



Eine mobile multi-user Mixed Reality Umgebung

Dr. Wolfgang Broll
Fraunhofer FIT
Schloss Birlinghoven
53754 Sankt Augustin

Dr. Thomas Müller
Fraunhofer IITB
Fraunhoferstr. 1
76131 Karlsruhe

Dr. Stefan M. Grünvogel
Laboratory for Mixed Realities
Am Coloneum 1,
50829 Köln

Dr. Hans Krüger
Physikalisches Institut Bonn
Nussallee 12
53115 Bonn

Martin Maercker
cutup GmbH
Hospeltstraße 39
50825 Köln

Frank Rohlfs
Theatertechnische Systeme
Siemensstraße 16-18
28857 Syke

Kurzfassung

Ziel des mqube Vorhabens ist die Entwicklung einer mobilen Mixed Reality Umgebung für mehrere Benutzer zur Unterstützung kooperativer Planungsprozesse. Dabei soll es einer Gruppe von Benutzern unter Beibehaltung ihrer gewohnten Kooperations- und Interaktionsmechanismen ermöglicht werden, komplexe Planungs- und Simulationsaufgaben unter Ausnutzung neuer Technologien der Computervisualisierung durchzuführen. Hierzu werden virtuelle Objekte, Charaktere oder Daten unter Zuhilfenahme von Augmented Reality Technologien direkt in das reale Arbeitsumfeld der Gruppe eingeblendet. Durch die Kombination realer Platzhalterobjekte mit virtuellen Artefakten zu einer symbiotischen Interaktionseinheit wird dabei eine neue Form einer greifbaren Benutzungsschnittstelle geschaffen. Diese ermöglicht es, die gewohnte Interaktion mit realen Gegenständen auf die Interaktion mit virtuellen Objekten zu übertragen. Die Einsatzmöglichkeiten des mqube-Systems werden an einer miniaturisierten Modellbühne (Mixed Reality Stage) zur interaktiven Veranstaltungs- und Bühnenplanung demonstriert. Zu den weiteren Anwendungsfeldern, die sich mit diesem System erschließen gehören die Planung von Großereignissen, Fernseh- und Bühnenshows, Produktpräsentationen, Messeauftritten, Bauvorhaben und Städteplanung.

1. Einleitung

Gesamtziel dieses Projektes ist die Entwicklung einer mobilen Multi-User Mixed Reality Umgebung. Unter dieser Bezeichnung ist die Entwicklung und Realisation eines grundlegend neuen Konzeptes zur computerbasierten Unterstützung kreativer Teamarbeit zu verstehen. Solche Teamarbeit ist charakteristisch für die künstlerische und technische Inszenierungs- und Planungsarbeit an Theatern und anderen Veranstaltungsstätten für szenische Darstellungen. Mit Hilfe des mqube-Systems wird es zukünftig möglich sein, computergenerierte Objekte sowohl als Abbild realer bühnentechnischer Maschinen und Einrichtungen, als auch stellvertretend für reale Darsteller in den realen Arbeitsbereich einer Gruppe zu integrieren und diese über die real vorhandenen Originalbedienelemente zu manipulieren. Auch technisch nicht versierten Personen wie beispielsweise den an den theaterspezifischen Produktionsprozessen beteiligten Künstlern und Regisseuren wird die Integration in ein mit modernsten Computertechnologien arbeitendes Team ermöglicht. Dabei wird aufgrund der Integration

realer und virtueller Beleuchtungseffekte eine Planung unter Berücksichtigung realistischer Lichteffekte möglich.

2. Die Mixed Reality Stage

Die Leistungsfähigkeit des gewählten Ansatzes wird durch das Applikationsszenario der *Mixed Reality Stage* realisiert. Bei der Mixed Reality Stage wird eine reale Modellbühne um virtuelle Komponenten wie Bühnenaufbauten, Prospekte oder virtuelle Akteure erweitert, welche die Grundlage für eine interaktive und kooperative Planung der Beleuchtung und Choreographie ermöglichen (s. Abbildung 1). Die Einzelkomponenten des Systems werden im Folgenden kurz beschrieben und in ihrem Zusammenwirken umrissen.

