

1. Einleitung

Dieses Kapitel gibt einen Überblick über den Inhalt der Arbeit. Nach einer kurzen Einführung in die Thematik wird auf die Motivation, das Ziel und den Aufbau der einzelnen Kapitel eingegangen.

1.1. Motivation

Applikationen der Virtuellen Realität (VR) werden in vielen unterschiedlichen Bereichen wie Unterhaltung, Medizin, Forschung oder Wirtschaft verwendet. Für die Steuerung der unterschiedlichen Anwendungen gibt es kein generelles Konzept. Das liegt einerseits daran, dass sie unterschiedlich in ihrer Funktionalität und Komplexität sind, andererseits existieren für die Interaktion in Virtuellen Umgebungen eine Vielzahl verschiedener Eingabegeräte, Eingabemethoden und Displays, an die die Steuerung angepasst werden muss. Zusätzlich müssen auch Erfahrung und Hintergrund der Benutzer[1] berücksichtigt werden, welche die VR-Applikation steuern. Hinsichtlich dieser Faktoren ist es eine große Herausforderung, ein passendes Interaktionskonzept zu entwickeln, welches den Anforderungen der jeweiligen VR-Applikation genügt.

Im Rahmen des VRGeo-Projektes ([VRGeo, 2008]) am Fraunhofer Institut für Intelligente Analyse- und Informationssysteme (IAIS) wird die Nutzbarkeit von Technologien und Techniken der Virtuellen Realität für die Geowissenschaften erforscht. Ein Bestandteil dieses Projektes ist der VRGeo Demonstrator, eine VR-Applikation, mit der seismische Volumendaten visualisiert werden. Für die Systemsteuerung (siehe 2.2) im Demonstrator gibt es im Moment zwei Möglichkeiten: externe Geräte oder 3D-Menüs (siehe 2.2.1). Bei der ersten Möglichkeit hat der Benutzer über einen Personal Digital Assistant (PDA) oder Tablet-PC Zugriff auf die Menüs und damit auf alle Funktionen der Software. Bei der zweiten Option, die momentan hauptsächlich benutzt wird, geschieht die Systemsteuerung über 3D-Menüs, die in der VR-Szene des Demonstrators integriert sind. Diese bieten zwar eine geeignete Lösung für die Systemsteuerung, bringen aber auch Probleme mit sich. Bedingt durch die Tiefe der Menü-Hierarchie wird bei der Ausführung eines bestimmten Befehls über das Menü der Fokus des Benutzers von der VR-Szene

[1] Alle in dieser Arbeit verwendeten Rollen wie „Benutzer“, „Anwender“ und „Proband“ sind geschlechtsneutral gemeint und gelten gleichermaßen für weibliche und männliche Personen.

genommen. Dieser Blickwechsel macht eine Neufokussierung und Neuorientierung notwendig und erfordert einen hohen Grad an Aufmerksamkeit. Ein weiteres Problem ist, dass durch die Navigation in den 3D-Menüs eine natürliche Interaktion mit dem VR-System nicht gegeben ist. In einer Virtuellen Umgebung tritt der Benutzer in eine synthetische Welt ein, in der er Objekte direkt manipulieren kann und seine Bewegungen und die Wirkungen seiner Aktionen unmittelbar erfahren kann. Menüs sind in diesem Zusammenhang unnatürlich und beeinträchtigen die Illusion einer Virtuellen Realität.

Eine in den VRGeo Demonstrator integrierte Spracheingabe soll die genannten Probleme lösen und ihr Nutzen soll in einer abschließenden Benutzerstudie untersucht werden.

1.2. Ziel der Arbeit

Ziel der Arbeit ist es, aufbauend auf die bestehende Systemsteuerung des VRGeo Demonstrators, ein Konzept für eine intuitive und natürliche Interaktionsmethode zu entwickeln, welche Spracheingabe und Zeigeoperation zum Mittel hat. Dabei wird ausgehend von den Anforderungen dieser VR-Applikation auf die Besonderheiten der multimodalen Interaktion in Systemen der Virtuellen Realität eingegangen.

Im Mittelpunkt steht die Entwicklung eines Interaktionskonzeptes im Expertenmodus, das trotz alledem aber auch für Einsteiger leicht zu erlernen und zu benutzen sein soll. Das Hauptziel dieser Arbeit ist es zu untersuchen, ob Spracheingabe als Interaktionsmöglichkeit akzeptiert wird und wenn ja, wie natürlich kombinierte Interaktion mit Sprache und Zeigeoperation für die Systemsteuerung in einer Virtuellen Umgebung ist.

Hierzu soll in einer abschließenden Benutzerstudie festgestellt werden, ob eine derartige Interaktion einen Mehrwert für den Benutzer darstellt. Als Testumgebung wird der obengenannte VRGeo Demonstrator benutzt, der in Kapitel 3.3 ausführlich beschrieben wird.

1.3. Aufbau

In Kapitel 2 wird zunächst auf die Eigenschaften Virtueller Umgebungen eingegangen und deren Elemente werden beschrieben. Danach werden konventionelle Interaktionsgeräte und Interaktionstechniken diskutiert, ebenso wie natürliche Benutzerschnittstellen und die multimodale Interaktion. Anschließend wird eine Auswahl an Spracherkennungssystemen und an handelsüblichen Mikrofonen vorgestellt, die zur Spracherkennung geeignet sind.

In Kapitel 3 wird die Entwicklungsumgebung für das Interaktionskonzept präsentiert. Es wird die Sprachschnittstelle SAPI 5.3 von Microsoft sowie das VR-Framework Avango™ und der VRGeo Demonstrator, die Applikation, an der die Sprachschnittstelle angebunden wird, vorgestellt. Es werden die Interaktionsgeräte und Interaktionstechniken im Demonstrator erklärt, sowie dessen Objekte in der VR-Umgebung. Anschließend wird das TwoView-Display vorgestellt.

In Kapitel 4 wird der Entwurf des Interaktionskonzeptes und dessen Implementierung präsentiert. Es wird auf den Sprach-Server und dessen Benutzeroberfläche eingegangen sowie auf die Implementierung am Demonstrator. Des Weiteren wird die Grammatik für die Sprachbefehle zur Steuerung der VR-Applikation vorgestellt.

In Kapitel 5 wird eine Benutzerstudie und die Auswertungen der Ergebnisse daraus präsentiert. Das letzte Kapitel 7 fasst diese Arbeit zusammen und schließlich werden Tendenzen und Fragestellungen besprochen.

2. Interaktion in Virtuellen Umgebungen

Die in dieser Arbeit vorgestellte Sprachsteuerung für Virtuelle Umgebungen basiert auf Techniken der multimodalen Interaktion. Diese bezeichnet eine Interaktionsform, bei der mehrere Modalitäten, wie Sprache und Gestik verwendet werden. In diesem Kapitel wird der Aufbau von VR-Systemen und deren Komponenten aufgeführt. Anschließend wird auf verschiedene Methoden der Interaktion für Virtuelle Umgebungen eingegangen.

2.1. Virtuelle Umgebungen

Eine Virtuelle Umgebung (oder Virtuelle Realität) beschreibt eine künstliche, räumliche Welt, die aus der Perspektive des Benutzers dargestellt wird [Bowman u. a., 2005]. Die wichtigsten Elemente einer Virtuellen Umgebung sind:

- eine virtuelle Welt
- Immersion
- sensorisches Feedback
- Interaktivität

Eine virtuelle Welt ist ein imaginärer Raum, der durch ein Medium manifestiert wird. Diese virtuelle Welt ist die Beschreibung einer Sammlung von Objekten sowie der Gesetzmäßigkeiten und Verhältnisse, welche diese Objekte steuern. Die Immersion beschreibt das Eintauchen in eine virtuelle Welt. Im Gegensatz zur passiven, mentalen Immersion, welche einen emotionalen Zustand beschreibt, spricht man in der Virtuellen Realität von der physikalischen Immersion. Hier betritt der Benutzer physisch das System; dieses wiederum reagiert auf seine Eingaben mit sensorischen, meist visuellen Antworten. Das visuelle Feedback ist immer relativ zur Position des Benutzers, was im Umkehrschluss bedeutet, dass das System Informationen über die Lage des Anwenders benötigt. Die Interaktion des Benutzers mit der Virtuellen Umgebung ist notwendig, damit diese authentisch wirkt. Dafür

Abbildung 2.1.: Zweiseitige Responsive Workbench™ (Quelle: [Wesche, 2004]), das TwoView-Display (Quelle: [TwoView, 2008]) und das i-Cone™ Panorama-Display (Quelle: [i Cone, 2008])

muss die VR-Applikation auf die Aktionen des Anwenders reagieren. Diese Aktionen beinhalten die Interaktion mit den Objekten in der Virtuellen Umgebung, sowie das Ändern des Blickwinkels, wenn sich der Benutzer bewegt [Sherman u. Craig, 2002].

Die betrachteten Grundelemente definieren den Aufbau für Systeme der Virtuellen Realität. Für die visuelle Darstellung werden Display- oder Projektionssysteme benötigt, die mittels Stereoskopie einen räumlichen Tiefeneindruck vermitteln. Für die Ermittlung von Position und Orientierung des Benutzers und seines Eingabegerätes werden Tracking-Systeme eingesetzt, deren Daten in ein VR-Software-Framework (siehe 3.2) zur Beschreibung und Steuerung der Virtuellen Umgebung, sowie zur Berechnung der 3D-Grafik eingesetzt werden. In den folgenden Abschnitten werden diese Elemente betrachtet.

2.1.1. Displays

Display-Systeme für VR-Applikationen müssen für die Erzeugung einer stereoskopischen Darstellung jeweils ein Bild pro Auge erzeugen. Die Bilder müssen zusätzlich getrennt werden, so dass für jedes Auge nur das zugehörige Bild zu sehen ist. Stereosehen wird durch die Parallaxe zwischen den beiden Bildern erreicht. Die Parallaxe ist die scheinbare Positionsänderung eines beobachteten Objektes, das von unterschiedlichen Blickpunkten betrachtet wird [Sherman u. Craig, 2002].

Um zwei Bilder für den Betrachter zu separieren, benötigt man zusätzliche Hardware, wie eine stereofähige Grafikkarte und eine Stereo-Brille. Das Display muss mindestens eine Bildwiederholrate von 100Hz oder besser liefern, da diese halbiert wird, wenn für jedes Auge ein Bild produziert werden soll. Die Separation der Bilder beim Benutzer wird durch eine aktive Stereo-Brille, auch Shutter-Brille (siehe auch 3.8) genannt, erreicht, welche mit der Bildwiederholrate des Displays synchronisiert wird. Die Bilder für das linke bzw. rechte Auge alternieren in einer hohen Frequenz auf dem Display; entsprechend blockt die Shutter-Brille das Bild

für das rechte bzw. linke Auge.

Bei passiven Stereo-Brillen werden Polarisationsfilter eingesetzt. Diese nutzen für die Kanaltrennung die Eigenschaften von linear oder zirkular polarisiertem Licht, wofür spezielle polarisationserhaltende Leinwände bzw. Projektionsflächen benötigt werden [Bowman u. a., 2005].

Bei einem *Head-Mounted Display* (HMD)(siehe Abbildung 2.2) geschieht die Trennung der stereoskopischen Bilder durch Verwendung zweier getrennter *LCD-Panels*, die in einem Helm oder in einer Brille integriert sind und die für das jeweilige Auge das entsprechende Bild anzeigen. Somit wird der Betrachter von seiner Außenwelt komplett abgeschirmt und kann seinen eigenen Körper in der Szene nicht wahrnehmen [Sherman u. Craig, 2002]. Projektionsbasierte Displays wie die *Responsive Workbench*™ [Krüger u. a., 1995] oder die *Cave*™ [Cruz-Neira u. a., 1993] benötigen nicht zwei Displays, um für jedes Auge ein Bild zu produzieren. Hier werden über passive oder aktive Verfahren die Bilder für jedes Auge getrennt. Bei passiven Verfahren verwendet man spezielle Filtertechniken, die sowohl an den Projektoren, wie auch an den Stereo-Brillen eingesetzt werden. Das Verfahren für die Shutter-Brille wurde im letzten Abschnitt erklärt.

Um die Objekte in einer Szene für den Benutzer korrekt darzustellen, muss mittels *Tracking* (siehe 2.1.2) die Position des Betrachters und seine Blickrichtung erfasst werden. Die meisten Displays können nur ein Stereobild produzieren, d.h. die Szene nur für einen Betrachter korrekt darstellen. Alle anderen Benutzer sehen ein verzerrtes Bild. Das *Two-View-Display* [TwoView, 2008] kann über eine Kombination von aktiven Shutter-Brillen und Polarisationsfiltertechnik an den Projektoren zwei stereoskopische Bilder produzieren, so dass die Szene für zwei Benutzer korrekt angezeigt wird (siehe Abbildung 2.1). Hier werden zwei Projektoren eingesetzt, die jeweils ein Aktiv-Stereo-Bild erzeugen. Die Ansichten der zwei Betrachter werden mit zirkularen Polarisationsfiltern vor den Projektoren und an den Shutter-Brillen getrennt.

Die panoramaförmige *i-Cone*™ [Simon u. Göbel, 2002] ist ein Gruppen-Display und verzichtet auf die Darstellung unterschiedlicher Perspektiven für jeden Betrachter. Die Szene wird für einen fiktiven Nutzer berechnet, der im Zentrum des zylindrischen Displays steht (siehe Abbildung 2.1). Dadurch wird nur für diese Perspektive die Szene korrekt dargestellt. Für weit entfernte Objekte ist diese Verzerrung kaum wahrnehmbar, jedoch sind diese umso stärker, je näher das Objekt am Betrachter ist. Interaktive Elemente, wie der Projektionsstrahl (siehe 2.1.4) am Eingabegerät der Benutzer, werden dadurch falsch dargestellt. Dieses Problem wird durch *Multi-Viewpoint Images* [Simon u. Scholz, 2005] gelöst. Hierzu werden unterschiedliche Bildelemente von mehreren Blickpunkten, welche mit den Betrachtern korrespondieren, projiziert und zu einem einzelnen Bild kombiniert. Auf diese Weise werden die Interaktionselemente für jeden Betrachter korrekt dar-

Abbildung 2.2.: (v.l.n.r.) Wii Remote mit A.R.T.-Target, Kabelloser Stylus mit A.R.T.-Target, Benutzer mit Datenhandschuhe und HMD-Display (entnommen aus [Bowman u. Wingrave, 2001])

gestellt.

2.1.2. Tracking

In vielen VR-Applikationen ist es für die Schnittstelle wichtig, Informationen über die Position des Benutzers oder eines Eingabegerätes im 3D Raum zu liefern. Man denke an die Situation, in welcher die Applikation die Orientierung des Kopfes eines Benutzers kennen muss, um die Ansicht der Szene korrekt darzustellen. Oder man denke an Eingabegeräte, deren Position und Orientierung unabdingbar sind, um eine Korrespondenz zur virtuellen Welt zu liefern [Bowman u. a., 2005]. Hierzu werden Tracking-Systeme eingesetzt, von welchen elektromagnetische und optische die gebräuchlichsten in Virtuellen Umgebungen sind. Erstere verwenden einen Sender, um durch drei orthogonale Spulen ein magnetisches Feld zu erzeugen. Diese wiederum generieren Strom in drei weiteren orthogonalen Spulen, welche in einem kleineren Empfänger eingebaut sind und vom Benutzer getragen werden. Das Signal in jeder Spule des Empfängers wird gemessen, um seine Position relativ zum Sender zu berechnen. Die Position und Orientierung des Senders ist bekannt, so dass die absolute Position des Empfängers berechnet werden kann. Optische Tracking-Systeme (z.B. [A.R.T., 2008]) hingegen benutzen visuelle Informationen, um die Position und Orientierung des Benutzers zu bestimmen. Dazu werden fixe Kameras benutzt, die das beobachtende Objekt aufnehmen. Über Bildverarbeitungsalgorithmen werden die Koordinaten von fest konfigurierten Markern aus unterschiedlichen Bildern, die von Kameras aufgenommen wurden, berechnet. Häufig verwendet man für die Marker oder Targets retroflektive Materialien, die mit Hilfe von Infrarotkameras erkannt werden [Sherman u. Craig, 2002].

2.1.3. Eingabegeräte für VR-Systeme

Eingabegeräte sind wie Tracking-Systeme eine Schnittstelle zwischen dem Benutzer und der Virtuellen Umgebung. Diese Geräte werden oft für spezielle Anwendungen extra angefertigt. Auch hier gibt es Geräte mit generischer Funktion für unterschiedliche Applikationen [Sherman u. Craig, 2002].

Eine der wichtigsten Eigenschaften für 3D-Eingabegeräte sind die Freiheitsgrade (degrees of freedom oder DOF). Die Freiheitsgrade beschreiben die unabhängigen Bewegungen eines Eingabegerätes in verschiedenen Dimensionen. Bei 6 DOF haben wir z.B. 3 Werte für die Position in den Achsen x, y, z (horizontal, vertikal, Tiefe) und 3 Werte für die Orientierung bezüglich dieser Achsen [Zhai u. Milgram, 1998].

Ein Eingabegerät generiert Daten diskret oder kontinuierlich. Bei einem diskreten Ansatz wird z.B. ein boolescher Wert übertragen, um in einen bestimmten Modus zu gelangen. Man denke dabei an eine Modellierapplikation, bei der man vom Zeichnenmodus in den Modelliermodus wechseln will. Will man aber eine Positionsänderung in Echtzeit aufnehmen, benötigt man eine Komponente, die kontinuierlich Datenwerte generiert und somit auf jede Bewegung des Nutzers reagieren kann [Bowman u. a., 2005].

Positions- oder Bewegungsdaten können entweder relativ oder absolut sein. Absolute Daten werden immer relativ zu einem Bezugspunkt im Raum geliefert. Bei relativer Datenübermittlung wird eine Auslenkung gegenüber einer Nulllage des Eingabegerätes geliefert. Eine konstante Auslenkung über einen größeren Zeitraum ergibt so z.B. eine größere Verschiebung. Geräte, die diese Übermittlungsart nutzen, reagieren direkt auf die Eingabe des Benutzers, ohne dass sie zusätzlich eine Position zum Benutzer einnehmen müssen [Bowman u. a., 2005].

Man unterscheidet ebenfalls zwischen aktiven und passiven Eingabegeräten. Ein aktives Eingabegerät benötigt einen Impuls um eine bestimmte Aktion auszulösen, unabhängig davon, ob es dann kontinuierlich oder diskret Daten generiert. Von passiven Eingabegeräten spricht man, wenn Eingabegeräte kontinuierlich Daten erzeugen, ohne dass sie auf irgendeine Weise aktiviert werden müssen. Ein Beispiel hierfür sind Tracking-Systeme, welche stetig Werte generieren, um eine mögliche Bewegung aufzunehmen [Bowman u. a., 2005].

Datenhandschuhe (siehe Abbildung 2.2) sind passive Eingabegeräte und werden dazu benutzt, die Haltung der Hand und bestimmte Gestiken zu erfassen. Das Gerät kann z.B. zwischen einer Faust, einer deiktischen (von griech. Deixeis, „zeigen“) Funktion und einer offenen Hand unterscheiden. Die Rohdaten der Datenhandschuhe werden in Form der Gelenkwinkel angegeben; die Software bestimmt dadurch die Stellung der Hand. Datenhandschuhe beinhalten zusätzlich einen Bewegungssensor, da sie zur Gestenerkennung benutzt werden [Bowman u. a., 2005].

Die „Wii Remote“ (oder „Wiimote“, siehe Abbildung 2.2) ist von Nintendo als primäres Steuerungsgerät für die Wii-Spielkonsole entwickelt worden, kann aber

über eine Bluetooth-Schnittstelle mit jedem PC verbunden werden und ist somit kabellos. Ein Hauptmerkmal der Wii Remote ist die Fähigkeit, die Position relativ zu einer Infrarotleiste, die von einer am Monitor befestigten *Sensor-Bar* ausgestrahlt wird, berechnen zu können. Zusätzlich hat die Wiimote einen Beschleunigungssensor und einen Lautsprecher integriert und bietet eine Vibrationsfunktion als haptisches Feedback. Durch diese Eigenschaften lässt sich die Wii Remote gut als Interaktionsgerät für VR-Applikationen einsetzen [Schou u. Gardner, 2007]. Will man absolute Daten erhalten, gibt es die Möglichkeit, die Wiimote zu tracken. Dazu werden retroflektive Marker an das Eingabegerät angebracht, so dass die Wiimote von Infrarotkameras getrackt werden kann. Dadurch können absolute Werte der Position und Orientierung geliefert werden.

2.1.4. Interaktionstechniken

Die Mensch-Computer Interaktion beschreibt den Prozess der Kommunikation zwischen Benutzer und Computer. Das Medium, durch das diese Kommunikation stattfindet, ist eine Benutzerschnittstelle. Diese übersetzt die Aktionen oder Eingaben des Benutzers in eine Form, die der Computer (oder eine Applikation) versteht und ihn dazu veranlasst, darauf zu reagieren. Die Ausgabe des Computers wird für den Benutzer wiederum übersetzt, so dass er darauf reagieren kann. Um bestimmte Aktionen über die Benutzerschnittstelle auszuführen werden Interaktionstechniken eingesetzt, die sowohl Software- als auch Hardware-Komponenten beinhalten [Bowman u. a., 2005].

[Mine, 1995] präsentiert 5 fundamentale Interaktionsformen:

- Navigation
- Selektion
- Manipulation
- Skalierung
- Systemsteuerung

Navigation beschreibt die Aktion, die den Benutzer von seiner momentanen Position zu einem neuen Ziel oder in eine gewünschte Richtung bewegt. Die Selektion betrifft das Auswählen bestimmter Objekte und erfordert einen Mechanismus für die Identifikation des selektierten Objektes sowie ein Signal, das die Selektion anzeigt. Bei der Manipulation wird die Position und Orientierung eines Objektes in der Virtuellen Umgebung bestimmt. Für das Objekt werden dafür drei Parameter benötigt: die Änderung der Position, die Änderung der Orientierung und das Rotationszentrum. Die Skalierung ermöglicht es, das ausgewählte Objekt vergrößern

oder verkleinern zu können. Als Parameter werden hierfür das Skalierungszentrum und der Skalierungsfaktor benötigt [Mine, 1995].

Um die ersten vier Interaktionsformen auszuführen, gibt es eine Anzahl an Interaktionstechniken, die abhängig vom Eingabegerät (siehe 2.1.3) und dem Tracking-System (siehe 2.1.2) sind. Ray-Casting Techniken benutzen eine Art Laserstrahl, der das zu selektierende Objekt in der Virtuellen Umgebung schneidet. Die Richtung des Strahls wird durch ein Eingabegerät (6 DOF), das der Benutzer hält, gegeben. Der Vorteil bei der Ray-Casting-Technik ist, dass Objekte gegriffen werden können, die nicht in unmittelbarer Nähe des Benutzers sind. Die Auswahl des Objektes geschieht dabei durch simples Zeigen auf dieses [Mine, 1995].