2.1. 3D-Augmentierung und AR-Framework

Die Realisierung der Mixed Reality Stage erfolgt dabei auf Basis von Augmented Reality, das heißt dem Betrachter werden virtuelle Objekte in sein persönliches Arbeitsumfeld eingeblendet. Dabei wird für den Betrachter die Illusion erzeugt, die virtuellen Objekte würden sich im realen Umfeld befinden. Dies erfordert eine perspektivegerechte stereoskopische 3D Darstellung in Abhängigkeit des Blickpunktes und der Blickrichtung des Betrachters. Um ein feste Verortung der virtuellen Objekte im realen Raum zu gewährleisten, muss sowohl die 3D Visualisierung als auch die Anpassung der Szene an den Betrachterblickpunkt dabei in Echtzeit erfolgen. Zur Projektion kommen dabei semitransparente Personal Displays zum Einsatz. Im Gegensatz zur weitverbreiteten Video-Augmentierung, bei der im Display des Betrachters das von einer Videokamera aufgenommene Bild mit den virtuellen Objekten kombiniert wird, wird hierbei lediglich das virtuelle Bild in das Display projiziert. Die reale Umgebung wird hingegen durch den Betrachter direkt wahrgenommen. Der Grund dafür liegt in der Wahrnehmung der Lichtverläufe durch die realen Bühnenscheinwerfer. Während semitransparente Displays hier lediglich zu einer Abdunkelung führen, gehen bei der Aufnahme über herkömmliche Videokameras aufgrund des eingeschränkten Dynamikbereichs für den Anwender entscheidende Informationen verloren.



Abbildung 1: Mixed Reality Stage

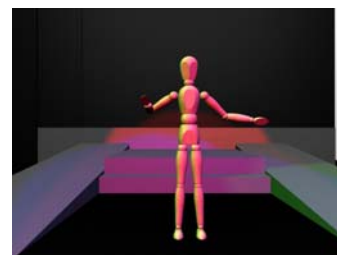


Abbildung 2: Simulation komplexer Beleuchtungssituationen

Im Anwendungsszenario der Mixed Reality Stage ergeben sich aufgrund der Vielzahl von Lichtquellen besondere Anforderungen an die 3D Visualisierung. Reale Bühnen und damit auch die sie repräsentierenden Modellbühnen haben häufig zwischen 50 und 200 Scheinwerfer. Um eine Verschmelzung der realen Bühnengegenstände mit den virtuellen Bühnenobjekten zu erzielen, ist eine adäquate Berücksichtigung der Bühnenbeleuchtung unabdingbar. Existierende Graphik-Hardware unterstützt jedoch in der Regel lediglich acht parallele Lichtquellen, Spezial-Hardware bis zu 24 Lichtquellen. Aus diesem Grund wurden spezielle Mechanismen zur Berechnung und Simulation der akkumulierten Auswirkung aktiver Lichtquellen auf einzelne virtuelle Objekte untersucht und in die 3D Darstellung integriert (s. Abbildung 2).

Zur Kooperationsunterstützung mehrerer Benutzer mit einer jeweils individuellen Visualisierung sind die Synchronisation einzelner Szenengraphen sowie eine flexible, netzwerkbasierte Anbindung der einzelnen Eingabe- und Tracking-Geräte erforderlich. Im Rahmen des Vorhabens wird daher derzeit ein verteiltes AR-Framework entwickelt.

2.2. Licht- und Bühnentechnik

Ein wichtiger Aspekt des Vorhabens ist darüber hinaus die Anbindung vorhandener Licht und Bühnensteuerungssysteme an die Augmented Reality Umgebungen der Modellbühne. Dabei erfolgt eine Integration der bühnentechnischen Steuerungsmodule, so dass sie die Kontrolle beweglicher Inszenierungs-Objekte im virtuellen Bühnenszenario unter Verwendung eines professionellen Steuerungssystems für bühnentechnische Antriebe ermöglichen. Reale Bedienpulte, die mit Fahrbefehlsgebern und berührungssensitiven Monitoren (Touch-Screens) ausgestattet sind, sollen dabei als primäre Steuerungseinheiten genutzt werden (Abbildung 3). Die in der realen Bühnentechnik vorhandenen Leit- und Achsrechner werden anhand von Simulationsprogrammen in der Form nachgebildet, dass eine virtuelle Bewegung der Dekorationsteile und Bühnenbilder komplett in den Displays der am Prozess beteiligten Künstlern und Regisseuren erscheint.

Im Bereich der professionellen Lichtsteuerung werden die Lichtsteuerpulte direkt mit dem mcube-System verbunden (Abbildung 4). Auf diese Weise wird eine parallele Ansteuerung der realen und virtuellen Bühnenscheinwerfer über dieses Pult ermöglicht. Andererseits können Änderungen aus der virtuellen Welt direkt zurück an Steuerpult und damit an die realen Bühnenscheinwerfer erfolgen. Die Anbindung geschieht dabei über DMX und MIDI.



Abbildung 3: Steuerpult vor realen Bühnenelementen: Laststangen

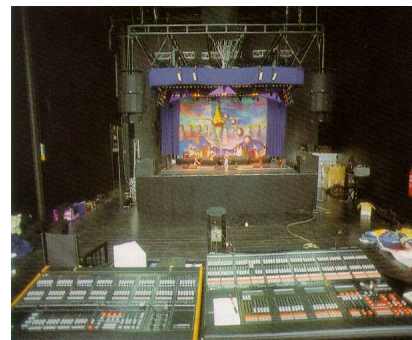


Abbildung 4: Modellbühne im Maßstab 1:4 mit Lichtsteuerpult

2.3. Head Tracking

Um virtuelle Objekte perspektivisch richtig in das Blickfeld der Anwender einblenden zu können, wird ein System benötigt, welches die Position und Blickrichtung (Kopf-Orientierung) der Benutzer der Mixed Reality Umgebung ermittelt. Der Eindruck der Verschmelzung von virtueller- und realer Welt ist dabei entscheidend von der Genauigkeit der Überlagerung und der Reaktionszeit des Systems bei einem sich bewegenden Benutzer abhängig. Weitere wesentliche Anforderungen an das Head-Tracking System für mqube sind Mehrbenutzerfähigkeit und Mobilität.

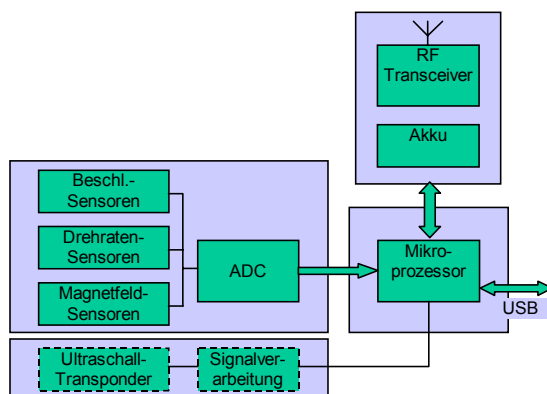


Abbildung 5: Blockschaltbild

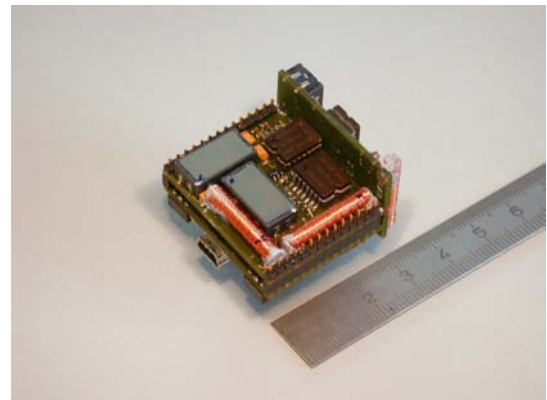


Abbildung 6: Inertial-Tracker Prototyp

Für das Motion-Tracking haben sich verschiedene Konzepte etabliert. Videobasierte Tracking-Systeme benutzen ein oder mehrere Videokameras, um mit einer entsprechenden Bildverarbeitung die Position und Orientierung von Objekten erkennen und verfolgen zu können. Ein anderes Konzept beruht auf einem künstlichen Magnetfeld, welches in einem begrenzten Raumvolumen ein Referenzsystem erzeugt, innerhalb dessen entsprechende Sensoren ihre Position und Orientierung messen können. Ein drittes Konzept benutzt Inertialsensoren zur Lagebestimmung und die über Signallaufzeiten (Ultraschall) gemessenen Abstände zu einem Referenzsystem zur Bestimmung der absoluten Position.