Eine relevante Interaktionstechnik für diese Arbeit ist die *Scaled-Grab-Technik* [Simon u. Dressler, 2005]. Bei dieser Technik geht es darum, mit Objekten unabhängig von ihrer Entfernung zum Benutzer effektiv interagieren zu können. Dies geschieht durch Kombinieren von

- der Selektion durch Ray-Casting,
- direkter Übertragung der Rotation des Eingabegerätes auf das manipulierte Objekt
- und beschleunigter Translation

Durch die Scaled-Grab-Technik wird die Reichweite des Armes skaliert, so dass Objekte unabhängig davon, wie weit sie vom Benutzer positioniert sind, mit einer Bewegung herangezogen werden können. Dabei wird der Abstand vom Schnittpunkt des Objektes zum Benutzer berechnet und somit wird der Skalierungsfaktor für die Armbewegung bestimmt [Simon u. Dressler, 2005].

2.2. Systemsteuerung und Interaktionstechniken in Virtuellen Umgebungen

Im Gegensatz zur direkten Interaktion in einer Virtuellen Umgebung mittels Navigation, Selektion, Manipulation und Skalierung, kann der Benutzer bei der Systemsteuerung Arbeitsabläufe durchführen, die über direkte Interaktionen nur schwer zu realisieren sind. Als Systemsteuerung werden nach [Bowman u. a., 2005] folgende Aktionen definiert:

- Der Befehl an das System, eine bestimmte Funktion auszuführen
- Die Änderung des Interaktionsmodus
- Die Änderung des Systemstatus

Bei der ersten Aktion gibt der Benutzer einen Befehl an die Applikation, eine bestimmte Funktion aufzurufen, wie z.B. das Speichern einer Szene oder das Erzeugen eines Objektes. Die zweite Aktion beschreibt die Änderung des Interaktionsmodus wie z.B. das Wechseln in den Zeichnenmodus (z.B. bei einem Bildbearbeitungsprogramm). Hierbei wird nichts am System selbst geändert, sondern nur an der weiteren Interaktion. Der dritte Punkt beschreibt Aktionen, wie das Aktivieren eines bestimmten Objektes. Dadurch ändert sich der Systemstatus und alle weiteren Aktionen beziehen sich dann auf genau dieses Objekt [Bowman u. a., 2005].

Eine Systemsteuerung wird in 2D-Desktop-Umgebungen (z.B. 2D-Applikationen auf einem Desktop-Computer) über Menüs oder textbasierte Kommandozeilen realisiert. In immersiven Umgebungen sind diese Techniken zur Systemsteuerung aber nicht sehr effektiv, denn die 6-DOF-Eingabe unterscheidet sich enorm von den 2 DOFs auf einem Desktop-Computer. Aus diesem Grund unterscheiden sich Eingabegeräte, die in Virtuellen Umgebungen benutzt werden, von der Tastatur und Maus am Desktop [Bowman u. a., 2005].

Im Folgenden werden einige Interaktionstechniken vorgestellt, die zur Systemsteuerung in Virtuellen Umgebungen eingesetzt werden. Als erstes wird die Steuerung über Menüs und anschließend die Steuerung mit natürlichen Schnittstellen wie Gestik und Spracheingabe behandelt.

2.2.1. Menüs

Menüs in VR-Umgebungen funktionieren in der gleichen Art und Weise wie in 2D-Desktopumgebungen. Unter Menü versteht man eine Liste von Einträgen zur Systemsteuerung, zur Aktivierung eines hierarchisch untergeordneten Menüs oder zum Öffnen einer Dialog-Instanz mit weiteren Eingabemöglichkeiten. Die hierarchische Ordnung wird genutzt, wenn die Anzahl der Menüeinträge einen annehmbaren Umfang des Menüs übersteigt und dieses somit unübersichtlich wird. In diesem Fall können thematisch oder funktional ähnliche Einträge in einem Untermenü zusammengefasst werden. Das Öffnen einer Dialog-Instanz gibt dem Benutzer die Möglichkeit weiterer Eingaben oder Einstellungen. Auch der Abbruch der zuletzt gewählten Funktion ist möglich. Das Auswählen der Menüeinträge kann als eine Kombination aus Manipulation und Selektion betrachtet werden. Ähnlich wie bei der direkten Interaktion mit Objekten werden auch Menüeinträge über Interaktionstechniken ausgewählt und anschließend aktiviert [Dressler, 2007].

3D-Menüs sind einfache Adaptionen aus 2D-Desktop-Umgebungen. Für Virtuelle Umgebungen ist allerdings eine natürlichere Art der Systemsteuerung erwünscht [Latoschik u. a., 1998] und aus diesem Grund werden im Folgenden gestik- und sprachbasierte Interaktionstechniken vorgestellt.

2.2.2. Gestenerkennung

Bei der Gestenerkennung kann eine bestimmte Körperhaltung und Körperbewegung eine Geste darstellen, die von einem Tracking-System (siehe 2.1.2) aufgenommen wird und mittels Algorithmen und mathematischer Methoden als solche erkannt wird. Eine große Bedeutung hat hierbei die Erkennung von Kopf- und Handgesten. Letztere können auch mittels eines Datenhandschuhs mit integrierten Beschleunigungs- oder Positionssensoren aufgenommen werden. Auch ein getracktes Eingabegerät kann zur Gestenerkennung benutzt werden.

In [Crowley u. Coutaz, 1996] werden der Gestik drei unterschiedliche Funktionen zugeteilt:

- Die ***semiotische*** Funktion wird benutzt, um eine Aussage bedeutungsvoll zu unterstreichen, wie das Winken zum Abschied.
- Die ***ergotische*** Funktion der Gestik wird mit dem Begriff der manuellen Arbeit assoziiert, wie das Formen von Objekten aus Ton.
- Die ***epistemische*** Funktion der Gestik erlaubt es einem Menschen, seine Umgebung durch taktile Wahrnehmung zu erkunden, wie das Streichen über ein Material, um dessen Struktur zu erfassen.

Für diese Arbeit ist die semiotische Funktion der Gestik von Interesse und dabei im Besonderen die deiktische Funktion, wie z.B. Zeigeoperationen auf ein Objekt oder eine Umgebung.

[Koons u. Sparrell, 1994] unterscheidet zusätzlich zwischen ikonischer und zeigebasierter Gestik. Von ikonischen Gesten spricht man, wenn die Hand ein Objekt und dessen Bewegung simuliert. Zum Beispiel kann folgender Satz eine entsprechende Geste begleiten: „Das Fahrzeug bewegt sich so“. Unter Berücksichtigung der relativen Position und Orientierung der Hand kann der Benutzer intuitiv das Objekt in die Szene positionieren und orientieren. Anzumerken sei hier, dass ikonische Gestik nur in Verbindung mit Spracheingabe Sinn ergibt. Zeigebasierte Gesten finden wir unter anderem in [Bolt, 1980], indem der Benutzer auf Objekte zeigt, um diese zu referenzieren und anschließend zu einem anderen Punkt in der Szene wechselt, um das entsprechende Objekt zu bewegen.

- „Put that“ - der Benutzer zeigt auf das ausgewählte Objekt
- „there“ - nun wird auf die Stelle gezeigt, an der das Objekt erscheinen soll

2.2.3. Sprachsteuerung

Sprachsteuerung wird hauptsächlich für eine multimodale Interaktion mit anderen Eingabemethoden benutzt (siehe 2.2.4). Abgesehen von einer sehr guten Spracherkennungsqualität sind weitere Faktoren zu beachten, wenn es gilt, ein gutes Interaktionskonzept mit Sprache zu entwickeln. Ein wichtiger Faktor ist das Mikrophon und dessen Platzierung. Idealerweise wird ein Raummikrofon benutzt. Hier taucht allerdings das Problem auf, dass dieses viel mehr aufnimmt als die Befehle des Anwenders. Man denke hier an Maschinengeräusche, Gespräche von Personen die sich im selben Raum befinden oder allgemeine Nebengeräusche. All dies beeinträchtigt die Spracherkennung schwer und sollte daher vermieden werden. Eine geeignete Lösung ist ein kabelloses Headset, das direkt vor dem Mund des Benutzers platziert wird. Auf diese Weise kann der Mikrofonpegel gesenkt werden, was störende Nebengeräusche minimiert [Bowman u. a., 2005].

Während der Interaktion in einer Virtuellen Umgebung ist oftmals der Austausch mit anderen Benutzern erwünscht, ohne dass Phrasen aus der Kommunikation vom System als Befehle erkannt werden. Eine Möglichkeit dieses Problem zu umgehen ist es, eine implizite Spracheingabe nach dem *Push-to-Talk* [Bowman u. a., 2005] Schema zu implementieren. Durch Drücken eines Buttons kann die Spracherkennung für die Dauer der Befehlsaussprache aktiviert werden. Um die Natürlichkeit des Interfaces zu bewahren, ist es sinnvoll, das Drücken des Buttons in die Interaktionsroutine zu integrieren.

Bei der Entwicklung einer Sprachapplikation sollte berücksichtigt werden, dass die Sprachschnittstelle unsichtbar für den Benutzer ist. Das heißt, dass die Person einerseits keinen Überblick über die gültigen Ausdrücke hat, andererseits aber auch nicht wissen kann, ob ein ausgesprochener Befehl erkannt wurde oder nicht. Hier kann visuelles oder auditives Feedback eingesetzt werden, so dass der Anwender über den Status des Systems informiert ist. Ebenfalls muss berücksichtigt werden, dass Spracheingabe nicht für Systeme sinnvoll ist, bei denen man wenig oder gar keine Zeit für das Erlernen der Sprachbefehle hat [Bowman u. a., 2005].

Im folgenden Abschnitt werden nun einige Implementierungen vorgestellt, die Spracheingabe und Gestik als Eingabemethoden benutzen und die als Grundlage für das hier entwickelte Interaktionskonzept dienen.

2.2.4. Multimodale Interaktion mit Spracheingabe und zeigebasierter Gestik

Wir haben bisher zwei Werkzeuge betrachtet (Gestik und Spracheingabe), die ein natürliches Interface ermöglichen. Des Weiteren gibt es weitere Techniken, die den menschlichen Körper als Eingabegerät benutzen, wie z.B. bioelektrischen Input, wobei Signale von Muskelnerven aufgenommen werden, die wiederum eine Funk-

tion auslösen. Auch Brain Input soll nicht unerwähnt bleiben. Hier werden vom Gehirn generierte Signale direkt als Input für ein System aufgenommen [Bowman u. a., 2005]. Da diese Eingabemethoden nicht weiter relevant für diese Arbeit sind, beschränken wir uns auf zeigebasierte Gestik und Spracheingabe.

Multimodale Systeme verarbeiten kombinierte Eingabemethoden wie Sprache, Gestik, Kopf- und Körperbewegung, haptischen Input und Input aus einem Eingabegerät. Diese Systeme präsentieren eine neue Richtung für die Modellierung und Umsetzung von Interaktionskonzepten, denn gestikbasierte Interaktionstechniken werden leistungsstärker und natürlicher, wenn sie mit Spracheingabe ergänzt werden [Latoschik u. a., 1998]. Richtig modellierte, multimodale Systeme integrieren sich ergänzende Modalitäten und liefern eine synergistische Mischung, in welcher die Stärken jedes Modus ausgeschöpft werden. Das dient dazu, dass der eine Modus die Schwächen des anderen ausgleichen kann [Oviatt, 1999].

[Hauptmann u. McAvinney, 1993] präsentiert eine Studie, bei der Benutzer Gesten und Sprache intuitiv einsetzen, um mit dem Computersystem zu kommunizieren. Dabei bekamen die Probanden Aufgaben gestellt, welche mit Gestik, Sprache oder einer Kombination aus beiden auszuführen waren. 70% der Aufgaben wurden mit der kombinierten Interaktion, während 13% mit Gesten und 16% mit Spracheingabe bewältigt wurden. Das spricht für eine multimodale Interaktion. Es bleibt zu untersuchen, ob dies auch für eine abstrakte Funktionalität gilt.

In [Thorisson u. a., 1992] wurde die natürliche Kommunikation zwischen Menschen beobachtet, in welcher Gestik, Intonation, Gesichtsausdruck und Blickrichtung als Kontext für die Sprache eingesetzt wird. Darauf aufbauend wurde ein Interaktionskonzept entwickelt, welches dem Benutzer erlaubt, in Echtzeit mit einem Graphik-Display durch Zeigeoperationen, Blickrichtung und Sprache zu interagieren. Das System reagiert entweder durch graphische Manipulation oder gesprochenen Antworten darauf.

Das „Put that there“-System von Bolt [Bolt, 1980] besteht aus einem Raum, in dem sich ein Display mit Rückprojektion befindet. Der Benutzer sitzt in der Mitte des Raumes und trägt magnetische Sensoren an den Handgelenken, so dass die Handposition über magnetisches Tracking berechnet werden kann. Der Benutzer kann nun Sprache, Gestik oder eine Kombination aus beidem verwenden, um Objekte auf dem Display hinzuzufügen, zu löschen oder zu bewegen. Die Mächtigkeit dieses Interaktionkonzeptes liegt darin, dass der Anwender spontan und auf eine natürliche Art und Weise mit Objekten, die er sieht, interagieren kann.

[Latoschik u. a., 1998] hat ein System entwickelt, in der eine virtuelle Montageanlage von einem Benutzer durch Sprache und Zeigeoperation manipuliert wird. Dabei wird die deiktische Funktion der Gestik zur Selektion von Objekten angewendet und ikonische Gestik, um diese Elemente zu manipulieren. Ziel ist es, den Benutzer aufzufordern, auf eine natürliche Weise mit dem System zu interagieren.

[Neal u. Shapiro, 1991] präsentiert ein intelligentes Interaktionsmodell, welches Sprache, Grafiken und Zeigeoperationen kombiniert, um eine natürliche Mensch-Maschine-Interaktion zu ermöglichen. Der Antrieb hier ist, die kognitive Belastung während der Interaktion mit dem System zu reduzieren. In Mark Billinghurst's Veröffentlichung „Put That Where?“ [Billinghurst, 1998] wird ein Interaktionskonzept vorgestellt, bei dem über Sprache Objekte in eine Szene gesetzt werden und es dann über Gesten festgelegt wird, wie diese Objekte relativ zueinander im Raum aufgestellt werden sollen. Billinghurst empfiehlt, dass Sprache zur Systemkontrolle und Gestik zur räumlichen Eingabe, wie Zeigeoperationen, eingesetzt werden soll.

Für eine Simulation von Gefäßrekonstruktion wird in [Zudilova, 2002] ein Interaktionskonzept vorgestellt, welches kontextabhängige Sprachsteuerung, Gesten und direkte Manipulation von 3D Objekten integriert. Für die Objektselektion und die Menünavigation wird Ray-Casting verwendet. Der Benutzer kann auf diese Weise Funktionen über das Menü oder über die Sprache absetzen. Zusätzlich wird in diesem System berücksichtigt, dass Anwender (in diesem Fall Radiologen, Chirurgen und Medizinstudenten) unterschiedliches Hintergrundwissen und unterschiedliche Erfahrung mit VR-Systemen besitzen. Aus diesem Grund ist eine adaptive Benutzerschnittstelle geschaffen worden, welche sowohl Experten als auch Anfängern gerecht wird. Dies wird erreicht, indem der Benutzer die Möglichkeit hat, sich im Voraus für eine Interaktionsmethode (direkte Manipulation oder Sprachsteuerung) zu entscheiden, dynamisch den Inhalt der Menüeinträge bestimmen kann oder eine individuelle Grammatik bestimmen kann.

[Malkewitz, 1998] ersetzt die Maus- und Tastaturfunktionen durch Zeigeoperationen mit dem Kopf bzw. mit Spracheingabe. Dieses Interaktionskonzept soll körperbehinderten Benutzern helfen, mit existierenden 2D-Desktop-Applikationen zu interagieren.

[Weimer u. Ganapathy, 1989] fand heraus, dass sich Gestenoperationen und Spracheingabe komplementieren und somit eine sehr mächtige Interaktionsschnittstelle schaffen.

Bisher wurde gezeigt, dass multimodale Interaktionstechniken eingesetzt werden, wenn es darum geht, eine Schnittstelle in virtuellen Umgebungen natürlicher und intuitiver zu gestalten. Um eine Sprachschnittstelle zu implementieren wird ein Spracherkennungssystem benötigt. Im Folgenden werden die Anforderungen an ein geeignetes Spracherkennungssystem definiert. Anschließend wird eine Auswahl solcher Systeme vorgestellt.

2.3. Spracherkennungssoftware

Während in früheren wissenschaftlichen Arbeiten über Spracherkennung die Spracherkennungssoftware oftmals in Eigenregie entwickelt worden ist, gibt es heute eine Reihe von kommerziellen Lösungen, welche den hohen Qualitätsansprüchen für eine fehlerarme Spracherkennung genügen. Die meisten Spracherkennungssysteme auf dem Markt sind zum Diktieren gedacht und eignen sich nicht gut, um eigene Applikationen zu entwickeln. Um unseren Anforderungen zu genügen, sollte ein Spracherkennungssystem

- **sprecherunabhängig** sein
- eine umfangreiche **Entwicklerumgebung** (SDK, Software Development Kit) bieten
- eine **Demoversion** liefern
- **kostengünstig** sein

Als nächstes wird eine Auswahl an Spracherkennungssystemen vorgestellt, die mit den oben aufgeführten Anforderungen verglichen wird.

Novotech GPMSC

Novotech bietet mit **GPMSC (General Purpose Machines' Speech Control)** ein sprecherunabhängiges Sprachsteuerungssystem, das hauptsächlich für die Maschinensteuerung gedacht ist und lediglich auf Microsoft Windows-Systeme läuft. Es bietet eine Demoversion an - das gesamte Paket aber beinhaltet keine Entwicklungsumgebung und funktioniert praktisch wie eine Black Box [Novotech, 2008].

COM VISION GENIE

COM VISION bietet mit seinem Produkt **GENIE** ein intelligentes Verfahren, das aus verschiedenen Software-Komponenten und Protokollen besteht. Die drei wesentlichen Bereiche sind Client, Server und Brainpack. Der Client ist das Endgerät, das mit Sprache bedient werden soll. Es können mehrere solcher Clients angebunden werden; selbst ein Client kann mehrere Clients beinhalten. Der Server kommuniziert mit den Clients, kennt deren aktuellen Status und kann diese auch ändern. Der Server enthält die eigentliche Spracherkennungssoftware. Das Brainpack beinhaltet die Intelligenz und die Steuerlogik der zu bedienenden Geräte. Es kennt die jeweiligen Eigenschaften der Clients und weiß, welche Aktionen aufgrund bestimmter Kommandos auszuführen sind. Der Server greift auf das Brainpack zu,

dieser steuert dementsprechend die in den Clients enthaltenden Informationen. Die unterstützten Betriebssysteme sind Microsoft Windows 2000, Microsoft Windows XP, Linux, FreeBSD, OpenBSD, NetBSD, Solaris Intel und Solaris Sun. Eine Java Version ist ebenso erhältlich, die auf allen Systemen arbeitet, auf denen Java 1.4.2 oder höher verfügbar ist. **COM VISION** bietet für **GENIE** zwar eine SDK, aber keine Demoversion [Comvision, 2008].

Philips Speech SDK 4.2

Philips hat die **Speech SDK 4.2** herausgebracht, welche an die unterschiedlichen Philips Spracherkennungssysteme angebunden werden kann: *Speech to Text recognition engine*, *Command and Control recognition engine*, *Spelling recognition engine* und *Verification recognition engine*. Trotz einer umfangreichen Entwicklungsumgebung und ausgefeilten Features ist das Spracherkennungssystem von **Philips** nicht sprecherunabhängig. Es bietet zwar ein *Quick Training* an, welches innerhalb 2 Minuten die benutzerspezifische Charakteristik der Stimme des Benutzers lernt, ist jedoch für den Gebrauch des hier vorgestellten Interaktionskonzeptes nicht akzeptabel. Neben Microsoft Windows werden auch die Linux und Macintosh Plattform unterstützt [Philips, 2008].

Microsoft Speech Recognizer 8.0 für Windows

Für die Implementierung des VRGeo Speech Control, der Sprachapplikation, die für diese Diplomarbeit entwickelt worden ist, wurde Microsoft Vistas Spracherkennungssystem benutzt. Microsoft bietet dafür:

- eine mächtige SDK,
- ein sehr gute Erkennungsqualität,
- ein sprecherunabhängiges System
- und eine kostengünstige Lösung, da das Spracherkennungssystem ab Vistas Home Basic Version zur Verfügung steht.

Die Qualität des Audiosignals spielt eine wichtige Rolle für die Spracherkennung. Im nächsten Abschnitt werden die Anforderungen an Mikrofone definiert, unter Berücksichtigung der besonderen Ansprüche, die eine VR-Umgebung stellt. Schließlich wird eine Auswahl an Mikrofonen präsentiert.

2.4. Mikrofone

Interagiert ein Benutzer in einer Virtuellen Umgebung sind seine Hände meistens mit mindestens einem Eingabegerät belegt. Aus diesem Grund ist der Einsatz eines Handmikrofons für die Spracherkennung ungeeignet. Zusätzlich sollte die zügige Anbringung des Mikrofons möglich sein und keine langwierige Vorbereitung benötigen. Hierzu bietet sich ein Headset oder ein Ansteckmikrofon besonders an. Ein weiterer Störfaktor, der beachtet werden muss, sind Umgebungsgeräusche, welche die Spracherkennung behindern können. Diesen Anforderungen entsprechend werden nun einige handelsübliche Mikrofonsysteme und deren wichtigsten Eigenschaften präsentiert.

AKG WMS 40 PRO Presenter Set

Das AKG Ansteckmikrofon ist ein kabelloses UHF (Ultra-High-Frequency) System. Es benötigt einen Taschensender und einen Empfänger, der an einem Verstärker oder an einem Mischpult angeschlossen wird. Alle UHF Kondensatormikrofone bieten eine hohe Audioqualität und sind deswegen optimal für Spracherkennungssysteme geeignet. Der Nachteil hierbei ist, dass zusätzliche Hardware (Mischpult, Verstärker) benötigt wird, um mit dem PC verbunden zu werden [AKG, 2008].

XoVox XCommunicator 6 PLUS

Das XCommunicator ist ein Bluetooth-Headset, das dem neuen Bluetooth 2.0 Enhanced DATA rate (EDR) Standard folgt. Das Headset wird mittels eines USB-Adapters an der USB-Schnittstelle mit dem PC verbunden. Dieses verfügt über eine Reichweite bis 20 Meter. Die Sprachqualität ist nicht ganz so hoch wie bei einem UHF-Mikrofon, jedoch werden Spracherkennungs-Systeme wie Dragon NaturallySpeaking unterstützt [xovox, 2008].

Plantronics CS60 USB DECT

Das Plantronics CS60 USB ist ebenfalls ein kabelloses Headset-System und folgt dem DECT-Standard (Digital Enhanced Cordless Telecommunications). DECT ist ein Standard für Schnurlostelefone sowie für die kabellose Datenübertragung im Allgemeinen [DECT, 2008]. Die Empfangsstation wird an der USB-Schnittstelle am PC angeschlossen. Auch dieses Headset ist wie das XCommunicator für Spracherkennung optimiert und hat allerdings eine größere Reichweite bis 100 Meter. Eine weitere positive Eigenschaft ist die integrierte Rauschunterdrückungsfunktion [Plantronics, 2008].

Aufgrund der genannten Qualitäten wurde das Plantronics CS60 USB DECT für die Implementierung des in dieser Arbeit vorgestellten Interaktionskonzeptes eingesetzt.