Der Ansatz des Inertialtrackings wird den Anforderungen nach Mobilität und Skalierbarkeit am ehesten gerecht. Der vom Silizium Labor der Universität Bonn entwickelte Tracker basiert daher auf modernen, integrierten Beschleunigungs- und Drehratensensoren zur Bestimmung der räumlichen Orientierung. Die Komponenten sind auf einer miniaturisierten, autarken Einheit integriert (Abbildung 5). Die Messwerte der Sensoren werden auf dem Tracker digitalisiert, im Mikroprozessor verarbeitet und über eine USB-Schnittstelle zum Hostrechner übertragen. Alternativ zu dieser drahtgebundenen Datenübertragung können die Messwerte über eine Funkschnittstelle (433 MHz Band oder Bluetooth) versendet werden. Das Referenzsystem, mit dem der Inertial-Tracker in der nächsten Entwicklungsstufe erweitert wird, nutzt Ultraschallsender und Empfänger zur absoluten Positionierung und arbeitet ebenfalls drahtlos. Dadurch wird ein hohes Maß an Mobilität und Skalierbarkeit des Trackingsystems erreicht. Sowohl die Lage- wie auch die Positionsbestimmung wird aus den Rohdaten mit Hilfe eines Kalman Filters berechnet. Dies gewährleistet eine robuste und fehlertolerante Datenverarbeitung, und ermöglicht eine Minimierung der für den Benutzer wahrnehmbaren Latenz in der Bilddarstellung.

2.4. Object Tracking

Damit virtuelle Objekte mit Hilfe realer Gegenstände in der Szene möglichst einfach und intuitiv bewegt werden können, müssen Position und Orientierung der realen Objekte mittels einer Kamera bezüglich der Szene geschätzt werden. Diese Aufgabe übernimmt das am Fraunhofer Institut IITB entwickelte Objekt-Tracking, welches fortlaufend die insgesamt sechs lagebestimmenden Parameter eines Objekts ermittelt und zur Weiterverarbeitung an die Zentralkomponente des mqube-Systems transferiert.

Zur Einrichtung der Tracking-Komponente ist die verwendete Kamera intern und extern zu kalibrieren. Die *interne Kalibrierung* ist essentiell für das Tracking und braucht lediglich bei einer Veränderung der Abbildungseigenschaften des Sensors (Verwendung einer neuen Kamera, Änderung der Kameraregler) durchgeführt zu werden. Die Durchführung geschieht sehr einfach mittels einer speziellen, mit bestimmten Marken beklebten Kalibrierplatte, welche in verschiedenen Drehlagen vor die Kamera gehalten wird. Die *externe Kalibrierung* dient als gemeinsames Bezugssystem von Zentral- und Tracking-Komponente und abstrahiert im Gesamtsystem von der Kameraposition in der Szene. Die Durchführung geschieht ähnlich wie bei der internen Kalibrierung mittels einer mit Marken beklebten Kalibrierplatte. Die Platte wird auf den Bühnentisch gelegt, ein Bild aufgenommen und daraus automatisch die externe Kalibrierung berechnet. Die Lage der Platte definiert dabei das zu verwendende Szenenkoordinatensystem. Bei Bedarf kann das Koordinatensystem durch Verwendung weiterer Plattenpositionen auf dem Tisch genau auf die Bühnenebene abgestimmt werden.

Zur echtzeitfähigen *Objektverfolgung* steht ein Verfahren zur Verfügung, welches mit 25 Hz die Lagedaten eines Würfels anhand einer aufgeklebten *Marke* ermittelt (s. Abbildung 7). Beim Einbringen des Würfels in den Sichtbereich der Kamera oder nach einer Vollverdeckung wird das Objekt automatisch erkannt, lokalisiert und die Verfolgung (re-)initialisiert. Außerdem wurde eine modellbasierte Verfolgung entwickelt, die mit *Bildkonturen* arbeitet (s. Abbildung 8). Das Verfahren ist in relativ hohem Maße robust gegenüber partiellen Objektverdeckungen, komplexem Szenenhintergrund, ungünstigen Beleuchtungsbedingungen, Glanzlichtern, Schattenwürfen sowie Bühnennebel, arbeitet dafür allerdings noch nicht in Echtzeit.