2.5. Zusammenfassung des Kapitels

In diesem Kapitel wurden die Elemente Virtueller Umgebungen definiert sowie deren Benutzerschnittstellen und Interaktionstechniken vorgestellt. Ebenfalls sind verschiedene Eingabegeräte, die diese Interaktionstechniken implementieren, präsentiert worden. Es wurde gezeigt, dass bei der Entwicklung natürlicher Benutzerschnittstellen ein Interaktionskonzept durch Spracheingabe leistungsfähiger gemacht werden kann. In dem Abschnitt über multimodale Interaktion wurde durch vorangegangene Arbeiten gezeigt, dass die Kombination aus zeigebasierten Gesten und Spracheingabe als eine sehr natürliche Interaktionsmethode akzeptiert wird. Mit diesem Wissen soll nun ein natürliches Interaktionskonzept entwickelt und anschließend in einer Benutzerstudie evaluiert werden. Im nächsten Kapitel wird vorerst auf die Entwicklungsumgebung des Systems eingegangen.

3. Entwicklungsumgebung

Im vorangegangenen Kapitel wurde auf Interaktionstechniken zur Systemkontrolle in Virtuellen Umgebungen eingegangen. Es wurden die Anforderungen für konventionelle Methoden zur Systemsteuerung mit Menüs sowie multimodale, natürliche Schnittstellen mit Gestik und Spracheingabe definiert. Im Folgenden wird die Entwicklungsumgebung für das hier vorgestellte Interaktionskonzept vorgestellt. Dazu gehören das Spracherkennungssystem von Microsoft Vista, das VR-Framework Avango^{TM}, das VR-System VRGeo Demonstrator und das TwoView Display-System. Die Elemente dieser Komponenten werden in diesem Kapitel ausführlich beschrieben.

3.1. Microsoft Speech Recognition Engine (SAPI 5.3)

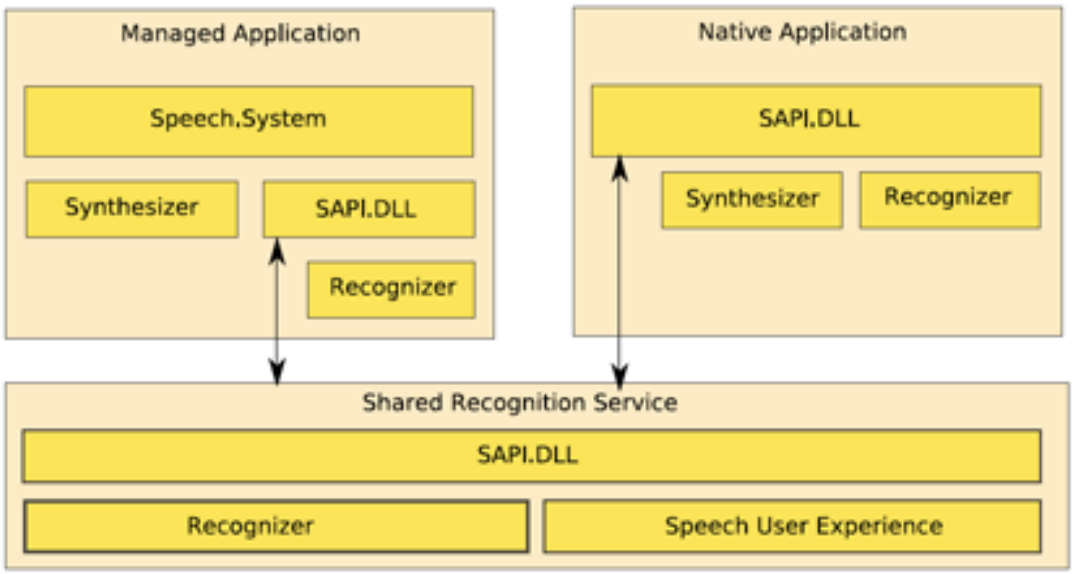


Abbildung 3.1.: *Aufbau der Speech API 5.3 in Windows Vista*

Als Entwicklungsumgebung für den Sprachserver wurde Microsoft Visual Studio mit C# als Programmiersprache gewählt. Für die Speech Recognition Engine in Windows Vista gibt es zwei Sprachschnittstellen:

- SAPI 5.3
- System.Speech.Recognition-Namespace im .NET Framework 3

Abbildung 3.1 zeigt die Beziehung der einzelnen Schnittstellen zu den Applikationen und dem darunterliegenden Spracherkennungssystem (*Recognizer*). Die Hauptkomponenten sind der *Synthesizer* und der *Recognizer*. Der Synthesizer erhält Text als Eingabe und produziert einen Audio-Stream als Ausgabe. Der Recognizer hingegen erhält einen Audio-Stream als Eingabeparameter und wandelt diesen folglich in Text um. Der Fokus hierbei liegt auf dem Recognizer, auf welchen über die SAPI 5.3 zugegriffen werden kann. Die Klassen im System.Speech.Recognition-Namespace sind als Wrapper für die SAPI-Funktionalität zu verstehen. Ein wichtiges Merkmal ist, dass beide Schnittstellen die SAPI Device Driver Interface (DDI) implementieren. Diese Schnittstelle macht das darunterliegende Spracherkennungssystem austauschbar. Das bedeutet, dass Entwickler, welche die SAPI 5.3 oder System.Speech benutzen, jedes andere Spracherkennungssystem heranziehen können, welches die SAPI DDI implementiert.

In Abbildung 3.1 ist zu sehen, dass der Synthesizer stets im gleichen Prozess wie dem der Applikation instanziiert wird, während der Recognizer auch in einem anderen Prozess, genannt SAPISVR.EXE instanziiert werden kann. Diese Tatsache liefert ein verteiltes Spracherkennungssystem, welches simultan von mehreren Applikationen genutzt werden kann, was von Vorteil ist, da der Recognizer große Laufzeitressourcen in Anspruch nimmt und somit der entsprechende Overhead reduziert werden kann [Microsoft, 2008].

3.2. Avango

Avango [Tramberend, 1999] ist ein objektorientiertes VR-Framework, das die Technologie zur Entwicklung verteilter und interaktiver VR-Applikationen bereitstellt. Hauptkomponenten sind die auf OpenGL aufbauende Szenengraph-Schnittstelle *Performer*, ein an *Inventor* angelehnter *Field*-Mechanismus, sowie ein auf der funktionalen Programmiersprache *Scheme* aufbauendes *Scripting-Interface*. Weitere Komponenten sind eine Netzwerkschicht zur Realisierung verteilter Applikationen, eine Abstraktion für Ausgabegeräte und ein *Device-Deamon* zur Anbindung von Eingabegeräten und Tracking-Systemen.

3.2.1. Fields in Avango

Ein Szenengraph ist eine objektorientierte Datenstruktur, mit der die logische oder in vielen Fällen die räumliche Anordnung der darzustellenden Szene beschrieben wird. In Avango werden die einzelnen Szenengraphknoten als *Nodes* bezeichnet und sind Instanzen von Avango- oder Performer-Typen. Um diese zu kapseln nutzt Avango das Field-Konzept aus Inventor. Nodes dienen als *Field-Container*, welche ihre *Fields* als Schnittstelle zu anderen Nodes oder Applikationen außerhalb anbieten. Fields und Field-Container unterstützen ein *Streaming-Interface* und können Avango-Objekte im Netzwerk verteilen. Auf diese Weise können Applikationen auf entfernten Maschinen in Cluster-Systemen betrieben werden.

Die Fields in Avango repräsentieren einzelne (*Single-Fields*) bzw. eine Liste (*Multi-Fields*) von Werten, die allesamt typgebunden sind. Über `getValue()` und `setValue()` werden die Werte in den Fields gelesen bzw. gesetzt. Entsprechend für die Multi-Fields gibt es die Funktionen `add1Value()` und `remove1Value()`. Unterschiedliche Fields gleichen Typs können über *Field-Connections* verbunden werden. Ändert sich ein Wert in einem *Source-Field*, so wird diese Änderung an alle verbundenen Fields weitergegeben. Somit können zusätzliche logische Verknüpfungen zwischen den Nodes hergestellt werden.

Der Datenfluß in den Field-Connections wird in jedem gerenderten Frame evaluiert. Die Weiterreichung der Änderungen in den Fields geschieht hierarchisch. Beginnend mit den Sensor-Knoten, an denen z.B. Eingabegeräte oder Zeitgeber angebunden sind, werden die Daten an Effektoren wie Avango-Objekte und Nodes übergeben. Letztere repräsentieren die eigentliche Applikation und reagieren auf geänderte Field-Werte, um ihrerseits anderen Objekten die geänderten Werte zur Verfügung zu stellen. Zum Schluss werden noch die Aktuatoren wie z.B. die virtuelle Kamera mit den aktuellen Werten versorgt.

3.2.2. Scripting in Avango

In Avango werden Komponenten und Nodes in C++ implementiert. Die Entwicklung von Applikationen kann weiterhin in C++ oder aber auch in der an *Lisp* angelehnten funktionalen Programmiersprache *Scheme* geschehen. *Avango-Nodes* werden auf diese Weise in *Scheme* instanziiert, in den Szenegraphen integriert und über Field-Connections verbunden. Typischerweise reagiert die Applikation somit über Script-Callbacks auf Field-Änderungen in Sensor- und Effektor-Knoten.

3.3. VRGeo Demonstrator

Der VRGeo Demonstrator ist eine auf Avango basierende, immersive VR-Applikation, die im Kontext der Öl- und Gasindustrie zur Exploration von seismischen

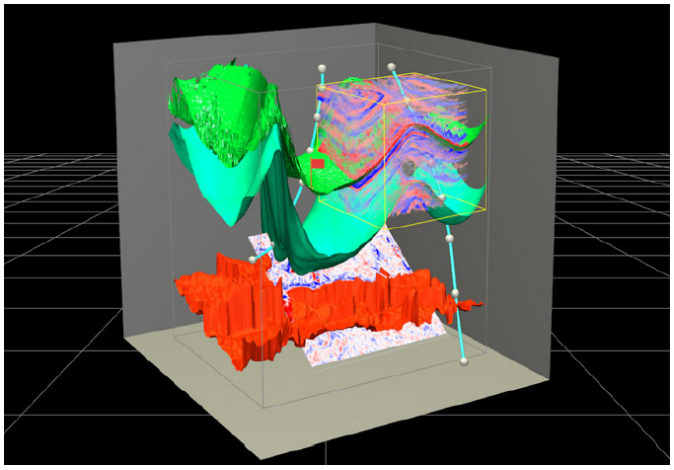


Abbildung 3.2.: *Der VRGeo Demonstrator - Das Bild zeigt den Arbeitsbereich (grauer Quader). Darin enthalten sind Horizons (grüne und rote Erdschichten), eine Schnittebene, eine Volumenlinse, Bohrpfade und eine Markierung in Form einer kleinen Flagge.*

Daten genutzt wird. Regelmäßig findet innerhalb des VRGeo Konsortiums ([VRGeo, 2008]) das VRGeo-Meeting statt, bei dem neue Technologien besprochen, integriert und die vorangegangenen Implementierungen diskutiert werden. Experten aus unterschiedlichen Fachrichtungen, wie z.B. VR, Geologie, Geophysik, Chemie und Softwareentwicklung treffen hier zusammen. Der Demonstrator unterstützt im Moment Multi-User-Interaktion mit Hilfe von OmniStereo und Multi-Viewpoint-Images in der i-Cone^{TM} [Simon u. Göbel, 2002] oder mit Hilfe von Head-Tracking im TwoView, sowie einen semi-immersiven Aufbau mit Interaktion an autostereoskopischen Displays am Desktop. In den folgenden Unterkapiteln wird näher auf die einzelnen Objekte innerhalb des Demonstrators eingegangen und im Besonderen auf die Interaktion mit diesen.

3.4. Die Interaktion im VRGeo Demonstrator

3.4.1. Das Interaktionsgerät

Als Eingabegerät wird im Demonstrator zur Zeit eine getrackte Wii Remote benutzt. Sie ist von Nintendo als primäres Steuerungsgerät für die Wii Konsole entwickelt worden, kann jedoch über eine Bluetooth-Schnittstelle mit jedem PC verbunden werden.

Die Wii Remote (siehe 2.1.3) ist ein kabelloses Eingabegerät und bietet eine Anzahl von Tasten, die individuell mit Funktionen belegt werden können. Zudem ist

Abbildung 3.3.: *Eine präparierte Wii Remote für den Einsatz im VRGeo Demonstrator. Am vorderen Ende sind die retroflektiven Marken zu sehen, die von den Infrarotkameras getrackt werden.*

sie unabhängig von der Nintendo Spielkonsole Wii zu einem kostengünstigen Preis zu erwerben. Diese Eigenschaften ermöglichen einen Einsatz als Eingabegerät im VRGeo Demonstrator. Für die Nutzung am TwoView-Display (siehe 3.8) wurden an ihr retroflektive Marker angebracht wie in Abbildung 3.3 zu sehen ist. So kann die Position und Orientierung der Wiimote von den vier am TwoView-Display angebrachten Infrarotkameras erfasst werden.

3.4.2. Pick-Ray-Interaktion

Neben der Wii Remote findet die Interaktion im VRGeo Demonstrator auch mit weiteren getrackten 6DOF-Eingabegeräten wie Wireless Styli und PDAs statt, sowie optional mit Tablet-PCs, welche eine 2D-GUI zur Systemkontrolle bieten. Diese Geräte, außer des Tablet-PCs, werden zur direkten Interaktion innerhalb des Demonstrators benutzt. Dabei entspringt dem jeweiligen Interaktionsgerät ein sog. *Pick-Ray*, der als eine Art Laserstrahl visualisiert wird. Der Pick-Ray implementiert die Ray-Casting Technik (siehe 2.1.4), welche im vorangegangenen Kapitel beschrieben wurde. Der Benutzer kann mit dem Laserstrahl auf Objekte zeigen und diese mittels eines Buttons selektieren, um sie dann zu manipulieren. Im VRGeo Demonstrator wird der Pick-Ray ebenfalls genutzt um innerhalb der 3D-Menüs (siehe 3.5) zu navigieren und Funktionen durch Aktivieren der Einträge aufzurufen. Als Interaktionstechnik wird hier die Scaled-Grab-Technik angewendet (siehe 2.1.4).

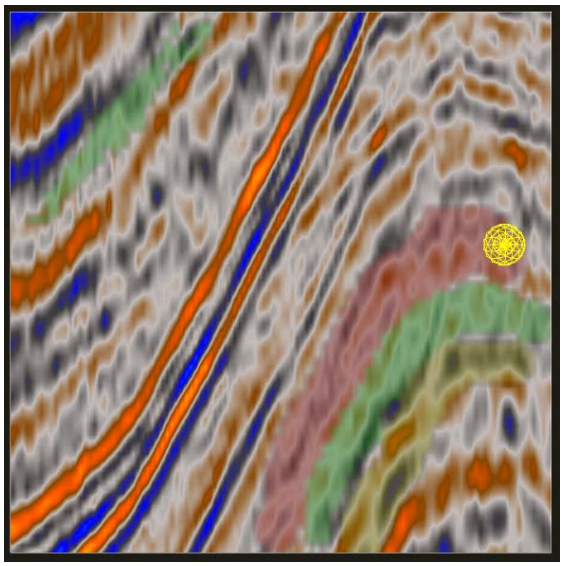


Abbildung 3.4.: *Eine Schnittebene mit beispielhafter Markierung in den Farben rot, grün und gelb im rechten, unteren Teil*

3.4.3. Zeichnen-Modus

Um Unsicherheiten in den seismischen Daten zu markieren, bietet der VRGeo Demonstrator ein Zeichnenwerkzeug, das sog. *Sketch-Tool* an. Hierzu ist ein Moduswechsel nötig, der durch die Betätigung eines Buttons auf dem Eingabegerät erreicht wird. In diesem Fall verschwindet der Pick-Ray und es erscheint eine freischwebende Kugel, mit der es möglich ist, in einer Volumenlinse zu zeichnen. Will man auf eine Schnittebene zeichnen, erscheint auf dieser ein Kreis (siehe Abbildung 3.4). Der Durchmesser der Kugel und des Kreises können verstellt werden, was Auswirkung auf die Stärke des Markierens oder Zeichnens hat. Folglich ist das Zeichnen lediglich in einer Volumenlinse oder auf einer Schnittebene möglich. Abbildung 3.4 zeigt eine Schnittebene, die jeweils in den Farben rot, grün und gelb an der rechten, unteren Seite beispielhaft markiert wurde. Das Zeichnen in einer Volumenlinse geschieht in 3-DOF und auf einer Schnittebene in 2-DOF. Der Demonstrator bietet vier verschiedene Visualisierungsmöglichkeiten um die markierte Stellen anzuzeigen:

- Desaturate
- Blur
- Warp
- Overlay

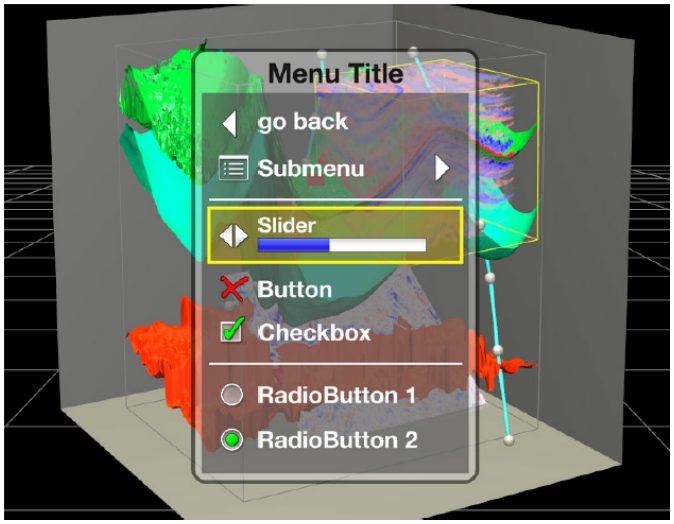


Abbildung 3.5.: Das 3D-Menu im VRGeo Demonstrator

Die Einstellungen können über das Untermenü „Uncertainty“ mittels Radiobuttons ausgewählt werden.

3.5. Die Systemsteuerung im VRGeo Demonstrator

Im Gegensatz zur direkten Interaktion mittels Navigation, Selektion und Manipulation in Virtuellen Umgebungen betrachten wir nun die indirekte Interaktion über Menüs und andere Bedienelemente im VRGeo Demonstrator. Als Systemsteuerung wird indes der Vorgang bezeichnet, bei dem der Benutzer (siehe auch 2.2):

- einen Befehl an das System gibt, um eine bestimmte Funktion auszuführen
- den Interaktionsmodus ändert
- den Systemstatus ändert

3.6. 3D-Menüs im VRGeo Demonstrator

Die Systemsteuerung im VRGeo Demonstrator findet über die sog. 3D-Menüs statt. Diese bieten die direkte Schnittstelle zu allen Funktionen auf die Elemente des Demonstrators. Lediglich die Änderung des Interaktionsmodus von Pick-Ray-Interaktion zu Zeichnenmodus wird über einen Button auf dem Eingabegerät

ausgeführt. Das 3D-Menü ist immersiv (siehe auch 2.1), d.h. es ist in die Szene eingebunden und kann innerhalb dieser mit einer 3-DOF-Interaktion (Rotation des 3D-Menüs ist unerwünscht) bewegt werden. Die Transparenz der Menüs sorgt dafür, dass Elemente, die dahinter liegen, nicht vollständig verdeckt werden. Somit kann man bei Aktionen, bei denen ein direktes Feedback erwünscht ist, das verändernde Objekt immer im Auge behalten.

Ein 3D-Menü besteht aus einer Reihe von unterschiedlichen Bedienelementen wie Knöpfen (buttons), Untermenüs (submenus), Schiebereglern (slider), Kontrollkästchen (checkboxes) und Optionsschaltflächen (radiobuttons) (siehe Abbildung 3.5). Die Navigation innerhalb des Menüs geschieht mittels Pick-Ray-Interaktion. Für die Menüinteraktion werden auf dem Eingabegerät 2 Buttons (siehe auch Abbildung 5.1) benötigt: einer, um einen Menüeintrag zu aktivieren und ein weiterer um das Menü aufzurufen. Bei der Aktivierung eines Untermenüs steigt man tiefer in die Menü-Hierarchie. Das heißt, jedes Menü kann ein weiteres beinhalten. Der Schieberegler kann direkt im Menü bewegt werden und wird nicht in einem zusätzlichen Menü geöffnet. Dabei ist es nicht notwendig, ihn direkt zu greifen. Da das Menü ein blockzentrieres Layout hat, reicht es, in den Schieberegler-Block zu „klicken“ und dann mit der entsprechenden Handbewegung den Regler nach rechts bzw. nach links zu bewegen. Das ist ein wichtiger Unterschied zu den gewöhnlichen 2D-Menüs, dass sich neben dem Aufrufen von Funktionen auch Parameter direkt im Menü verändern lassen können.

Die 3D-Menüs im VRGeo Demonstrator sind kontextabhängig. Für jedes Objekt gibt es ein spezielles Kontextmenü, das mit unterschiedlichen Funktionen belegt ist. Der Vorteil hierbei ist, dass sich der Benutzer nicht mit überladenen Menüs abmühen muss, sondern kleinere Menüeinheiten mit weniger Elementen angeboten bekommt, je nachdem, auf welches Objekt das Menü aufgerufen wird.

3.7. Elemente des VRGeo Demonstrators und deren Funktionen

3.7.1. Arbeitsbereich (Workspace)

Im Kontext des VRGeo Demonstrators werden sog. *Workspaces* oder Arbeitsbereiche benutzt, welche jeweils einen volumetrischen Datensatz sowie dazugehörige, benutzerdefinierte Elemente enthalten. Der Arbeitsbereich ist in einem Quader eingeschlossen. Es können lediglich die Innenseiten des Quaders gesehen werden, da die Außenseiten durch Backface-Culling entfernt wurden. Dazu werden einfach die Oberflächennormalen der Außenseiten umgedreht. Auf diese Weise kann der Benutzer, unabhängig von der Orientierung, in den Workspace hineinblicken. Mit dem Pick-Ray kann somit der Workspace an der rückwärtigen Begrenzung ange-

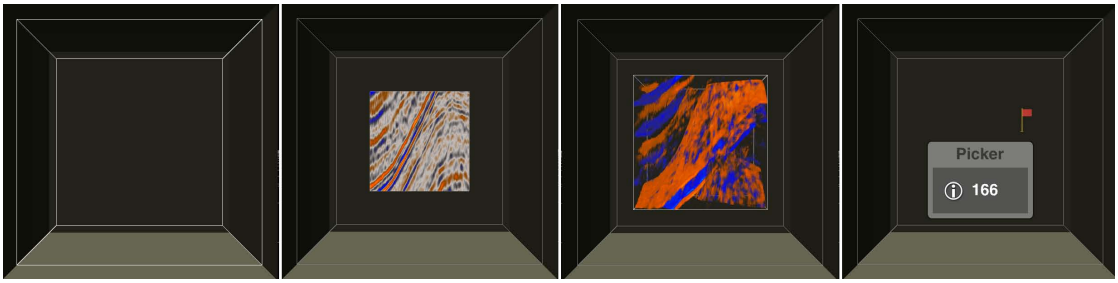


Abbildung 3.6.: *Die Objekte im VRGeo Demonstrator v.l.n.r: Leerer Arbeitsbereich, Schnittebene, Volumenlinse, Markierung*

fasst und in der Szene positioniert werden. Die Translation und Rotation kann in einer 6-DOF Interaktion bewerkstelligt werden. Das Kontextmenü des Arbeitsbereiches beinhaltet folgende Funktionen:

- Laden/Speichern des aktuellen Arbeitsbereiches
- Löschen
- Umbenennen
- Erzeugung neuer Objekte

3.7.2. Volumenlinse (Volume Lense)

Innerhalb eines Arbeitsbereichs können mehrere Volumenlinsen erzeugt werden, die für die räumliche Darstellung der sich im Arbeitsbereich befindenden seismischen Daten genutzt werden. Die Volumenlinsen sind beliebig skalierbar und können im Rahmen des Workspaces in 6-DOF frei orientiert und positioniert werden. Eine weitere Einstellungsmöglichkeit der Volumenlinsen ist die Darstellungsqualität. Diese bestimmt die Anzahl der Schnittebenen, die benutzt werden, um das Volumen zu visualisieren. Eine Reduzierung der Darstellungsqualität führt somit zu einer Leistungssteigerung der Applikation. Die Skalierung und Darstellungsqualität lässt sich direkt über einen Schieberegler im 3D-Menü verändern. Das Kontextmenü lässt folgende Einstellungen zu:

- Löschen
- Umbenennen
- Skalieren
- Darstellungsqualität

3.7.3. Schnittebene (Volume Slice)

Ähnlich wie die Volumenlinsen sind auch die Schnittebenen für die Darstellung der Volumendaten verantwortlich, mit dem Unterschied, dass hier jeweils eine Schicht des Volumens angezeigt wird. Es können beliebige Schnittebenen erstellt, diese skaliert und innerhalb des Arbeitsbereiches in 6-DOF frei positioniert werden. Die Menüeinstellungen hierzu sehen folgendermaßen aus:

- Löschen
- Umbenennen
- Skalieren

3.7.4. Markierung (Data Picker)

Ein weiteres Element des VRGeo Demonstrators ist der sogenannte *Data Picker*. Dieser wird verwendet, um einen Datenpunkt aus dem volumetrischen Datensatz direkt darzustellen. Beim Zeigen oder Bewegen einer solchen Markierung wird ein Label angezeigt, welches den Dichtwert des volumetrischen Datensatzes an dieser Stelle angibt. Markierungen erlauben keine Rotation und bieten somit eine 3-DOF-Interaktion. Die Menüeinstellungen beim Data-Picker umfassen:

- Löschen
- Umbenennen

3.7.5. Globale Einstellungen

Außerhalb des Arbeitsbereiches gibt es zusätzlich globale Einstellungsmöglichkeiten, die zu berücksichtigen sind. Hier ist das Laden eines abgespeicherten oder Erzeugen eines neuen Arbeitsbereichs möglich. Zusätzlich kann auf Benutzereinstellungen zugegriffen werden, welche globale Einstellungen zulassen. Das globale Menü erlaubt folgende Einstellungen:

- Erstellen eines neuen Arbeitsbereichs
- Laden eines vorhandenen Arbeitsbereichs
- Benutzereinstellungen
- Beenden der Applikation

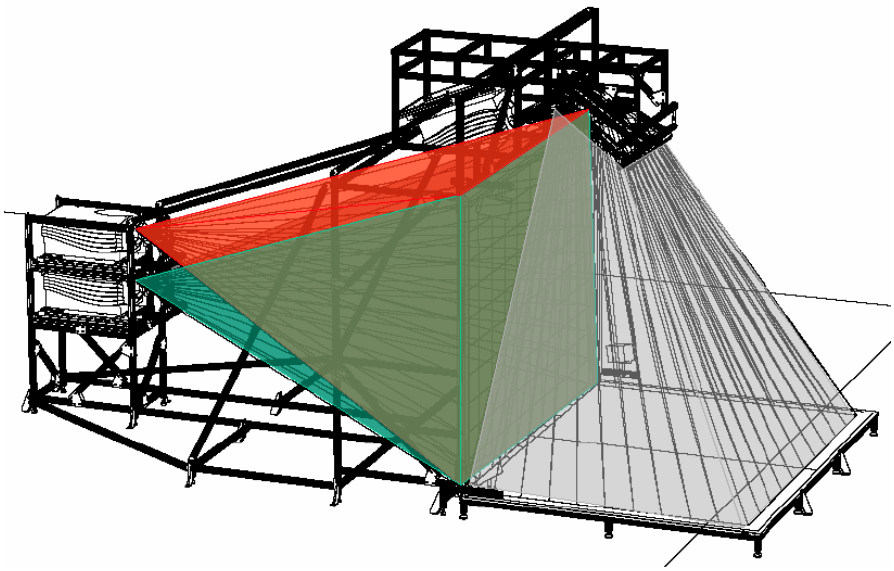


Abbildung 3.7.: Das TwoView-Display

3.8. Das TwoView-Display System

Display-Systeme für VR-Applikationen müssen für die Erzeugung einer stereoskopischen Darstellung jeweils ein Bild pro Auge erzeugen. Die Bilder müssen zusätzlich getrennt werden, so dass für jedes Auge nur das zugehörige Bild zu sehen ist. Dies geschieht üblicherweise durch sog. Blendenbrillen (Shutterbrillen), deren Gläser aus zwei Flüssigkristallanzeigen (je eine für jedes Auge) bestehen, die elektronisch von durchlässig zu undurchlässig wechseln.

Das TwoView ist ein Projektionssystem für mehrere Benutzer. Es erlaubt die synchrone Darstellung zweier stereoskopischer Bilder, so dass zwei Benutzer gleichzeitig ihre jeweils korrekte Perspektive erhalten. Die Rückprojektion auf der Projektionsfläche erfolgt zur Zeit durch zwei Barco Galaxy 6000 DLP Projektoren. An diesen ist jeweils ein zirkular polarisierender Filter angebracht, wodurch das Bild für jeden User getrennt wird. Die Trennung der Augen erfolgt hingegen durch Shutterbrillen.

3.9. Zusammenfassung des Kapitels

In diesem Kapitel wurde die Entwicklungsumgebung für die Implementierung des Interaktionskonzeptes vorgestellt. Es wurde der Aufbau des Spracherkennungssystems von Microsoft Vista und dessen Programmierschnittstelle SAPI 5.3 präsentiert. Des Weiteren wurden das VR-Framework Avango und die darauf aufbauende VR-

Applikation VRGeo Demonstrator vorgestellt. Es wurde ausführlich auf die Interaktion mit den Elementen des Demonstrators sowie auf die Systemsteuerung über die 3D-Menüs und die unterschiedlichen Interaktionsmodi (Zeichnen-Modus und Pick-Ray-Modus) eingegangen. Zum Schluss wurde noch das TwoView Display-System vorgestellt. Mit dieser Grundlage soll nun ein multimodales Interaktionskonzept mit Spracheingabe und Zeigeoperation für den VRGeo Demonstrator entworfen und realisiert werden.

4. Entwurf und Realisierung eines multimodalen Interaktionskonzeptes

In Kapitel 2 wurde gezeigt, welche Interaktionstechniken und Konzepte existieren, um in einer Virtuellen Umgebung zu interagieren. Auch wurde näher in die Methoden der natürlichen Interaktion, wie Spracheingabe und Gestik, eingegangen. Hierbei lag der Fokus insbesondere auf der Zeigeoperation, welche die deiktische Funktion der Gestik darstellt. Es wurden Techniken der multimodalen Interaktion vorgestellt, die hilfreich sind, über Sprache und Zeigeoperation natürliche Interaktionskonzepte zu entwickeln. Es wurden ebenfalls viele Beispiele aus vorangegangenen Arbeiten betrachtet, welche mit multimodaler Interaktion eine effektive und mächtige Interaktionsschnittstelle realisiert haben. Kapitel 3 gab einen Überblick über das Spracherkennungssystem von Microsoft Vista, das VR-Framework Avango und die VR-Applikation VRGeo Demonstrator. Mit diesen Voraussetzungen soll nun ein Konzept vorgestellt werden, das eine natürliche Interaktion mit dem VRGeo Demonstrator ermöglichen soll.

Im Folgenden wird der Entwurf und die Realisierung eines natürlichen, multimodalen Interaktionskonzeptes für den VRGeo Demonstrator vorgestellt. Es wird die Server-Client-Architektur mit dem VRGeo Speech Control als Server und dem Demonstrator als Client präsentiert. Anschließend wird die Grammatik für das Spracherkennungssystem erläutert. Anhand von Beispielen wird gezeigt, wie diese Grammatik in den Sprachserver eingesetzt wird. Schließlich werden die Schnittstellen von Server und Client aufgezeigt und die jeweiligen Implementationsdetails besprochen.

4.1. Entwurf

In Kapitel 2 wurden die Anforderungen für ein multimodales Interaktionskonzept und Beispiele aus Veröffentlichungen besprochen. Auf diesen Grundlagen soll nun ein Entwurf für die Interaktion im VRGeo Demonstrator vorgestellt werden.

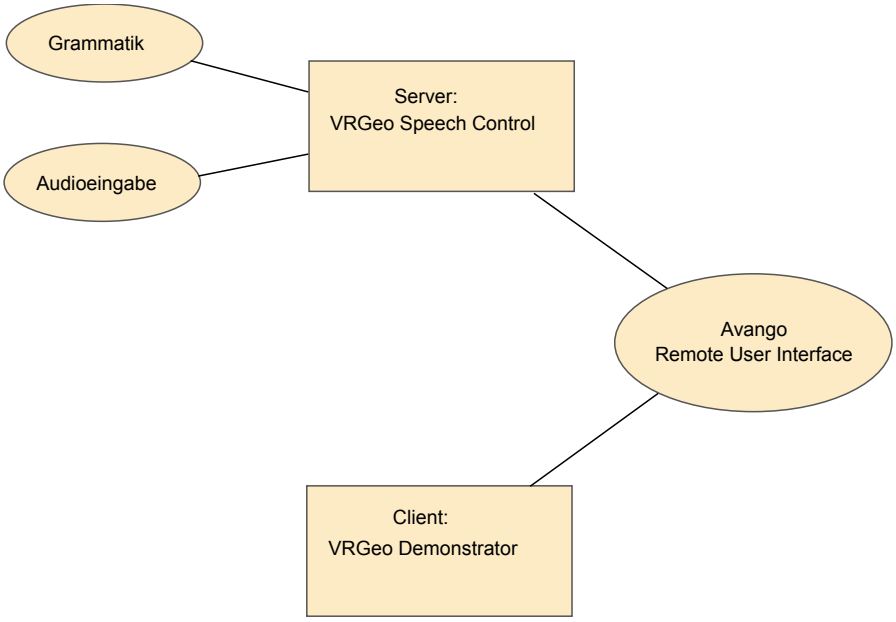


Abbildung 4.1.: *Schematische Darstellung der Server-Client-Architektur*

4.1.1. Natürliche Interaktion

In Kapitel 2.2.4 wurde gezeigt, dass durch multimodale Interaktion mit Spracheingabe und Zeigeoperation eine Benutzerschnittstelle natürlicher gestaltet werden kann [Latoschik u. a., 1998]. Aus diesem Grund soll am VRGeo Demonstrator zusätzlich ein Sprachserver integriert werden, welcher erkannte Sprachbefehle an die Applikation zurücksendet. Diese lösen daraufhin die entsprechende Funktion aus. Die Sprachapplikation soll auf einem separaten Rechnersystem implementiert werden. Dies hat den Vorteil, dass der Demonstrator unabhängig vom Spracherkennungssystem ist und dieses einfach ausgetauscht werden kann. Ein weiterer Vorteil ist, dass die Sprachapplikation die Leistung des Demonstrators nicht beeinträchtigt, da beide auf unterschiedliche Ressourcen zugreifen.

Die Interaktion im Demonstrator findet aktuell mittels Wii Remote und Pick-Ray-Interaktion (siehe 3.4.2) statt. Es bietet sich an, die vorhandene Interaktionstechnik beizubehalten und den Pick-Ray für die Zeigeoperation des multimodalen Konzeptes zu nutzen. Auf diese Weise kann auf zusätzliche Eingabegeräte verzichtet und die existierende Implementierung genutzt werden.

4.1.2. Einfache Erlernbarkeit des Systems

Sprachsteuerung gilt als Interaktion im Expertenmodus, da der Benutzer die Sprachbefehle und die Funktionalität der Applikation genau kennen muss. Die Herausfor-

derung hierbei ist es, ein Interaktionskonzept zu entwickeln, das sowohl Experten als auch Laien einen schnellen Einstieg ermöglichen soll. Um das zu realisieren, soll eine kontextabhängige Hilfe implementiert werden. Diese Hilfe soll in Form eines *Hilfe-Panels* in der Szene erscheinen und abhängig vom Kontext die momentan gültigen Sprachbefehle anzeigen. Auf diese Weise muss der Benutzer nicht im Voraus Listen von Sprachbefehlen auswendig lernen, um mit dem System interagieren zu können. Es ist auch kein Unterbrechen der Arbeit nötig, was die Natürlichkeit des Konzeptes beeinträchtigen würde. Der Benutzer soll jederzeit wissen, welche Sprachbefehle er benutzen kann und nach und nach auf die Hilfe verzichten können.

4.1.3. Interaktion mittels Sprachsteuerung

Die Interaktion im VRGeo Demonstrator findet mittels Pick-Ray-Interaktion und Wii Remote als Eingabegerät statt (siehe 3.4). Folgende Interaktionsmöglich-keiten sollen auch über Sprache gegeben sein:

- **Systemsteuerung**: Der Benutzer soll Objekte erstellen und löschen, den Interaktionsmodus (Zeichnen-Modus, Pick-Ray-Modus) ändern und den Sprachserver mit dem Client verbinden können.
- **Manipulation**: Die Skalierung der Volumenlinse und Schnittebene soll über Sprache möglich sein.
- **Symbolischer Input**: Es sollen über Sprache die verschiedenen Objekte im Demonstrator umbenannt werden können (siehe 3.7).

4.1.4. Semantische Interpretation

Jeder Benutzer hat eine unterschiedliche Art sich auszudrücken. Aus diesem Grund soll es möglich sein, für die gleiche Funktion, mehrere Sprachbefehle zu haben, die allesamt die gleiche semantische Interpretation erlauben. Auf diese Weise kann das System für jeden Benutzer natürlicher gesteuert werden, da er sich auf seine Art und Weise ausdrücken kann.

4.2. Implementierung

Das im Rahmen dieser Diplomarbeit entwickelte System basiert auf einer Server-Client-Architektur, die schematisch so aufgebaut ist wie in Abbildung 4.1 gezeigt wird. Der VRGeo Speech Control-Server basiert auf Microsoft Vistas Spracherkennungssystem (siehe 3.1) und gibt die erkannten Sprachbefehle an den Client weiter. Der Server ist ebenfalls für die Grammatik zuständig, die in Form von

XML-Dateien eingelesen werden und für die Audioeingabe, die in Form eines Streams durch ein Mikrofon eingespeist wird. Der VRGeo Speech Control-Server verarbeitet also das eingehende Audiosignal und gibt durch die festgesetzten Regeln der Grammatik das erkannte Resultat zurück. Die Schnittstelle zum Client bildet das Avango Remote User Interface, mit welchem verschiedene Applikationen über das TCP-Protokoll in einem Netzwerk kommunizieren können. Der VRGeo Demonstrator (siehe 3.3) dient als Client, der die Sprachbefehle vom Sprachserver empfängt und dann die entsprechenden Funktionen ausführt.

4.2.1. Der VRGeo Speech Control-Server

Wie schon in Kapitel 4.1 erwähnt, wurde für die Spracherkennung die Programmiersprache C# und als Sprachschnittstelle der System.Speech.Recognition-Namespace unter dem .NET Framework 3 benutzt. Die Hauptbestandteile der Sprachschnittstelle sind:

- Sprach-Server (SpeechServer)
- Grammatik (Grammar)
- Spracherkennungssystem (SpeechRecognitionEngine)
- Remote User Interface (RuiClient)

Der SpeechServer stellt mittels eines Host-Namens und einer Port-Nummer die Verbindung zum VRGeo Demonstrator über das Avango Remote User Interface (RUI) her. Der Event-Handler der RUI wartet auf Sprachbefehle und führt ein Update auf das entsprechende Field (siehe 3.2.1) durch, das mit dem Demonstrator verbunden ist. Die SpeechRecognitionEngine-Klasse hat die Methode `SetInputToDefaultAudioDevice()`, welche der Instanz von SpeechRecognitionEngine den Standard-Audio-Input zuweist. Dieser lässt sich in der Systemsteuerung von Windows unter den Sound-Einstellungen definieren. `handleEvent()` ruft die Methode `speechRecognized()` auf, sobald SpeechRecognitionEngine einen Audio-Stream verarbeitet hat und eine erkannte Phrase zurückgibt. Mit `loadGrammar()` werden alle Grammatiken in Form von XML-Files in SpeechRecognitionEngine geladen. Die Grammar-Klasse liest die einzelnen Grammatiken, die in Form von SRGS-Dokumenten vorliegen. Jede Grammatik hat einen eindeutigen Namen und kann somit über die Methode `enableGrammar()` aktiviert bzw. deaktiviert werden. Es gibt zusätzlich noch eine sog. *Dictation Grammar*. Diese beinhaltet ein Vokabular von ungefähr 5000 Wörtern mit Expertenkontext aus der Öl- und Gasindustrie ([bhpbilliton, 2008]) und wird eingesetzt, um im Demonstrator über Sprache Objekte zu benennen.

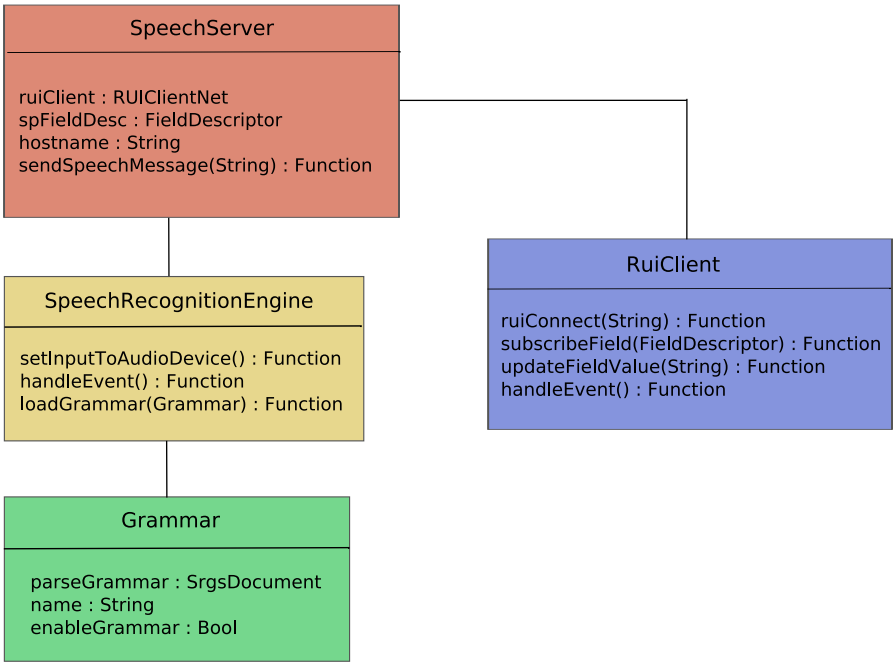


Abbildung 4.2.: *Klassendiagramm des VRGeo Speech Control-Servers.*

4.2.2. Die Benutzeroberfläche des VRGeo Speech Control-Servers

Die Benutzeroberfläche wurde innerhalb des System.Windows.Forms-Namespace gestaltet. Dieser enthält Klassen zum Erstellen Windows-basierter Anwendungen, mit denen die im Betriebssystem Microsoft Windows verfügbaren Benutzeroberflächenfeatures optimal genutzt werden können. Es gibt jeweils Klassen für die Erstellung von Text Box-, Combo Box-, Label-, List View- und Button-Steuerelementen. Die Anordnung der Steuerelemente im Anzeigenbereich kann mit der Layout-Klasse bestimmt werden.

Wie man in Abbildung 4.3 sehen kann, ist das obere Drittel der GUI für die Verbindung mit dem VRGeo Demonstrator verantwortlich. Unter Hostname kann der Rechnername angegeben werden, auf den die Applikation läuft, und unter Port wird der Wert für den TCP-Port eingesetzt. Mit „connect“ wird anschließend die Verbindung zum Demonstrator hergestellt. Mit „Send Message To Client“ kann manuell ein Sprachbefehl eingegeben werden. Das erweist sich als besonders praktisch, wenn man zu Testzwecken eine Phrase testen will, ohne dass man die Grammatik erweitern muss. Diese Phrase wird dann direkt in das Avango-Field geschrieben, durch das Server und Client verbunden sind. „Switch Grammar“ ist eine Combo-Box und beinhaltet alle Grammatiken, welche die Instanz

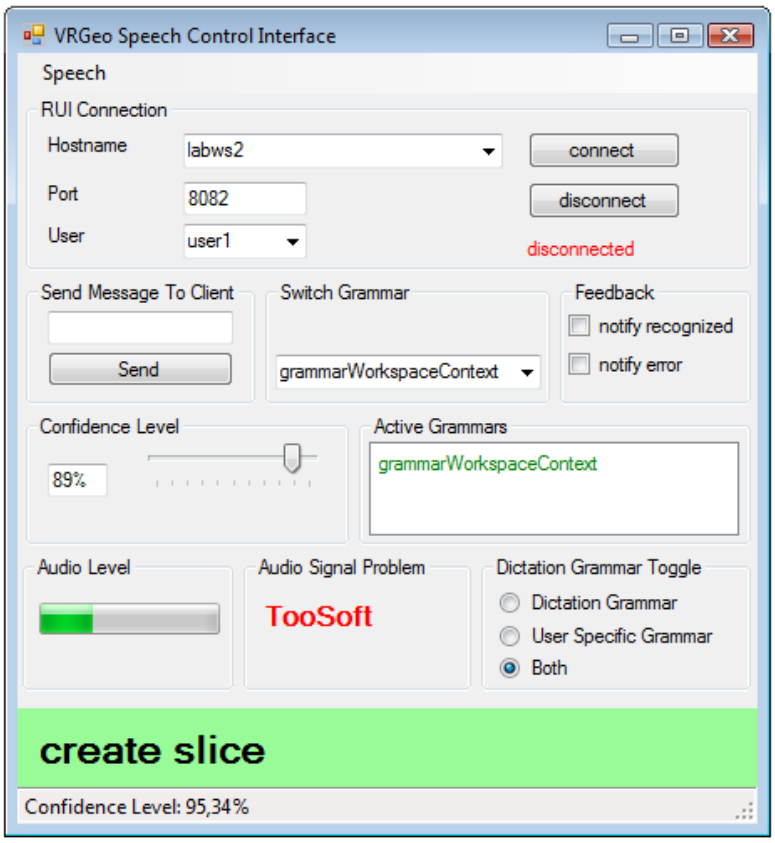


Abbildung 4.3.: *Die Benutzeroberfläche des VRGeo Speech Control mit den unterschiedlichen Einstellungsmöglichkeiten.*

von SpeechRecognitionEngine geladen hat. Hier kann man auch zu Testzwecken und *on-the-fly* Grammatiken wechseln. Unter „Feedback“ kann man sich entscheiden, ob bei falsch bzw. richtig erkannten Phrasen eine Rückmeldung erfolgen soll. Diese Rückmeldung erfolgt über das „Rumble-Feature“ der Wii Remote in Form einer Vibration. Es existiert auch ein akustisches Feedback über den Lautsprecher des Headsets, das während der ganzen Interaktion aktiviert ist. Das akustische Feedback lässt einen Ton erklingen, wenn der „Confidence Level“ der erkannten Phrase unter dem eingestellten Wert liegt. Über den „Confidence Level“ kann die Toleranz reguliert werden, mit der das Spracherkennungssystem eine Phrase als erkannt zurückgibt bzw. verwirft. Unter „Active Grammars“ kann beobachtet werden, welche Grammatiken im Moment aktiv sind. Da der Mikrofonpegel eine wichtige Rolle für die Erkennungsqualität spielt, kann unter „Audio Level“ kontrolliert werden, ob das System übersteuert wird oder nicht. Unter „Dictation Grammar Toggle“ kann angegeben werden, ob die „Dictation Grammar“ aktiviert

werden soll oder nicht. Am unteren Ende der GUI kann man auf einen grünen Balken die erkannte Phrase sehen und darunter den „Confidence Level“, mit dem diese erkannt wurde. Liegt der „Confidence Level“ der erkannten Phrase unter dem eingestellten Wert, färbt sich der Balken rot.

4.3. Grammatik

Eine Grammatik für Spracherkennungssysteme ist eine Menge von Wörtern, welche dem System mitteilen, auf welche Muster es „hören“ soll. Je geringer die Auswahl der Wortmuster ist, umso schneller und genauer ist die Erkennung, da das System unter einer kleineren Wortmenge entscheiden muss. Im nächsten Abschnitt wird der W3C-Standard für Grammatiken näher beschrieben.

4.3.1. W3C Speech Recognition Grammar Specification

Die Grammar-Klasse des System.Speech-Recognition-Namespaces unterstützt den W3C Speech Recognition Grammar Specification Standard (SRGS) [SRGS, 2008]. Dieser unterstützt zwei unterschiedliche Formate, wovon das eine auf XML und das andere auf das ABNF Format [Knuth, 1964] basiert. Wir beschränken uns hier auf das XML-Format [XML, 2008].

Die Elemente der SRGS

Es werden nun die Elemente der SRGS beschrieben die nötig sind um eine Grammatik zu definieren.

- `<grammar>` das Wurzelelement einer Grammatik
- `<rule>` dient zur Erzeugung von Grammatikregeln
- `<item>` dient der Definition einzelner Grammatikeinträge
- `<token>` wird zur Bildung von zusammengesetzten Begriffen verwendet
- `<one-of>` wird zur Erzeugung von Auswahllisten verwendet
- `<ruleref>` wird zur Referenzierung von Grammatikregeln verwendet
- `<tag>` wird zur semantischen Interpretation von Ausdrücken verwendet
- `<example>` dient hauptsächlich zu Dokumentationszwecken und kann zum Testen des Grammatik Interpreters verwendet werden

Jedes Element <grammar> ist eine Sequenz von <rule>-Tags. Die Top Level <rule> wird durch das <root> Attribut im <grammar>-Tag referenziert und das Attribut <scope> jeder <rule> legt fest, ob die Regel für die Instanz des SpeechRecognitionEngine sichtbar ist (public oder private). Sublevel <rule>-Referenzen erstellt man mit dem <ruleref>-Tag. Das <item>-Tag wird verwendet, um Phrasen zu gruppieren und das <one-of>-Tag wird für eine Auswahl möglicher Phrasen eingesetzt. Mit dem <tag>-Element hat man die Möglichkeit unterschiedlichen Phrasen dieselbe Interpretation zu geben. SRGS spezifiziert nicht die Inhalte dieser Elemente. Das wird vom W3C Standard Semantic Interpretation for Speech Recognition übernommen [SISR, 2008]. SISR basiert auf ECMAScript-Ausdrücke [ECMA, 2008] innerhalb der SRGS-Tags. Im Folgenden wird ein Ausschnitt aus einer SRGS-Grammatik gezeigt:

```
<grammar xml:lang="en-US" root="grammarLensContext"
   tag-format="semantics/1.0"
     version="1.0" xmlns="http://www.w3.org/2001/06/grammar">

  <rule id="grammarDelete" scope="public">

  <item>
    <ruleref uri="#deleteNow"/>
  </item>

  <rule id="deleteNow">
    <one-of>
      <item>delete now</item>
      <item>ok</item>
      <item>yes</item>
      <item>right</item>
    </one-of>
    <tag>out="delete now";</tag>
  </rule>

</grammar>
```

Es handelt sich um eine einfache Grammatik, die benutzt wird um einen Löschvorgang zu bestätigen. Der Benutzer kann folgende vier Ausdrücke dafür benutzen: „delete now“, „ok“, „yes“ und „right“. Das semantische Ergebnis des SpeechRecognizer ist immer „delete now“. In der Tabelle 4.1 sind alternative Sprachbefehle für die Sprachsteuerung im VRGeo Demonstrator aufgelistet. Diese Liste ist beliebig erweiterbar und kann an den Benutzer oder die Benutzergruppe angepasst werden.

Sprachbefehl	Alternativen	Funktion
create workspace	workspace	Erstellt Arbeitsbereich
	create workspace	
	new workspace	
open main menu	main menu	Öffnet das Hauptmenü
	open main menu	
close menu	close	Schließt jedes Kontextmenü
	close menu	
show speech help	speech commands	Aktiviert Hilfe für Sprachbefehle
	show speech commands	
	show help	
	show speech help	
ok	delete now	Bestätigung für Löschvorgang
	ok	
	right	
	yes	
create slice	slice	Erstellt Volumenscheibe
	new slice	
	create slice	
create lens	lens	Erstellt Volumenlinse
	new lense	
	create lens	
save workspace	save	Speichert Arbeitsbereich ab
	save workspace	
open slice menu	slice menu	Öffnet Kontextmenü
	open menu	
	menu	
rename slice	rename	Benennt Volumenscheibe
	rename slice	
lens quality	quality	Ändert Volumenlinsenqualität
	change lens quality	
	lens quality	
delete	erase	Löscht Objekt
	delete	

Tabelle 4.1.: *Alternative Sprachbefehle und ihre semantische Interpretation*

Objekt	Sprachbefehl	Funktion
global	create workspace	Erstellt Arbeitsbereich
	load workspace	Lädt einen Arbeitsbereich
	open main menu	Öffnet das Hauptmenü
	close menu	Schließt jedes Kontextmenü
	sketch mode on	Wechselt zum Zeichnenmodus
	sketch mode off	Wechselt zum Pick-Ray-Modus
	show speech help	Aktiviert Hilfe für Sprachbefehle
	user settings	Öffnet Menü für Einstellungen
	ok	Bestätigung für Löschvorgang
Arbeitsbereich	create slice	Erstellt Volumenscheibe
	create lens	Erstellt Volumenlinse
	workspace menu	Öffnet Kontextmenü
	delete workspace	Löscht Arbeitsbereich
	uncertainty [mode]	Ändert Uncertainty-Modi
	ok	Bestätigt Löschvorgang
	close menu	Schließt jedes Kontextmenü
	save workspace	Speichert Arbeitsbereich ab
	sketch mode on	Wechselt zum Zeichnenmodus
	sketch mode off	Wechselt zum Pick-Ray-Modus
	user settings	Öffnet Menü für Einstellungen
Volumenscheibe	create slice	Erstellt Volumenscheibe
	create lens	Erstellt Volumenlinse
	open slice menu	Öffnet Kontextmenü
	scale	Setzt Befehl zum Skalieren ab
	delete	Löscht Volumenscheibe
	ok	Bestätigt Löschvorgang
	close menu	Schließt jedes Kontextmenü
	rename slice	Benennt Volumenscheibe
	user settings	Öffnet Menü für Einstellungen
Volumenlinse	create slice	Erstellt Volumenscheibe
	create lens	Erstellt Volumenlinse
	open lens menu	Öffnet Kontextmenü
	scale	Setzt Befehl zum Skalieren ab
	quality	Ändert Volumenlinsenqualität
	delete	Löscht Volumenlinse
	ok	Bestätigt Löschvorgang
	close menu	Schließt jedes Kontextmenü
	rename lens	Benennt Volumenlinse
	user settings	Öffnet Menü für Einstellungen
Schieberegler	[10,20..100] percent	Setzt entsprechenden Wert
	up	Schiebt Regler nach rechts
	down	Schiebt Regler nach links
	full scale	Stellt volle Größe ein
	half scale	Stellt halbe Größe ein
RUI	connect user	Verbindet Benutzer mit Client

Tabelle 4.2.: *Sprachbefehle in Abhängigkeit vom Kontext*

4.3.2. Kontextabhängige Grammatik

Um eine möglichst minimale Grammatik zu benutzen und damit die Spracherkennungsqualität zu erhöhen, wurden abhängig vom Kontext mehrere Grammatiken erstellt. Da für jedes Objekt im Demonstrator eine unterschiedliche Funktionalität existiert, bietet sich das besonders an. So wird beispielsweise im Kontext einer Schnittebene kein Sprachbefehl zur Einstellung der Darstellungsqualität der Volumenlinse benötigt. In der Tabelle 4.2 werden alle Sprachbefehle zu dem jeweiligen Kontext aufgeführt. Es fällt auf, dass z.B. die Schnittebene und die Volumenlinse den gleichen Befehl für den Löschvorgang haben, nämlich „delete“. Da aber das Objekt referenziert wird, auf das mit dem Pick-Ray gezeigt wird, werden Zweideutigkeiten aufgelöst. Zusätzlich zu den in der Tabelle aufgeführten Sprachbefehlen, können auch die Einträge im Menü als Sprachbefehle abgesetzt werden. Dies soll das System intuitiver machen, da davon ausgegangen wird, dass der Benutzer geneigt ist, das auszusprechen, was er liest.

4.4. Implementierung am VRGeo Demonstrator

Das Interaktionskonzept des VRGeo Demontrators wurde in Kapitel 3.4 vorgestellt. Zur bestehenden Entwicklungsumgebung wurde eine weitere Klasse `speechControlInterface` implementiert. Diese hält eine Instanz des `userRepresenter`, welcher die Eingabegeräte und 3D-Menüs der einzelnen User verwaltet. Die Schnittstelle für die Sprachsteuerung ist nicht multi-user-fähig; deshalb gibt es nur eine Instanz des `speechControlInterface`. Die `speechControlInterface`-Klasse hat Zugriff auf die Funktionalität des `userRepresenter` und kennt somit das Eingabegerät des Benutzers, den Benutzer selbst und die von ihm selektierten Objekte, welche zur Referenzierung der Sprachsteuerung wichtig sind. Der Benutzer kann die Sprachsteuerung auf zwei Arten nutzen: implizit und explizit. Bei der impliziten Sprachsteuerung verarbeitet das Spracherkennungssystem durchgehend das Audiosignal. Bei der expliziten Sprachsteuerung wird die Spracherkennung durch Drücken eines Buttons aktiviert. Diese Einstellung kann man im VRGeo Demonstrator unter den User Settings vornehmen. Zusätzlich kann man dort die Hilfe für die Sprachbefehle anzeigen. Diese Hilfe wird in Form eines Panels angezeigt, das kontextabhängig die gültigen Sprachbefehle anzeigt. Dieses *Hilfe-Panel* unterscheidet sich optisch vom 3D-Menü (vergleiche Abbildung 3.5 mit Abbildung 4.4), so dass es nicht zu Verwechslungen durch den Benutzer kommen kann. Das *Hilfe-Panel* kann beliebig in der Szene positioniert und an- und ausgeschaltet werden. In Abbildung 4.4 ist die Hilfe für den Kontext der Schnittebene zu sehen.

Abbildung 4.4.: *Die kontextabhängige Hilfe mit gültigen Sprachbefehlen für eine Schnittebene*

4.5. Die Interaktion mit dem VRGeo Demonstrator

In Abbildung 4.5 ist in Form eines Sequenzdiagramms der Ablauf einer Interaktion mit Sprache und Zeigeoperation zu sehen. Nachdem der VRGeo Speech Control-Server gestartet wird, werden die Grammatiken in den SpeechRecognizer geladen. Diese werden daraufhin auf die korrekte Syntax überprüft. Der Benutzer interagiert nun mit der Applikation und zeigt auf ein bestimmtes Objekt in der Szene. Daraufhin sendet der Demonstrator den zugehörigen Grammatikkontext zum Sprachserver, welche anschließend aktiviert werden. Der Benutzer setzt einen Sprachbefehl ab, welcher in Form eines Audiosignals zum SpeechRecognizer gesendet wird. Dieser verarbeitet das Audio-Signal und überprüft, ob eine Übereinstimmung mit einem in der Grammatik gültigen Ausdruck existiert. Ist dies der Fall, wird das Ergebnis an den SpeechServer gesendet. Der SpeechServer schreibt den Ausdruck in das entsprechende Avango-Field, auf das wiederum im Demonstrator ein Update gemacht wird. Dieses Update löst dann die entsprechende Funktion im Demonstrator aus.

4.6. Zusammenfassung des Kapitels

In diesem Kapitel wurde der Entwurf für das multimodale Interaktionskonzept beschrieben. Es wurde die Implementierung des VRGeo Speech Control-Servers vorgestellt sowie dessen grafische Benutzeroberfläche mit den dazugehörigen Einstel-

lungsmöglichkeiten. Weiterhin wurde der W3C-Standard für Grammatiken präsentiert und an einem Beispiel gezeigt, wie diese eingesetzt wird. Schließlich wurde die Implementierung am VRGeo Demonstrator und die Interaktion darin mit Hilfe eines Sequenzdiagrammes beschrieben.

Im folgenden Kapitel wird eine Benutzerstudie vorgestellt, mit der überprüft werden soll, wie intuitiv und natürlich die Interaktion mit dem vorgestellten Konzept ist. Ebenfalls soll überprüft werden, ob Sprache als Interaktionsmittel von den Benutzern akzeptiert wird.

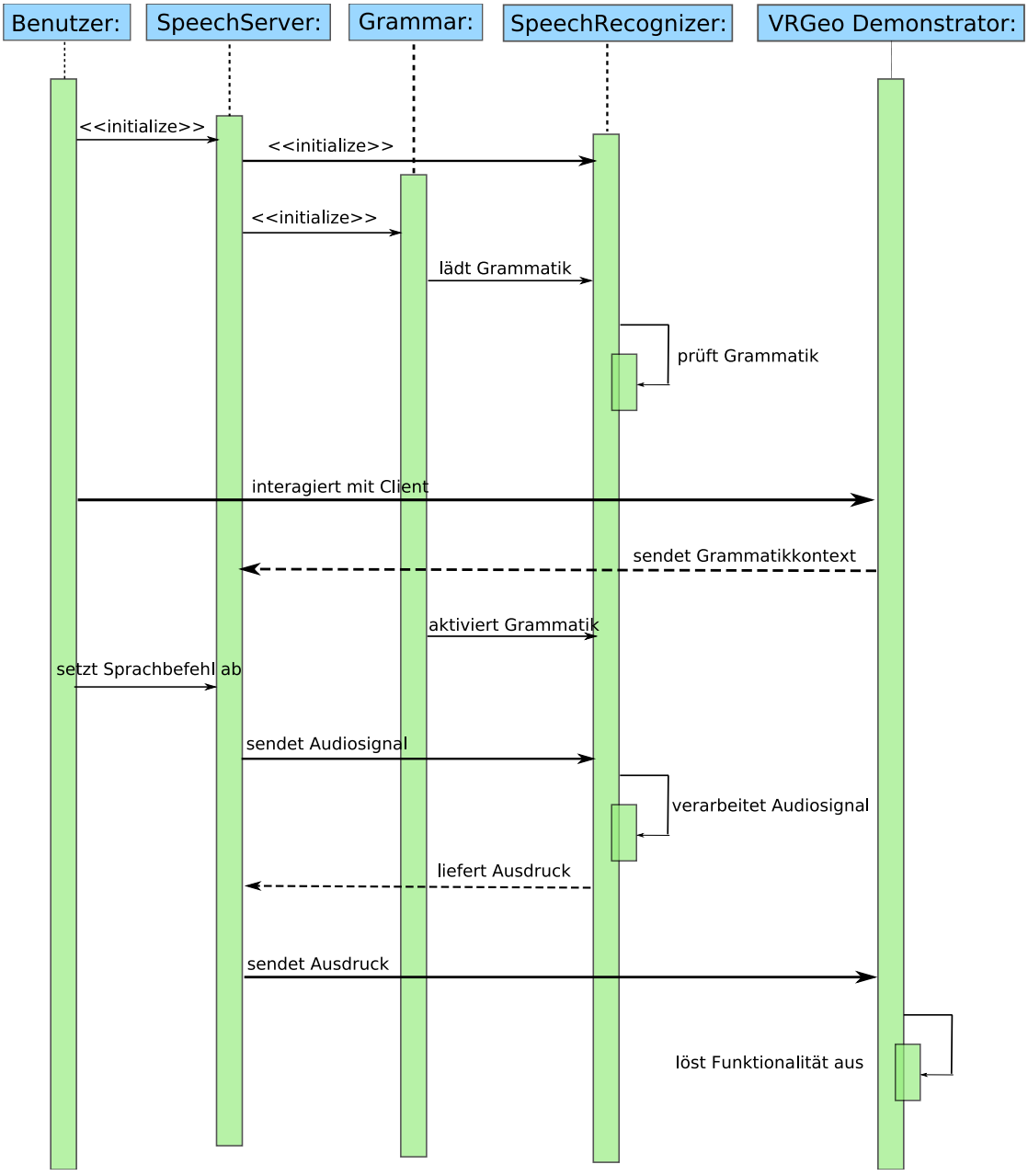


Abbildung 4.5.: *Sequenzdiagramm zur Interaktion im VRGeo Demonstrator*

5. Evaluierung

Nachdem in den vorangegangenen Kapiteln die Entwicklungsumgebung und das dafür implementierte Interaktionskonzept zur kombinierten Sprach- und Menü-Steuerung vorgestellt wurden, soll in diesem Kapitel die Benutzbarkeit des Systems untersucht werden. Dazu stellen sich folgende Fragen:

- Wird vom Benutzer Sprachsteuerung als Interaktionsmethode akzeptiert?
- Wenn ja: wie intuitiv ist die kombinierte Interaktion mit Spracheingabe im Kontext des VRGeo Demonstrators?
- Für welche Interaktion wird sich der Benutzer entscheiden, wenn ihm beide Methoden (Pick-Ray-Interaktion und Sprachsteuerung) zur Verfügung stehen?

Zur Untersuchung dieser Fragen wurde eine Benutzerstudie durchgeführt, die in diesem Kapitel präsentiert wird. Als erstes wird das Konzept und die Methode der Studie vorgestellt, welche eine Trainingsphase, die eigentliche Aufgabenstellung und das Ausfüllen eines Fragebogens umfasst. Schließlich werden die Ergebnisse der Studie ausgewertet, weitere Beobachtungen werden diskutiert und es werden die Kommentare der Benutzer aufgeführt.

5.1. Konzeption und Hypothese der Benutzerstudie

Um klar unterscheiden zu können, welche Interaktionsmethode der Benutzer anwendet, wird zwischen Sprachsteuerung und Pick-Ray-Interaktion unterschieden. Darunter versteht man:

- ***Sprachsteuerung***: Jede Aktion, die der Proband entweder durch Zeigeoperation und Spracheingabe oder nur durch Spracheingabe ausführt, wie es z.B. beim Moduswechsel der Fall ist, wo ein Zeigen auf ein bestimmtes Objekt nicht notwendig ist.
- ***Pick-Ray-Interaktion***: Das Steuern der Applikation über die 3D-Menüs und die Anwendung der physikalischen Steuerelemente auf dem Eingabegerät (siehe 3.4.2).

Während der Trainingsphase erlernt der Benutzer beide Methoden ausführlich. Danach bekommt er eine Aufgabe gestellt, die aus vier Teilaufgaben besteht. Jede dieser Teilaufgaben kann in atomare Aktionen aufgeteilt werden, welche entweder über Sprachsteuerung oder Pick-Ray-Interaktion ausführbar sind. Jede Aktion wird protokolliert. So ist es möglich, genau zu bestimmen, welche Aktion mit welcher Methode ausgeführt worden ist und wie groß der Anteil jeder Interaktionsmethode war. Auf diese Weise kann festgestellt werden, ob der Benutzer die Sprache als Interaktionsmöglichkeit akzeptiert und anwendet. Auch kann hier anhand des Verhältnisses ermittelt werden, ob sich der Benutzer jeweils für eine Interaktionsmöglichkeit entscheidet, oder beide kombiniert. Kombiniert er beide, kann angenommen werden, dass eine Mischung sinnvoller ist als jede Methode für sich betrachtet. Um zu messen wie intuitiv die Benutzerschnittstelle ist, wurde ein Fragebogen vorbereitet, welchen der Proband nach der Aufgabenstellung ausfüllen sollte. Die Antworten sollen Aufschluss darüber geben, ob und in welchem Maße die Interaktion mit dem System intuitiv war. Aus diesem Grund wurde auch der Benutzer während der Aufgabenstellung beobachtet und sein Verhalten wurde protokolliert.

Es wird angenommen, dass der Benutzer einen großen Teil der Aktionen über Sprachsteuerung ausführen wird. Auch wird davon ausgegangen, dass sich der Benutzer nicht für die eine oder für die andere Interaktionsmöglichkeit entscheidet, sondern dass er eine Kombination aus beiden bevorzugen wird.

5.2. Methode

5.2.1. Testumgebung

Der VRGeo Speech Control-Server läuft auf einer HP xw4400 Workstation mit einem Intel Core 2 Prozessor (2,66 GHz) und einem Windows Vista Business N Betriebssystem. Für den Demonstrator steht ein Rechnersystem mit einem Quad-Core AMD Opteron Prozessor (2,6 GHz) zur Verfügung, auf dem die Linux-Distribution CentOS installiert ist. Die Benutzerstudie fand im TwoView (siehe 3.8) statt und als Interaktionsgerät wurde eine getrackte Wiimote (siehe 3.4.1) benutzt. Das Plantronics CS60 USB Headset (siehe 2.4) wurde für die Spracheingabe eingesetzt. In Abbildung 5.1 sieht man die physikalischen Steuerelemente auf der Wii Remote und deren Funktionen für den Demonstrator. Für die Benutzerstudie kommen lediglich drei Buttons zum Einsatz. Der Button für das Öffnen des Kontextmenüs, der Primärbutton für die Selektion und Manipulation und der Button für den Wechsel vom Pick-Ray-Modus zum Zeichnenmodus.

Benutzer-ID	1	2	3	4	5	6	7	8	9	10	11	12
Geschlecht	m	m	m	m	m	m	m	m	m	m	w	m
Alter	32	30	28	27	27	26	23	24	38	28	26	53
Erfahrung in VR	1	4	1	2	1	4	5	2	2	5	3	4

Tabelle 5.1.: *Relevante Merkmale der Probanden.*

5.2.2. Subjects

Für die Benutzerstudie haben sich 12 Probanden zur Verfügung gestellt. Die Tabelle 5.1 gibt deren relevanten Merkmale an. Ihre Erfahrung in VR wurde daran gemessen, wie oft sie VR-Applikationen benutzen. Die Häufigkeit sollten sie folgendermaßen bewerten:

- täglich = 1
- oft = 2
- regelmäßig = 3
- selten = 4
- nie = 5

Bezüglich der Erfahrung mit VR-Systemen und des Alters ergeben sich folgende Durchschnittswerte:

- Erfahrung mit VR-Systemen: 2,83
- Alter: 30,2

5.2.3. Gruppenreihenfolge- und Einmaleffekt

Damit die Reihenfolge der in der Trainingsphase erlernten Interaktionsmethode keinen Einfluss auf die Wahl derselben hat, begann die Hälfte der Probanden mit dem Erlernen der Sprachsteuerung und die andere Hälfte mit der Pick-Ray-Interaktion. Bleibt der Anteil der benutzten Interaktionsmethode bei beiden Gruppen gleich, kann man davon ausgehen, dass kein Reihenfolgeeffekt existiert.

Es stellt sich noch die Frage, ob der Proband die gleiche Interaktionsroutine beibehält, wenn er die gleiche Aufgabe mehrmals nacheinander auszuführen hat.

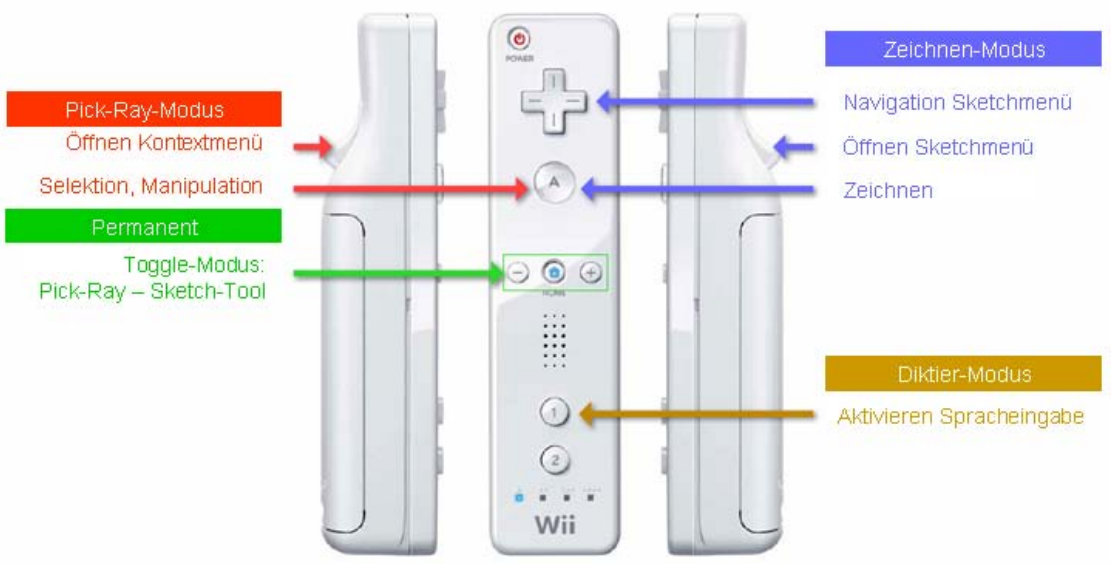


Abbildung 5.1.: *Die Buttons und ihre Funktionen auf der Wii Remote.*

Gibt es einen Einmaleffekt, oder bleibt die Interaktionswahl bei mehreren Versuchen gleich? Um das herauszufinden, wurde der Proband gebeten, die Aufgabenstellung drei Mal hintereinander auszuführen. Sind in den drei Durchgängen keine signifikanten Unterschiede bei der Wahl der Interaktionsroutine zu bemerken, kann angenommen werden, dass kein Einmaleffekt existiert.

5.2.4. Der Ablauf

Die Benutzerstudie bestand aus einer Einführung, einer Trainingsphase, der eigentlichen Aufgabenstellung und anschließend dem Ausfüllen eines Fragebogens. Vor dem Training bekam der Proband eine Einführung darüber, wie der VRGeo Demonstrator entstanden ist und wozu er dient. Er wurde darüber aufgeklärt, welchen Sinn die Benutzerstudie hat, wie lange sie dauern wird und wie sie aufgebaut ist. Anschließend wurde der Proband aufgefordert, die Shutterbrille und das Headset aufzusetzen. Während eines kurzen Sprechtests wurde der Mikrofonpegel richtig eingestellt und der Benutzer konnte ein Gefühl dafür bekommen, wie laut bzw. leise er ins Mikrofon sprechen kann. Die Dauer des Tests betrug ca. 30-40 Minuten.

5.2.5. Trainingsphase

In der Trainingsphase wurde der Benutzer mit beiden Interaktionsmethoden vertraut gemacht: Die Sprachsteuerung und die Pick-Ray-Interaktion. Um Reihenfol-

geeffekte auszuschließen, hatte die eine Hälfte der Probanden das Training mit der Sprachsteuerung begonnen und die andere Hälfte mit der Pick-Ray-Interaktion. Die Trainingsphase war sehr intensiv, denn es wurde Wert darauf gelegt, dass der Benutzer die Funktionen, die später in der Aufgabenstellung gebraucht werden, gut verinnerlicht. Nach der Trainingsphase unter Anleitung durfte nun der Proband so lange mit beiden Methoden trainieren, bis er sich für die eigentliche Aufgabenstellung sicher fühlte. Dieser Teil des Trainings ist sehr wichtig, da jeder Benutzer unterschiedlich lange Lernzeiten benötigt. Danach wurde mit der eigentlichen Aufgabe begonnen.

5.2.6. Die Aufgabenstellung

Die Aufgabenstellung wurde so konzipiert, dass sie einen normalen Arbeitsablauf eines Geowissenschaftlers oder eines jeden Experten simuliert, der mit dem VR-Geo Demonstrator arbeitet und seismische Daten exploriert. Es war wichtig alle Aspekte der Interaktion mit dem Demonstrator abzufangen, um eine reelle Arbeitssituation zu simulieren.

Vor der Aufgabenstellung wurde dem Probanden die Funktion des Eingabegerätes erklärt. Dabei wird zwischen vier Funktionen unterschieden:

- **Zeigefunktion**: Bei der Sprachsteuerung wird das Objekt, auf das mit dem Pick-Ray gezeigt wird, referenziert und darauf der Sprachbefehl angewendet.
- **Selektion/Manipulation**: Ein Objekt, das manipuliert werden soll, wird mit dem Pick-Ray selektiert und anschließend manipuliert.
- **Menü-Navigation**: Mit dem Pick-Ray wird in der Menü-Hierarchie navigiert und durch drücken eines Buttons die entsprechende Funktion ausgeführt.
- **Moduswechsel**: Durch Drücken des entsprechenden Buttons wird von dem Pick-Ray-Modus in den Zeichnenmodus gewechselt.

Um dem Benutzer alle Aspekte der Interaktion mit der Wii Remote bewusst zu machen, wurde die Aufgabenstellung so konzipiert, dass sie Zeigefunktion, Menü-Navigation, Moduswechsel, Manipulation und Zeichnen beinhaltet. Der Test wurde in 4 Teilaufgaben unterteilt und auf diese Weise dem Probanden mitgeteilt:

1. Erstelle bitte eine Schnittebene, vergrößere diese und zeichne etwas darauf!
2. Als nächstes schiebe bitte die Schnittebene nach hinten, zieh den Arbeitsbereich näher heran, erstelle eine Volumenlinse und setze deren Darstellungsqualität herunter!

3. Nun vergrößere bitte die Volumenlinse und wechsle in den Uncertainty Warp Modus!

4. Als nächstes, drehe bitte den Arbeitsbereich so, dass du die Veränderung sehen kannst, gehe zurück in den Uncertainty Overlay Modus und lösche alle Objekte!

Nun soll an der ersten Teilaufgabe gezeigt werden, wie sie in atomare Aktionen aufgeteilt wurde. Diese Schritte müssen für folgende Teilaufgabe ausgeführt werden:

- Erstelle bitte eine Schnittebene, vergrößere diese und zeichne etwas darauf!

1. Arbeitsbereich erstellen
2. Schnittebene erstellen
3. Schieberegler für die Skalierung sichtbar machen
4. Schieberegler bewegen
5. in den Zeichnenmodus wechseln
6. in den Pick-Ray-Modus wechseln

In der Tabelle 5.2 sind in der ersten Spalte die einzelnen Aktionen ersichtlich und in Spalte zwei und drei jeweils der Sprachbefehl für die Sprachsteuerung bzw. der Aufruf durch die Pick-Ray-Interaktion.

Insgesamt gibt es sieben atomare Aktionen, die sowohl mit Sprachsteuerung als auch mit Pick-Ray-Interaktion ausführbar sind:

- ***Slider-Interaktion***: Die Skalierung der Objekte findet mittels eines Sliders statt. Der Slider kann entweder über Sprache (z.B. „value up“, „value down“) oder Pick-Ray bewegt werden.

- ***Menü-Selektion***: Darunter sind diese Befehle zu verstehen, die direkt aus den Menüs gelesen und ausgesprochen werden.

- ***Wechsel des Interaktionsmodus***: Der Wechsel vom Zeichnenmodus zum Pick-Ray-Modus.

- ***Skalierung eines Objektes***: Bevor der Slider zur Skalierung des Objektes oder der Darstellungsqualität (bei Volumenlinsen) erscheint, muss der Befehl zur Skalierung gegeben werden. Entweder durch Öffnen des Kontextmenüs oder durch den Sprachbefehl „scale lens/slice“. Die eigentliche Skalierung geschieht somit über die Slider-Interaktion.

Aktion	Sprachsteuerung	Pick-Ray-Interaktion
1. Objekt erstellen	"create workspace"	Kontextmenü - New Workspace
2. Objekt erstellen	"create slice"	Kontextmenü - New Slice
3. Objekt skalieren	"scale slice"	Kontextmenü - Scale
4. Slider-Interaktion	z.B. "value up"	Interaktion mit Pick-Ray
5. Wechsel Interaktionsmodus	"sketch mode on"	Button Eingabegerät
6. Wechsel Interaktionsmodus	"sketch mode off"	Button Eingabegerät

Tabelle 5.2.: *Die Arbeitsschritte für Teilaufgabe 1 und deren Aufteilung in atomare Aktionen.*

- ***Löschen eines Objektes***: Das Löschen eines Objektes kann auch hier entweder durch das Kontextmenü oder durch den Sprachbefehl „delete“ ausgeführt werden.
- ***Wechsel des Uncertainty-Modus***: Der Uncertainty-Modus kann auf das gezeichnete Volumen geändert werden.
- ***Erstellen eines Objektes***: Das Erstellen eines Objektes bezieht sich sowohl auf den Arbeitsbereich, wie auch auf die Schnittebenen und Volumenlinsen.

Die ganze Aufgabe bestand aus insgesamt 16 atomaren Aktionen, die entweder über Pick-Ray-Interaktion oder Sprachsteuerung auszuführen waren. Aktionen, wie das Zeichnen oder Manipulieren von Objekten, wurden nicht protokolliert, da für diese nur eine Interaktionsmöglichkeit existiert.

5.2.7. Der Fragebogen

Nach dem Test wurde jeder Proband dazu aufgefordert, einen Fragebogen auszufüllen, um seine eigenen Eindrücke zum Interaktionskonzept zu vermitteln. Die vollständigen Fragebögen samt Auswertung befinden sich im Anhang. Durch den Fragebogen sollten Erkenntnisse zu folgenden Fragestellungen gewonnen werden:

- Wie wird die Anwendung der Sprache als Interaktionsmittel empfunden?

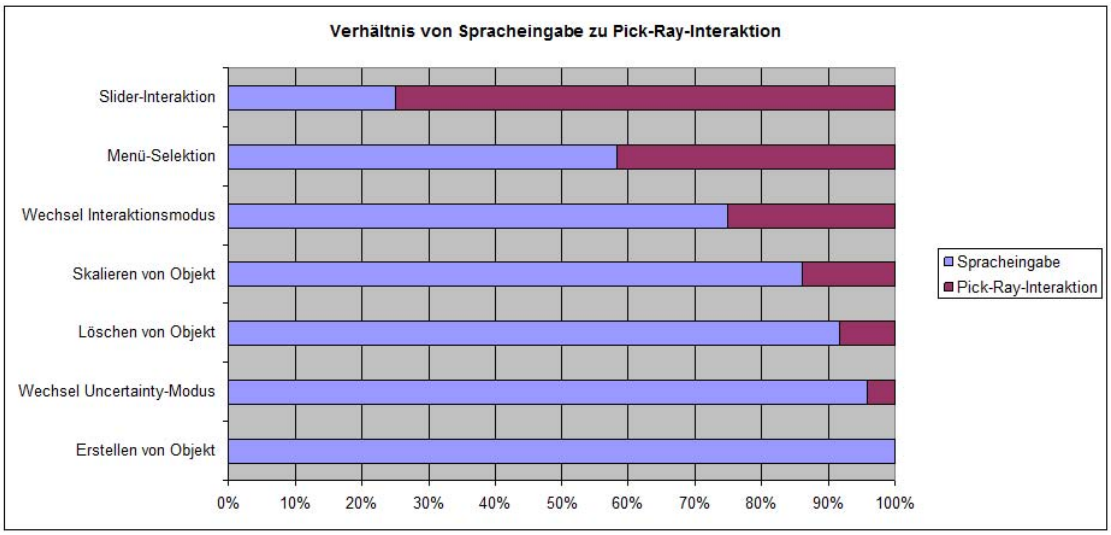


Abbildung 5.2.: *Der helle Teil der Balken zeigt den Anteil der Spracheingabe als Interaktionswahl über alle Benutzer für jede einzelne Aktion. Der dunkle Teil hingegen zeigt die Pick-Ray-Interaktion als Interaktionswahl an.*

- Sollte die Applikation nur mit Sprache gesteuert werden oder soll sie nur als ein Komplement zur konventionellen Steuerung gesehen werden?
- Gibt es Aktionen die schlecht über Sprache zu steuern sind und wenn ja welche?
- Ist es natürlicher durch Sprache und Zeigeoperation mit der VR-Applikation zu interagieren?

5.3. Auswertung der Ergebnisse

Die erste Frage, die in der Einleitung dieses Kapitels gestellt wurde, war: Wird Spracheingabe als Interaktionsmethode akzeptiert? Das Ergebnis dieser Benutzerstudie ergab, dass 76% aller Aktionen mit Sprachsteuerung ausgeführt wurden. Das ist ein sehr positives Ergebnis und bestätigt unsere Annahme, dass Sprache als Interaktionsmittel angenommen und angewendet wird. Dieser Effekt war aber nicht für alle Interaktionen gültig. Auf der Grafik in Abbildung 5.2 ist zu sehen, dass 75% der Slider-Interaktionen über Pick-Ray-Interaktion ausgeführt wurden, während für sämtliche Befehle zur Erstellung eines Objektes die Sprachsteuerung benutzt wurde.

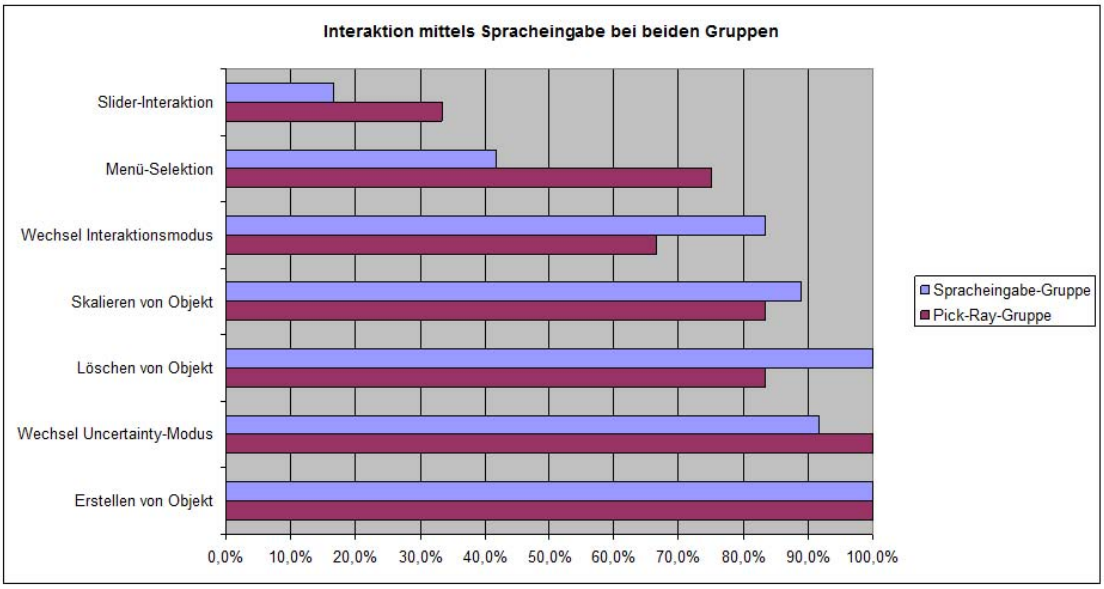


Abbildung 5.3.: *Hier wird lediglich die Spracheingabe als Interaktionswahl für die Spracheingabe- bzw. Pick-Ray-Gruppe angezeigt.*

5.3.1. Auswertung des Reihenfolge- und Einmaleffekts

Um den Reihenfolgeeffekt (siehe 5.2.3) zu untersuchen, wurden die Probanden in zwei Gruppen aufgeteilt. Die Spracheingabe-Gruppe hat die Trainingsphase mit der Sprachsteuerung während die Pick-Ray-Gruppe mit der Pick-Ray-Interaktion begonnen hat. Während der Aufgabenstellung wurde protokolliert wie oft ein Proband Spracheingabe bzw. Pick-Ray-Interaktion zur Ausführung der einzelnen Aktionen angewendet hat. Nun soll untersucht werden, ob ein signifikanter Unterschied der Wahl der Interaktionsart während der Aufgabenstellung bei beiden Gruppen existiert. Dazu wird im Folgenden der Signifikanztest beschrieben.

Signifikanztest

Zum statistischen Nachweis von Unterschieden oder Effekten, wie in diesem Beispiel der Reihenfolgeeffekt oder der Einmaleffekt, werden häufig Signifikanztests eingesetzt. Das Ergebnis eines solchen Tests wird als P-Wert (Wahrscheinlichkeitswert) ausgegeben. Anhand dieses P-Wertes wird entschieden, ob beobachtete Unterschiede statistisch signifikant sind (wenn der P-Wert kleiner ist als das Signifikanzniveau α von z.B. 5%) oder nicht [Beller, 2004].

Dazu muss vor dem Test eine Hypothese aufgestellt und dann anhand von mehr oder weniger genauen Stichprobenergebnissen über die Korrektheit dieser Hypothese entschieden werden. Zwei wichtige Varianten von Hypothesen sind

F-Test	*Spracheingabe-Gruppe*	*Pick-Ray-Gruppe*
Mittelwert	12,16666667	11,5
Varianz	8,566666667	5,1
Beobachtungen	6	6
Freiheitsgrade (df)	5	5
Prüfgröße (F)	1,679738562	
P(F<=f) einseitig	0,291547445	
Kritischer F-Wert bei einseitigem Test	5,050338814	

T-Test	*Spracheingabe-Gruppe*	*Pick-Ray-Gruppe*
Mittelwert	12,16666667	11,5
Varianz	8,566666667	5,1
Beobachtungen	6	6
Gepoolte Varianz	6,833333333	
Hypothetische Differenz der Mittelwerte	0	
Freiheitsgrade (df)	10	
t-Statistik	0,441726104	
P(T<=t) einseitig	0,334043151	
Kritischer t-Wert bei einseitigem t-Test	1,812461505	
P(T<=t) zweiseitig	0,668086301	
Kritischer t-Wert bei zweiseitigem t-Test	2,228139238	

Tabelle 5.3.: *Auswertung der Daten durch den F-Test und T-Test mittels Microsoft Excel. Der F-Test gibt an, dass es keinen signifikanten Unterschied bez. der Varianzen beider Gruppen existiert. Der T-Test zeigt, dass kein signifikanter Unterschied zwischen den Werten der beiden Gruppen existiert*

Unterschieds- und Zusammenhangshypothesen. Unterschiedshypothesen behaupten Unterschiede zwischen Gruppen, zum Beispiel in Bezug auf Mittelwerte oder Häufigkeiten. Zusammenhangshypothesen machen Aussagen über Korrelationen zwischen Variablen. Beide Arten von Hypothesen lassen sich weiter danach charakterisieren, ob sie ungerichtet oder gerichtet bzw. unspezifisch oder spezifisch sind. Ungerichtete Hypothesen behaupten nur das Vorliegen irgendeines Unterschieds bzw. Zusammenhangs, während gerichtete Hypothesen auch dessen Richtung spezifizieren, also ob eine Gruppe besser oder schlechter abschneidet als die andere oder ob eine Korrelation positiv oder negativ ausfällt. Unspezifische Hypothesen sagen nichts über die Größe des behaupteten Unterschieds bzw. Zusammenhangs, während spezifische Hypothesen die Größe genau angeben [Beller, 2004].

In dem vorliegenden Fall soll untersucht werden ob ein Unterschied bei der Wahl der Interaktion zwischen beiden Gruppen existiert. Die Prüfung erfolgt durch den Vergleich zweier einander ausschließender Hypothesen H_0 und H_1:

- H_0-Hypothese: Es existiert kein signifikanter Unterschied zwischen den untersuchten Gruppen.
- H_1-Hypothese: Es existiert ein signifikanter Unterschied zwischen den untersuchten Gruppen.

Die erste ungerichtet-unspezifische Hypothese wird *Nullhypothese* genannt und drückt aus, dass es keinen signifikanten Unterschied zwischen den beiden Gruppen

Spracheingabe-Gruppe	user 1	user 2	user 3	user 4	user 5	user 6
create object	3	3	3	3	3	3
select menu entry	1	2	1	1	2	2
scale object	1	3	2	3	3	3
slider interaction	0	3	1	0	2	0
mode switch	0	2	2	2	0	2
uncertainty mode switch	2	2	2	2	2	2
delete object	0	1	1	1	1	1
Gesamt	7	16	12	12	13	13

Pick-Ray-Gruppe	user 7	user 8	user 9	user 10	user 11	user 12
create object	3	3	3	3	3	3
select menu entry	1	2	1	0	0	1
scale object	3	3	3	2	3	2
slider interaction	0	2	1	0	0	0
mode switch	2	2	2	2	0	2
uncertainty mode switch	2	2	2	2	2	1
delete object	1	1	1	1	1	1
Gesamt	12	15	13	10	9	10

Tabelle 5.4.: *Gesamte Anzahl der Wahl von Spracheingabe als Interaktionsmethode für jeden Benutzer während der gesamten Aufgabenstellung.*

gibt. Von dieser Hypothese wird vorläufig ausgegangen. Die zweite nennt man *Alternativhypothese*, denn sie bietet das Gegenstück zur Nullhypothese H_0 [Beller, 2004].

Es gibt verschiedene Verfahren zur Durchführung von Signifikanztests. Der studentische T-Test ist ein Verfahren für die Prüfung von korrelativen Zusammenhängen zwischen zwei Gruppen und wird für drei Typen definiert:

- Gepaarte Gruppen: Jeder Wert der ersten Gruppe korrespondiert zu einem passenden Wert der zweiten Gruppe.
- Nicht gepaarte Gruppen mit gleicher Varianz: Werte der beiden Gruppen korrespondieren nicht, die Varianz beider Gruppen ist aber gleich.
- Nicht gepaarte Gruppen mit ungleicher Varianz: Werte der beiden Gruppen korrespondieren nicht, die Varianz beider Gruppen ist aber unterschiedlich.

Die Varianz ist definiert als Durchschnitt der quadrierten Abweichungen der einzelnen Werte einer Gruppe vom arithmetischen Wert aller Werte. Die Quadrierung hat die Folge, dass größere Abweichungen der Werte stärker ins Gewicht fallen, da Quadratzahlen sehr schnell wachsen [Beller, 2004].

Es soll nun untersucht werden welcher Typ des studentischen T-Tests angewendet werden kann. Im Fall des zu untersuchenden Reihenfolgeeffekts gibt es zwei Gruppen (Spracheingabe-Gruppe und Pick-Ray-Gruppe) mit ungepaarten Werten. Es soll nun entschieden werden ob die Varianzen der beiden Gruppen gleich oder unterschiedlich sind. Die Werte der beiden Gruppen bestehen aus der Anzahl der

gewählten Interaktionsmethode (in diesem Fall wurde als Referenz die Spracheingabe gewählt) für jeden Benutzer. Für die komplette Aufgabenstellung wurden 16 Aktionen protokolliert. Der Tabelle 5.4 kann man am grünen Balken die Werte für beide Gruppen entnehmen. User 1 hat z.B. 7 von insgesamt 16 Aktionen mit Spracheingabe ausgeführt.

Um die Varianzen zwischen zwei Gruppen zu vergleichen bietet Microsoft Excel den F-Test an. Dieser Test berechnet die einseitige Wahrscheinlichkeit, dass sich die Varianzen von zwei Gruppen nicht signifikant unterscheiden. Die Parameter für den F-Test sind die einzelnen Werte der zwei Gruppen sowie der Signifikanzwert oder das α-Niveau von 0,05 (5%). Dieser Wert gibt die Wahrscheinlichkeit für einen Fehler an, den man gerade noch akzeptieren würde. Die obere Tabelle 5.3 zeigt die Ergebnisse des F-Tests bezüglich der Werte der beiden Gruppen an. Man kann in Zeile 6 sehen, dass der Wahrscheinlichkeitswert P = 0,29 beträgt. Da dieser Wert größer als der Signifikanzwert von 0,05 ist können wir die H_0-Hypothese akzeptieren.

Ausgehend von diesem Ergebnis wird nun der T-Test für nicht gepaarte Gruppen mit gleicher Varianz angewendet. Als Parameter werden wieder die Werte beider Gruppen angegeben und ein Signifikanzwert von 0,05. Der unteren Tabelle 5.3 ist ein Wahrscheinlichkeitswert P = 0,67 zu entnehmen. Dieser Wert ist größer als der Signifikanzwert von 0,05 und somit kann man auch hier die H_0-Hypothese annehmen und davon ausgehen, dass kein signifikanter Unterschied zwischen den Werten der beiden Gruppen existiert.

Der Grafik in Abbildung 5.3 kann entnommen werden wie oft Spracheingabe insgesamt bei allen Probanden als Interaktion gewählt wurde. Es wird lediglich zwischen der Spracheingabe- und der Pick-Ray-Gruppe unterschieden. Bei der Sprachsteuerung-Gruppe, welche das Training damit begonnen hat, wurden 74,6% aller Befehle über Sprache ausgeführt, während es bei der Pick-Ray-Gruppe 77,4% waren.

In der gleichen Art wurde auch der Einmaleffekt untersucht. Dafür wurden die Probanden aufgefordert die Aufgabenstellung drei Mal hintereinander durchzuführen. Dabei wurden die Werte des ersten Durchgangs mit den Werten des dritten Durchgangs verglichen, um festzustellen ob ein signifikanter Unterschied bei der Interaktionswahl existiert. Für jeden Proband wurde die Anzahl der Aktionen pro Durchgang, die mit Spracheingabe bzw. Pick-Ray-Interaktion ausgeführt wurden, protokolliert. Für die Auswertung wurde die Spracheingabe als Referenz berücksichtigt. Die zwei Gruppen, die hier untersucht wurden sind gepaart, d.h. es handelt sich um abhängige Werte, da der gleiche Test von denselben Probanden drei Mal durchgeführt wurde. Jeder Wert der ersten Gruppe korrespondiert zu einem passenden Wert der zweiten Gruppe. Aus diesem Grund wurde zur Datenanalyse der T-Test für gepaarte Gruppen eingesetzt.

	Erster Versuch	*Dritter Versuch*
Mittelwert	11,5	11,83333333
Varianz	5,1	5,366666667
Beobachtungen	6	6
Pearson Korrelation	0,821922855	
Hypothetische Differenz der Mittelwerte	0	
Freiheitsgrade (df)	5	
t-Statistik	-0,597614305	
P(T<=t) einseitig	0,288065863	
Kritischer t-Wert bei einseitigem t-Test	2,015049176	
P(T<=t) zweiseitig	0,576131726	
Kritischer t-Wert bei zweiseitigem t-Test	2,570577635	

Tabelle 5.5.: *T-Test für gepaarte Gruppen zur Untersuchung des Einmaleffekts.*

Die Tabelle 5.5 zeigt einen Wahrscheinlichkeitswert P = 0,58 an. Dieser liegt über dem Signifikanzwert 0,05 und somit kann hier die H_0-Hypothese weiterhin angenommen werden, dass kein signifikanter Unterschied zwischen den gepaarten Werten der beiden Gruppen existiert.

Aus der Grafik in Abbildung 5.4 geht hervor, wie oft in jedem der drei Durchgänge Spracheingabe als Interaktion gewählt wurde. Es kann eine Tendenz festgestellt werden, dass der Benutzer bei derselben Interaktionswahl bleibt, für die er sich anfangs entschieden hat.

5.3.2. Auswertung des Fragebogens

Auf die Frage wie intuitiv die kombinierte Interaktion ist, hat die Auswertung der Fragebögen Aufschluss gegeben. Die Probanden konnten auf einer fünfstufigen Ratingskala ihre Antwort zu jeder Frage markieren. Für die Berechnung des Mittelwertes und der Standardabweichung wurde 1 für eine positive und 5 für eine negative Antwort eingesetzt. Im Folgenden werden die Fragen und die jeweiligen Interpretationen der Ergebnisse vorgestellt:

1. Wie empfanden Sie die Anwendung von Sprache zur Steuerung des Systems?

- Der Proband konnte in einer fünfstufigen Skala von „Sehr angenehm“ bis „Nicht angenehm“ ankreuzen. Diese Frage soll Auskunft darüber geben, ob der Benutzer Hemmungen hat mit dem System zu „sprechen“. Der Mittelwert der Antworten liegt bei 1,5 während die Standardabweichung 0,52 beträgt. Insofern empfanden alle Benutzer die Steuerung mit Sprache als angenehm.

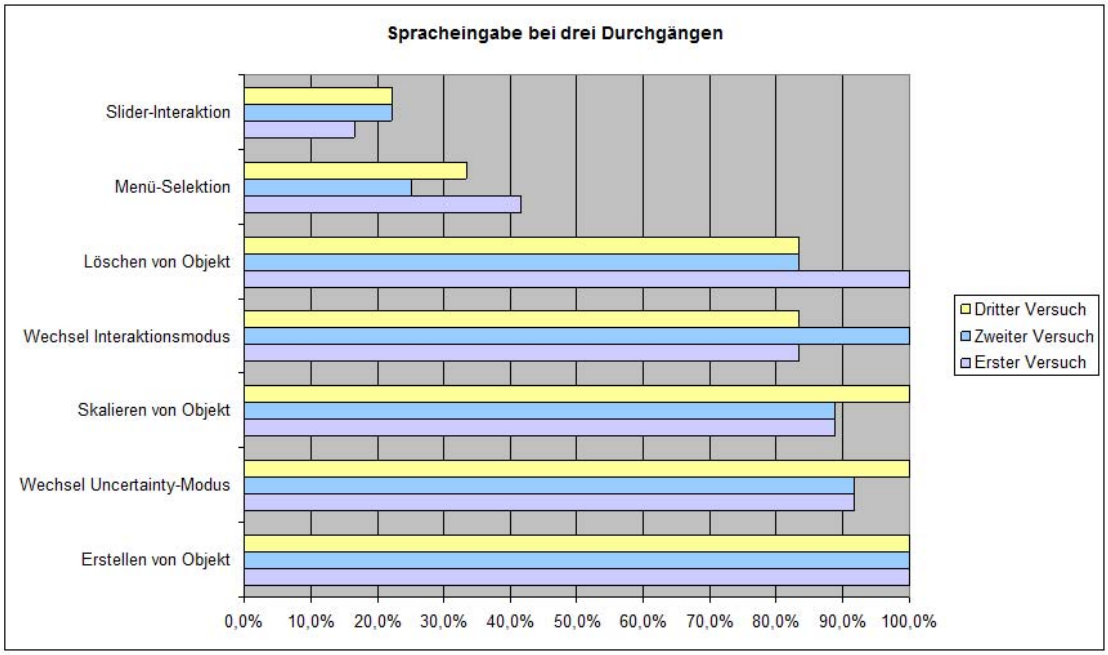


Abbildung 5.4.: *Das Diagramm zeigt die Wahl der Spracheingabe als Interaktionswahl für alle Probanden in drei Durchgängen.*

2. Würden Sie gerne mehr Befehle über Sprache steuern?

- Die positive Antwort ist „Mehr Befehle“ und die negative „Weniger Befehle“. Der Mittelwert liegt bei 2,33 und die Standardabweichung bei 0,65. Das Ergebnis soll zeigen, ob der Benutzer an bestimmten Stellen die Sprachsteuerung vermisst hat. Die Auswertung zeigt, dass die Anzahl der Sprachbefehle optimal war mit einer Tendenz zu „Mehr Befehle“.

3. Würden Sie das System lieber nur über Sprache steuern?

- Die Auswertung zeigt, dass die Benutzer das System weder nur mit Sprache noch komplett ohne Sprache steuern wollten. Der Mittelwert von 2,92 und die Standardabweichung von 0,29 in der Ratingskala von „Nur Sprache“ bis „Ohne Sprache“ ist die Bestätigung der Annahme, dass sich die Probanden nicht nur für eine Interaktionsmöglichkeit entschieden haben, sondern für eine Kombination aus beiden.

4. Würden Sie lieber komplett auf die Sprachsteuerung verzichten?

- Auf der fünfstufigen Skala von „Ja“ bis „Nein“ bestätigt der Mittelwert von 4,67 und die Standardabweichung von 0,65 die Ergebnisse der Protokolle, die

während der Aufgabenstellung aufgenommen wurden. Keiner der Probanden hat gänzlich auf die Sprachsteuerung verzichtet. Das ist wiederholt ein Hinweis auf die Akzeptanz der Sprachsteuerung als Interaktionsmöglichkeit.

5. Geben Sie bitte an, welche Tasks Sie über Sprache bzw. nicht über Sprache steuern würden?

- Bei Aktionen wie das Erstellen, Löschen und Skalieren von Objekten, war Sprachsteuerung die Präferenz der Mehrheit der Benutzer. Die Slider-Interaktion hingegen wurde bevorzugt über Pick-Ray-Interaktion ausgeführt, da sie eine iterative Aktion ist und deswegen angenehmer über den Pick-Ray zu steuern. Dieser Weg ist schneller und der Benutzer bekommt ein direktes Feedback. Die Möglichkeit der Sprachsteuerung ist zwar auch hier gegeben, aber die ständige Wiederholung der Phrasen „value up“, „value down“ etc., um die gewünschte Größe zu erhalten, ist unnatürlich und stößt auf Widerwillen. Dagegen erscheint das Aufrufen von z.B. „create slice“ um eine Schnittebene zu erstellen viel angenehmer und natürlicher.

6. Sind Sie der Meinung, dass die gestellten Aufgaben durch die Sprachsteuerung intuitiver auszuführen waren als ohne Sprachsteuerung?

- Durch diese Frage soll in Erfahrung gebracht werden ob der Proband das Gefühl hatte, das System mit der Sprachsteuerung intuitiv bedienen zu können. Die Ratingskala reichte von „Intuitiv“ bis „Nicht intuitiv“ und die Berechnung des Mittelwertes und der Standardabweichung ergaben 1,83 bzw. 0,72. Daraus kann geschlossen werden, dass die Probanden das Gefühl hatten das System „aus einer Eingebung heraus“ bedienen zu können.

7. Hatten Sie das Gefühl durch die Kombination Sprache/Zeigen „natürlicher“ mit dem System interagieren zu können?

- Eine der Anforderung des Systems war die natürliche Bedienbarkeit. Diese sollte durch die Kombination von zeigebasierter Gestik und Spracheingabe erreicht werden. Der Mittelwert von 1,75 und die Standardabweichung von 0,62 bestätigen die Annahme von einem natürlichem Interaktionskonzept.

8. Ist Ihrer Meinung nach eine Kombination aus Sprache und Zeigen besser zur Anwendungssteuerung geeignet als eine nur zeigebasierte Steuerung?

- Die Probanden sollten darüber entscheiden ob die Steuerung des Systems über die kombinierte Interaktion besser bzw. effektiver war. Der Mittelwert von 1,17 und die Standardabweichung von 0,39 bestätigen diese Annahme.

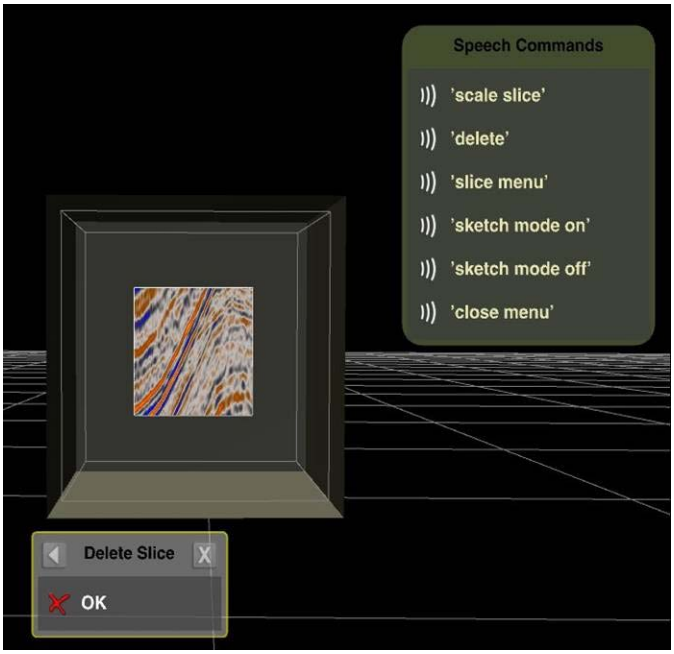


Abbildung 5.5.: *Aktion zum Löschen eines Objektes. Der Benutzer muss den Löschvorgang über die Pick-Ray-Interaktion mit „OK“ bestätigen. Auf dem Hilfe-Panel ist kein Eintrag mit „OK“ für den Sprachbefehl zu sehen.*

Die Auswertung des Fragebogens hat alle Annahmen bestätigt. Zusammengefasst kann man sagen, dass Spracheingabe eine große Akzeptanz bei den Benutzern erreicht hat, dass die Kombination beider Interaktionstechniken besser ist als jede für sich und dass die Steuerung über diese Kombination natürlicher ist.

5.3.3. Weitere Beobachtungen

Bisher wurde festgestellt, wie ein Benutzer reagiert, wenn ihm beide Methoden, Sprachsteuerung und Pick-Ray-Interaktion zur Verfügung stehen und dass er gewillt ist Sprache zu benutzen, wenn er die richtigen Befehle kennt. Aber kann man daraus folgern, dass der Benutzer intuitiv interagiert?

Will man im Demonstrator ein Objekt wie eine Volumenlinse oder eine Schnittebene löschen, muss man dies mit Drücken auf „OK“, das in Form eines Menüs erscheint, mit dem Pick-Ray bestätigen. Wie wir auf Abbildung 5.5 sehen können, ist auf der Liste der gültigen Sprachbefehle kein entsprechender Eintrag aufgeführt, um den Löschvorgang über Sprache zu bestätigen und es war auch keinem Benutzer bewusst, dass man den Sprachbefehl „OK“ anwenden kann. Trotzdem wurden 66% aller Löschvorgänge über Sprache bestätigt. Hier hat der Benutzer begonnen intuitiv mit der Applikation zu interagieren, da es sich um einen natürlichen Vor-

gang handelt, etwas mit „OK“ zu bestätigen. Daraus ist auch zu erkennen, dass der Benutzer anfängt sich auf die Funktionalität der Applikation zu verlassen.

Die Hilfe für die Sprachbefehle sind am Anfang sehr nützlich. Sind dem Benutzer die Befehle geläufig, wird die Hilfe nicht mehr benötigt. Es wurde festgestellt, dass die Probanden dazu tendiert haben nur die ersten 3 oder 4 Einträge zu lesen. Vielen war nicht einmal bewusst, dass die unterschiedlichen Arbeitsbereiche über die Sprache zu öffnen waren, da diese am Ende der Liste standen. Daraus kann geschlossen werden, dass weitaus weniger Sprachbefehle nötig sind.

Von enormer Wichtigkeit ist, dass die Spracherkennung zuverlässig funktioniert, da es für den Benutzer schnell frustrierend werden kann, wenn eine bestimmte Phrase nicht erkannt wird, woraufhin dieser dann zur Pick-Ray-Interaktion wechselt.

5.3.4. Kommentare der Benutzer

Am Ende des Fragebogens war noch Raum für Bemerkungen. Hier konnten die Probanden Anmerkungen und Anregungen zur Interaktion mit dem VRGeo Demonstrator niederschreiben. Im Folgenden werden die Kommentare der Benutzer aufgelistet und interpretiert.

- *Sprachsteuerung ist eine gute Unterstützung, da weniger ermüdende Zeigeoperationen nötig sind (z.B. Submenünavigation durch Sprache meist überflüssig)*
- *Sprache ersetzt mitunter eine Reihe von Schritten: Sketchen unterbrechen, Menü öffnen, Uncertainty Submenu öffnen, Darstellung ändern, Menü schließen (Fünf Schritte ersetzt durch einen Sprachbefehl)*
- *Sprachsteuerung hat sich besonders als Hot-Keys erwiesen, wenn der normale Eintrag in einem Untermenü liegt*

Hier lässt sich die Effektivität der Sprachschnittstelle erkennen. Je tiefer sich eine Funktion in der Menü-Hierarchie befindet, umso praktischer erscheint ein Aufruf über Sprache.

- *Problem: User kann sich während der Sprachsteuerung nicht normal unterhalten*

Dieses Problem wurde durch den Push-To-Talk-Button gelöst, ist aber bei der Benutzerstudie nicht eingesetzt worden. Der Push-to-Talk-Button ist in Umgebungen wichtig, in denen die Geräuschkulisse durch Maschinengeräusche oder durch weitere Personen sehr hoch ist. Vor allem aber in Multi-User Anwendungen wenn ein verbaler Austausch zwischen zwei Benutzern notwendig ist.

- *Sprachkommandos als Liste zu zeigen hilft sehr gut beim Lernen, könnte bei vielen Kommandos aber überflüssig werden*
- *Hilfe-Panels sehr hilfreich, Applikation ist dadurch schneller zu verstehen und zu bedienen*
- *Änderung der verfügbaren Sprachbefehle besser anzeigen z.B. durch Farben*

Durch die Hilfe-Panels ist ein schneller Einstieg in die Sprachsteuerung ermöglicht worden. Der Benutzer konnte direkt Sprachbefehle geben, ohne diese vorher von einem Blatt o.ä. auswendig zu lernen. Das Layout und die Farbe des Hilfe-Panels ändern sich in der momentanen Implementierung nicht in Abhängigkeit mit dem Kontext. Durch die visuelle Änderung könnte allerdings eine benutzerfreundlichere Schnittstelle geschaffen werden, da im Moment die Änderung nicht immer wahrgenommen wird. Vor allem, wenn die Einträge in den Hilfe-Panels die gleiche Anzahl für unterschiedliche Objekte haben.

- *Shutterbrille und Mikrofon sind etwas einengend*
- *Integriertes Mikrofon an der Shutterbrille wäre angenehmer*

Ein generelles Problem in Virtuellen Umgebungen ist, dass der Benutzer mit mehreren Geräten umgehen muss. Das Mikrofon an der Shutterbrille zu befestigen sollte somit in Betracht gezogen werden. Auch Ansteckmikrofone bieten sich an, weil sie nicht direkt am Kopf befestigt werden. Diese benötigen allerdings zusätzliche Hardware (Verstärker, Mischpult).

- *Probleme bei der Nutzung des Mikrofons (zu laut, zu leise), in diesem Fall wurde auf Pick-Ray-Interaktion übergegangen*
- *Sehr störend wenn Sprachbefehl nicht erkannt wird*

Eine gute Qualität der Spracherkennung ist sehr wichtig für die Sprachsteuerung. Hier ist ein gutes Feedback nötig, damit der Benutzer erkennen kann wieso die Applikation nicht auf seine Sprachbefehle reagiert. Das kann unterschiedliche Gründe haben: Der Befehl existiert nicht oder wurde nicht erkannt, das Audiosignal war zu stark oder zu schwach, der Sprachbefehl bezieht sich auf den falschen Kontext oder die Sprachapplikation ist abgestürzt.

- *Das Kontextmenü sollte sich ebenfalls über Sprache öffnen und schließen lassen*

Während der Aufgabenstellung der Benutzerstudie, haben einige Probanden versucht das Kontextmenü mit „open menu“ oder „close menu“ zu öffnen bzw. zu schließen. Dieser Sprachbefehl wurde danach in die Funktionalität des Demonstrators eingebaut.

- *Slider-Interaktion ist mit Pick-Ray-Interaktion besser*

Aus den Protokollen, die während der Aufgabenstellung aufgezeichnet wurden, ist ersichtlich, dass 75 % aller Slider-Interaktionen mittels Pick-Ray-Interaktion durchgeführt worden sind. Skalierung ist eine Art der Manipulation und deswegen scheint es natürlicher zu sein, diese „manuell“ auszuführen.

6. Zusammenfassung

In dieser Diplomarbeit wurde ein multimodales Interaktionskonzept vorgestellt, das Spracheingabe und Zeigeoperation zum Mittel hat. Dabei wurden insbesondere die Anforderungen an die Interaktion mit dem VRGeo Demonstrator berücksichtigt. Das vorgestellte Interaktionskonzept orientiert sich an vorangegangenen Arbeiten zur Interaktion in Virtuellen Umgebungen, die ausführlich in Kapitel 2 besprochen wurden.

Um die vorhandene Benutzerschnittstelle des VRGeo Demonstrator natürlicher und intuitiver zu gestalten, wurde zum vorhandenen System Sprachsteuerung implementiert. Diese ist als ein Expertenmodus zu verstehen, da der Benutzer nicht nur die entsprechenden Sprachbefehle beherrschen muss, sondern auch über die Funktionalität des Systems Kenntnisse haben sollte. Durch die kontextabhängigen Hilfe-Panels, auf welchen die gültigen Sprachbefehle stehen, ist selbst Laien ein schneller Einstieg ermöglicht worden. Zusätzlich zu der Hilfe besteht noch die Möglichkeit, die Einträge aus den Menüs als Sprachbefehle abzusetzen.

Die Zeigeoperation hilft einerseits, Zweideutigkeiten zu eliminieren und andererseits, die Interaktion in dem Demonstrator natürlicher zu gestalten. Der Benutzer kann sich zusätzlich für eine implizite oder explizite Spracheingabe entscheiden. Die explizite bietet die Möglichkeit, sich während der Interaktion mit einer anderen Person zu unterhalten, ohne dass das Spracherkennungssystem darauf reagiert. Es wurde ebenfalls die Benennung von Objekten über Spracheingabe realisiert. Dafür steht eine Grammatik zur Verfügung, die aus einem Expertenvokabular der Öl- und Gasindustrie besteht.

Die Benutzerstudie hat gezeigt, dass Spracheingabe als Interaktion in einem VR-System akzeptiert wird. Indiz dafür war die Tatsache, dass 76% aller Befehle über Sprache ausgeführt wurden. Es hat sich ebenfalls gezeigt, dass die Benutzer vorzugsweise eine Kombination aus beiden Interaktionstechniken - Pick-Ray-Interaktion und Spracheingabe - eingesetzt haben, anstatt sich nur für eine zu entscheiden. Anwender haben während der Aufgabenstellung angefangen, mit der Applikation intuitiv zu interagieren, indem sie den Löschvorgang eines Objektes im Demonstrator mit der Aussage „OK“ bestätigt haben, obwohl dieser Sprachbefehl nicht auf dem Hilfe-Panel angezeigt war. Nur die Tatsache, dass dieser Befehl innerhalb eines Menüs auf einem Button stand und dass Spracheingabe als solche zur Verfügung stand, bewegte die Benutzer zu dieser spontanen Aktion.

Dieses multimodale Interaktionskonzept ist zwar speziell für den VRGeo De-

monstrator entwickelt worden, kann aber leicht auf andere projektionsbasierte VR-Systeme übertragen werden, in denen Spracherkennung und Zeigeoperation möglich ist. Die Ergebnisse der hier vorgestellten Benutzerstudie lassen sich ebenfalls auf andere VR-Anwendungen übertragen, bei welchen Systemsteuerung eine wichtigere Rolle einnimmt als die Manipulation von Objekten.

7. Ausblick

Trotz der vorgestellten Problemlösungen, bleiben für die Zukunft weitere Herausforderungen für die natürliche Interaktion mit VR-Applikationen bestehen. Im Moment sind nur deiktische Gestikfunktionen möglich. Aktuell werden im Rahmen des VRGeo Projektes auch ikonische Gesten erforscht und implementiert. Diese können dazu benutzt werden, Objekte zu manipulieren.

Interessant für die Sprachsteuerung ist ein Multiuser-Ansatz, da diese derzeit lediglich von nur einem Benutzer angewendet werden kann. Mit der Push-to-Talk-Technik und einem Richtmikrofon, welches primär den frontal eintreffenden Schall aufnimmt, kann dies ermöglicht werden.

Der Wortschatz für die Grammatik steht im momentanen System fest. Selbst wenn die semantische Interpretation unterschiedliche Sprachbefehle für dieselbe Funktion zulässt, so ist es doch nicht ohne großen Aufwand möglich für jeden Benutzer alle alternativen Befehle mit einzubeziehen. Es besteht zurzeit die Möglichkeit Grammatiken *on-the-fly* zu erstellen. Um eine adaptive Schnittstelle zu bekommen, könnte dem Benutzer die Möglichkeit gegeben werden, eigene Sprachbefehle in die Grammatik zu integrieren, um so einen individuellen Wortschatz zu erstellen. Im Anschluss daran kann diese Grammatikerweiterung als Profil gespeichert und beliebig ergänzt werden. Ähnlich kann mit den Menüeinträgen verfahren werden. Diese könnten beliebig ausbaufähig sein und für jeden Benutzer in einer Art Interaktionsprofil gespeichert werden können.

Ein verbessertes Feedback für nicht ausgeführte Sprachbefehle kann die Sprachschnittstelle enorm verbessern. Es gibt mehrere Gründe warum ein Sprachbefehl nicht erkannt wird. Dies kann mit einem zu starkem bzw. schwachem Audiosignal zusammenhängen. Ein weiterer Grund kann sein, dass der Sprachbefehl im aktuellen Kontext bzw. generell nicht existiert. In der Benutzerstudie wurde festgestellt, dass wenn ein Sprachbefehl nicht ausgeführt wurde, die Probanden direkt auf die Pick-Ray-Interaktion ausgewichen sind. Der Benutzer sollte daher in jedem Fall ein Feedback über den Status der Sprachschnittstelle bekommen, um dann entsprechend darauf reagieren zu können. Eine Möglichkeit könnte es sein, die Rückmeldung über den Sprach-Synthesizer mittels gesprochenen Texts zu realisieren. Aussagen wie „Bitte, wiederholen sie denn Befehl“, „Dieser Befehl existiert nicht“ oder „Bitte stellen sie den Mikrofonpegel richtig ein“ können durch einen Dialog zwischen Benutzer und Anwendung die Natürlichkeit der Schnittstelle enorm erhöhen.

Hilfe-Panels sind hilfreich und gewährleisten das schnelle Erlernen von Sprachbefehlen. Ein Nachteil ist jedoch, dass diese die VR-Szene zusätzlich überfüllen, da sie im aktivierten Status immer präsent sind. Eine Idee zur Lösung dieses Problems wäre, die Hilfe lediglich anzuzeigen, wenn diese vom Benutzer explizit über Spracheingabe angefordert wird. Z.B. könnte bei dem Ausdruck „slice“ ein Hilfe-Panel mit sämtlichen Befehlen bezüglich der Schnittebene, bzw. alle Sprachbefehle welche „slice“ beinhalten, erscheinen.

Eidesstattliche Erklärung

Hiermit versichere ich, die vorliegende Arbeit selbstständig und unter ausschließlicher Verwendung der angegebenen Literatur und Hilfsmittel erstellt zu haben.

Die Arbeit wurde bisher in gleicher oder ähnlicher Form keiner anderen Prüfungsbehörde vorgelegt und auch nicht veröffentlicht.

Bonn, 18. September 2008 _____

Unterschrift

Literaturverzeichnis

[AKG 2008] AKG: *http://www.akg.com/site/products*. 9. Juli 2008, 14:24

[A.R.T. 2008] A.R.T.: *http://www.ar-tracking.de*. 22. August 2008, 13:26

[Beller 2004] Beller, Sieghard: *Empirisch forschen lernen*. Bern, Switzerland : Verlag Hans Huber, 2004. – ISBN 3456840918

[bhpbilliton 2008] bhpbilliton: *http://www.bhpbilliton.com/bbContentRepository/docs/OurBusiness/Petroleum/SpeakingOilGas.pdf*. 22. August 2008, 14:50

[Billinghurst 1998] Billinghurst, Mark: Put that where? voice and gesture at the graphics interface. In: *SIGGRAPH Comput. Graph.* 32 (1998), Nr. 4, S. 60–63. http://dx.doi.org/http://doi.acm.org/10.1145/307710.307730. – DOI http://doi.acm.org/10.1145/307710.307730. – ISSN 0097–8930

[Bolt 1980] Bolt, Richard A.: "Put-that-there": Voice and gesture at the graphics interface. In: *SIGGRAPH Comput. Graph.* 14 (1980), Nr. 3, S. 262–270. http://dx.doi.org/http://doi.acm.org/10.1145/965105.807503. – DOI http://doi.acm.org/10.1145/965105.807503. – ISSN 0097–8930

[Bowman u. a. 2005] Bowman, Doug A. ; Kruijff, Ernst ; Laviola, Joseph J. ; Poupyrev, Ivan: *3D User Interfaces - Theory And Praxis*. Addison Wesley, 2005

[Bowman u. Wingrave 2001] Bowman, Doug A. ; Wingrave, Chadwick A.: Design and Evaluation of Menu Systems for Immersive Virtual Environments. In: *VR '01: Proceedings of the Virtual Reality 2001 Conference (VR'01)*. Washington, DC, USA : IEEE Computer Society, 2001. – ISBN 0–7695–0948–7, S. 149

[Comvision 2008] Comvision: *http://www.comvision.tv/de/downloads/genie.pdf*. 4. Juli 2008, 5:55

[i Cone 2008] Cone i: *http://www.iais.fraunhofer.de/643.html*. 29. August 2008, 16:18

[Crowley u. Coutaz 1996] Crowley, James L. ; Coutaz, Jöelle: Vision for man machine interaction. In: *Proceedings of the IFIP TC2/WG2.7 Working Conference on Engineering for Human-Computer Interaction*. London, UK, UK : Chapman & Hall, Ltd., 1996. – ISBN 0–412–72180–5, S. 28–45

[Cruz-Neira u. a. 1993] Cruz-Neira, Carolina ; Sandin, Daniel J. ; DeFanti, Thomas A.: Surround-screen projection-based virtual reality: the design and implementation of the CAVE. In: *SIGGRAPH '93: Proceedings of the 20th annual conference on Computer graphics and interactive techniques*. New York, NY, USA : ACM, 1993. – ISBN 0–89791–601–8, S. 135–142

[DECT 2008] DECT: *http://de.wikipedia.org/wiki/DECT*. 9. Juli 2008, 15:25

[Dressler 2007] Dressler, Armin: *Benutzergerechte Menügestaltung für Virtuelle Umgebungen im Expertenkontext*. 2007

[ECMA 2008] ECMA: *http://www.ecma-international.org/publications/files/ECMA-ST/Ecma-262.pdf*. 29. August 2008, 16:55

[Hauptmann u. McAvinney 1993] Hauptmann, Alexander G. ; McAvinney, Paul: Gestures with speech for graphic manipulation. In: *Int. J. Man-Mach. Stud.* 38 (1993), Nr. 2, S. 231–249. http://dx.doi.org/http://dx.doi.org/10.1006/imms.1993.1011. – DOI http://dx.doi.org/10.1006/imms.1993.1011. – ISSN 0020–7373

[Knuth 1964] Knuth, Donald E.: backus normal form vs. Backus Naur form. In: *Commun. ACM* 7 (1964), Nr. 12, S. 735–736. http://dx.doi.org/http://doi.acm.org/10.1145/355588.365140. – DOI http://doi.acm.org/10.1145/355588.365140. – ISSN 0001–0782

[Koons u. Sparrell 1994] Koons, David B. ; Sparrell, Carlton J.: Iconic: speech and depictive gestures at the human-machine interface. In: *CHI '94: Conference companion on Human factors in computing systems*. New York, NY, USA : ACM, 1994. – ISBN 0–89791–651–4, S. 453–454

[Krüger u. a. 1995] Krüger, Wolfgang ; Bohn, Christian-A. ; Fröhlich, Bernd ; Schüth, Heinrich ; Strauss, Wolfgang ; Wesche, Gerold: The Responsive Workbench: A Virtual Work Environment. In: *Computer* 28 (1995), Nr. 7, S. 42–48. http://dx.doi.org/http://dx.doi.org/10.1109/2.391040. – DOI http://dx.doi.org/10.1109/2.391040. – ISSN 0018–9162

[Latoschik u. a. 1998] Latoschik, Marc E. ; Fröhlich, Martin ; Jung, Bernhard ; Wachsmuth, Ipke: Utilize Speech and Gestures to Realize Natural Interaction in a Virtual Environment. In: *Industrial Electronics Society, IECON '98*, 1998, S. 2028–2033

[Malkewitz 1998] Malkewitz, Rainer: Head pointing and speech control as a hands-free interface to desktop computing. In: *Assets '98: Proceedings of the third international ACM conference on Assistive technologies*. New York, NY, USA : ACM, 1998. – ISBN 1–58113–020–1, S. 182–188

[Microsoft 2008] Microsoft: *http://msdn.microsoft.com/en-us/magazine/cc163663.aspx*. 5. Juli 2008, 6:45

[Mine 1995] Mine, Mark R.: Virtual Environment Interaction Techniques. Chapel Hill, NC, USA : University of North Carolina at Chapel Hill, 1995. – Forschungsbericht

[Neal u. Shapiro 1991] Neal, Jeannette G. ; Shapiro, Stuart C.: Intelligent multi-media interface technology. (1991), S. 11–43. http://dx.doi.org/http://doi.acm.org/10.1145/107215.128690. – DOI http://doi.acm.org/10.1145/107215.128690. ISBN 0–201–50305–0

[Novotech 2008] Novotech: *http://www.novotech-gmbh.de*. 4. Juli 2008, 6:46

[Oviatt 1999] Oviatt, Sharon: Ten myths of multimodal interaction. In: *Commun. ACM* 42 (1999), Nr. 11, S. 74–81. http://dx.doi.org/http://doi.acm.org/10.1145/319382.319398. – DOI http://doi.acm.org/10.1145/319382.319398. – ISSN 0001–0782

[Philips 2008] Philips: *http://www.speechrecognition.philips.com/index.asp?file=1644e*. 4.7.2008 2008, 9:06

[Plantronics 2008] Plantronics: *http://www.plantronics.com*. 9. Juli 2008, 14:41

[Schou u. Gardner 2007] Schou, Torben ; Gardner, Henry J.: A Wii remote, a game engine, five sensor bars and a virtual reality theatre. In: *OZCHI '07: Proceedings of the 2007 conference of the computer-human interaction special interest group (CHISIG) of Australia on Computer-human interaction: design: activities, artifacts and environments*. New York, NY, USA : ACM, 2007. – ISBN 978–1–59593–872–5, S. 231–234

[Sherman u. Craig 2002] Sherman, William R. ; Craig, Alan B.: *Understanding Virtual Reality: Interface, Application, and Design*. San Francisco, CA, USA : Morgan Kaufmann Publishers Inc., 2002. – ISBN 1558603530

[Simon u. Dressler 2005] Simon, Andreas ; Dressler, Armin: Interaction and Co-located Collaboration in Large Projection-Based Virtual Environments. In: *Human-Computer Interaction - INTERACT 2005*, 2005, S. 364–376

[Simon u. Göbel 2002] Simon, Andreas ; Göbel, Martin: The i-Cone A Panoramic Display System for Virtual Environments. In: *PG '02: Proceedings of the 10th Pacific Conference on Computer Graphics and Applications*. Washington, DC, USA : IEEE Computer Society, 2002. – ISBN 0–7695–1784–6, S. 3

[Simon u. Scholz 2005] Simon, Andreas ; Scholz, Sascha: Multi-Viewpoint Images for Multi-User Interaction. In: *vr* 00 (2005), S. 107–113. http://dx.doi.org/http://doi.ieeecomputersociety.org/10.1109/VR.2005.55. – DOI http://doi.ieeecomputersociety.org/10.1109/VR.2005.55. – ISSN 1087–8270

[SISR 2008] SISR: *http://www.w3.org/TR/semantic-interpretation/*. 16. Juli 2008, 14:55

[SRGS 2008] SRGS: *http://www.w3.org/TR/speech-grammar/*. 16. Juli 2008, 13:57

[Thorisson u. a. 1992] Thorisson, Kristinn R. ; Koons, David B. ; Bolt, Richard A.: Multi-modal natural dialogue. In: *CHI '92: Proceedings of the SIGCHI conference on Human factors in computing systems*. New York, NY, USA : ACM, 1992. – ISBN 0–89791–513–5, S. 653–654

[Tramberend 1999] Tramberend, Henrik: Avocado: A Distributed Virtual Reality Framework. In: *vr* 00 (1999), S. 14. `http://dx.doi.org/http://doi.ieeecomputersociety.org/10.1109/VR.1999.756918`. – DOI http://doi.ieeecomputersociety.org/10.1109/VR.1999.756918. – ISSN 1087–8270

[TwoView 2008] TwoView: *http://www.iais.fraunhofer.de/645.html*. 29. August 2008, 16:15

[VRGeo 2008] VRGeo: *http://www.vrgeo.com*. 24. Juli 2008, 11:55

[Weimer u. Ganapathy 1989] Weimer, D. ; Ganapathy, S. K.: A synthetic visual environment with hand gesturing and voice input. In: *CHI '89: Proceedings of the SIGCHI conference on Human factors in computing systems*. New York, NY, USA : ACM, 1989. – ISBN 0–89791–301–9, S. 235–240

[Wesche 2004] Wesche, Gerold: *Conceptual Free-Form Styling in Virtual Environments*, Universität des Saarlandes, Diss., 2004

[XML 2008] XML: *http://www.w3.org/XML/*. 29. August 2008, 15:41

[xovox 2008] xovox: *http://www.ieco.de/downloads/xcommunicatorbrochure.pdf*. 9. Juli 2008, 14:25

[Zhai u. Milgram 1998] Zhai, Shumin ; Milgram, Paul: Quantifying coordination in multiple DOF movement and its application to evaluating 6 DOF input devices. In: *CHI '98: Proceedings of the SIGCHI conference on Human factors in computing systems*. New York, NY, USA : ACM Press/Addison-Wesley Publishing Co., 1998. – ISBN 0–201–30987–4, S. 320–327

[Zudilova 2002] Zudilova, E. V.: A Multi-Modal Interface for an Interactive Simulated Vascular Reconstruction System. In: *ICMI '02: Proceedings of the 4th IEEE International Conference on Multimodal Interfaces*. Washington, DC, USA : IEEE Computer Society, 2002. – ISBN 0–7695–1834–6, S. 313

A. Anhang

Benutzer Nr. Alter: Datum:

Fragebogen

1) Wie empfanden Sie die Anwendung von Sprache, zur Steuerung des Systems?

◯ ---------- ◯ ---------- ◯ ---------- ◯ ---------- ◯

Sehr angenehm Nicht angenehm

2) Wuerden Sie gerne mehr Befehle ueber Sprache steuern?

◯ ---------- ◯ ---------- ◯ ---------- ◯ ---------- ◯

Mehr Befehle Weniger Befehle

3) Wuerden Sie das System lieber nur mit Sprache steuern?

◯ ---------- ◯ ---------- ◯ ---------- ◯ ---------- ◯

Nur Sprache Ohne Sprache

4) Wuerden Sie lieber komplett auf die Sprachsteuerung verzichten?

◯ ---------- ◯ ---------- ◯ ---------- ◯ ---------- ◯

Ja Nein

5) Geben Sie bitte an, welche Tasks Sie ueber Sprache bzw. nicht ueber Sprache steuern wuerden (Mehrfachwahl erlaubt)?

- Objekt erstellen ◯ Ja ◯ Nein
- Objekt loeschen ◯ Ja ◯ Nein
- Objekt skalieren ◯ Ja ◯ Nein
- Slider Interaktion ◯ Ja ◯ Nein
- Sketch Modus aendern ◯ Ja ◯ Nein

Benutzer Nr. Alter: Datum:

6) Sind Sie der Meinung, dass die gestellten Aufgaben durch die Sprachsteuerung intuitiver auszufuehren waren als ohne Sprachsteuerung?

○ ---------- ○ ---------- ○ ---------- ○ ---------- ○

Intuitiv Nicht intuitiv

7) Hatten Sie das Gefuehl, durch die Kombination Sprache/Zeigen, 'natuerlicher' mit dem System interagieren zu koennen?

○ ---------- ○ ---------- ○ ---------- ○ ---------- ○

Natuerlich Nicht natuerlich

8) Ist Ihrer Meinung nach eine Kombination aus Sprache und Zeigen besser geeignet zur Anwendungssteuerung als eine nur zeigebasierte Steuerung?

○ ---------- ○ ---------- ○ ---------- ○ ---------- ○

Besser Schlechter

9) Wie schaetzen Sie Ihre Erfahrung in folgendem Bereich ein:

- Ich benutze VR Applikationen

○ ---------- ○ ---------- ○ ---------- ○ ---------- ○

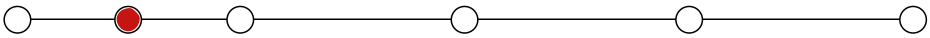
taeglich oft regelmaeßig selten nie

10) Hier haben Sie noch Platz fuer weitere Anmerkungen und Anregungen: Hatten Sie Probleme mit der Nutzung des Mikrofons? Empfanden Sie das Mikrofon als stoerend? Haben Sie Verbesserungsvorschlaege? Was fanden Sie besonders gut oder besonders schlecht? War die Trainingszeit ausreichend?

Auswertung des Fragebogens

Die roten Kreise und Zahlen darunter geben den Mittelwert der Ergebnisse wider und die Zahl in der Klammer stellt die Standardabweichung dar.

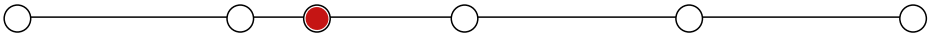
1) Wie empfanden Sie die Anwendung von Sprache, zur Steuerung des Systems?



Sehr angenehm — Nicht angenehm

Mittelwert (Standardabweichung) : 1,5 (0,52)

2) Wuerden Sie gerne mehr Befehle ueber Sprache steuern?



Mehr Befehle — Weniger Befehle

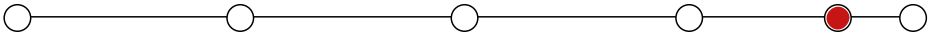
Mittelwert (Standardabweichung) : 2,33 (0,65)

3) Wuerden Sie das System lieber nur mit Sprache steuern?

Nur Sprache — Ohne Sprache

Mittelwert (Standardabweichung) : 2,92 (0,29)

4) Wuerden Sie lieber komplett auf die Sprachsteuerung verzichten?



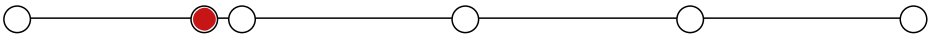
Ja — Nein

Mittelwert (Standardabweichung) : 4,67 (0,65)

5) Geben Sie bitte an, welche Tasks Sie ueber Sprache bzw. nicht ueber Sprache steuern wuerden (Mehrfachwahl erlaubt)?

- Objekt erstellen	Ja		Nein	1,00 (0,00)
- Objekt loeschen	Ja		Nein	1,17 (0.39)
- Objekt skalieren	Ja		Nein	1,08 (0,29)
- Slider Interaktion	Ja		Nein	1,67 (0,49)
- Sketch Modus aendern	Ja		Nein	1,33 (0,49)

6) Sind Sie der Meinung, dass die gestellten Aufgaben durch die Sprachsteuerung intuitiver auszufuehren waren als ohne Sprachsteuerung?



Intuitiv — Nicht intuitiv

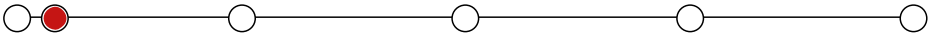
Mittelwert (Standardabweichung) : 1,83 (0,72)

7) Hatten Sie das Gefuehl, durch die Kombination Sprache/Zeigen, 'natuerlicher' mit dem System interagieren zu koennen?

Natuerlich — Nicht natuerlich

Mittelwert (Standardabweichung) : 1,75 (0,62)

8) Ist Ihrer Meinung nach eine Kombination aus Sprache und Zeigen besser geeignet zur Anwendungssteuerung als eine nur zeigebasierte Steuerung?



Besser — Schlechter

Mittelwert (Standardabweichung) : 1,17 (0,39)

9) Wie schaetzen Sie Ihre Erfahrung in folgendem Bereich ein:

- Ich benutze VR Applikationen

haeufig — nie

Mittelwert (Standardabweichung) : 2,83 (1,53)