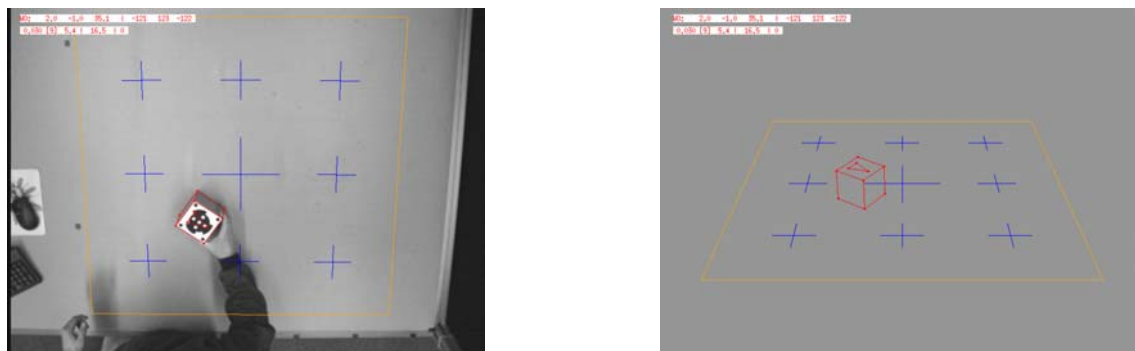


Abbildung 7: Markenbasierte Verfolgung eines Würfels. Die Lageschätzung ist mit roter Farbe visualisiert, das Szenenkoordinatensystem in Form einer Szenentischebene mit gelb und blau (links: Darstellung bezüglich Kamerabild; rechts: bezüglich einer virtuellen Ansicht).

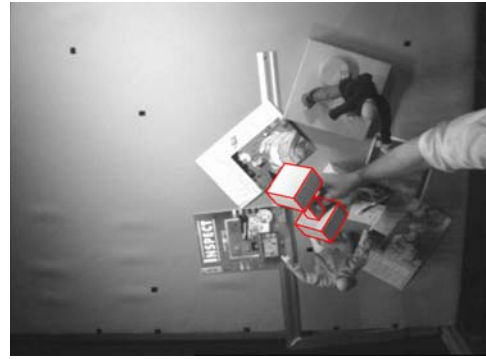
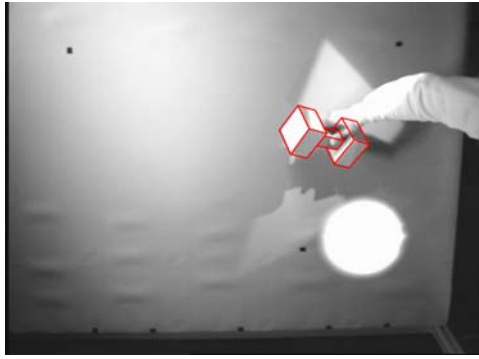


Abbildung 8: Konturbasierte Verfolgung eines hantelförmigen Objekts bei Scheinwerferlicht und partieller Verdeckung durch die Hand (links) sowie vor komplexem Hintergrund (rechts). Die Lageschätzung ist mit roter Farbe visualisiert.

2.5. Character-Animation

Nicht nur das Zusammenspiel zwischen Bühnen- und Lichttechnik muss für eine Inszenierung miteinander abgestimmt werden, auch die Wirkung dieser dramaturgischen Mittel auf die Gestalt der Künstler ist von entscheidender Bedeutung. Bisher wurden für die Darstellung von Akteuren auf der Probebühne statische Gliederpuppen verwendet, die sich natürlich nicht selbstständig bewegen konnten. Durch den Einsatz von virtuellen Charakteren wird es möglich, die Bewegung realer Akteure auf der Bühne im Zusammenspiel von Licht und Bühnenaufbauten zu simulieren.

Ziel ist es hierbei, den Benutzern von mqube eine möglichst einfache Möglichkeit an die Hand zu geben, virtuelle Akteure auf der Bühne zu steuern. So soll der Character etwa einen vom Benutzer festgelegten Pfad auf der Bühne entlang gehen, wobei der Bewegungsstil angepasst und andere Bewegungen überblendet werden können. Die vom Institut Laboratory for Mixed Realities entwickelte Animation-Engine zur Animation von Charakteren stellt hierfür Bewegungen in Form von Motion Models dar. Dies sind eigenständige Bewegungen wie z.B. Gehen, Stehen, Springen, Winken. Da jede menschliche Bewegung ihre eigenen speziellen Aufgaben besitzt, verfügt jedes Motion Model über einen eigenen Satz von Parametern, die die Bewegung beschreiben. So besitzt das Motion Model *Gehen* etwa die Parameter *Geschwindigkeit* und *Stil*. Aus diesen Parametern erzeugt jedes Motion Model individuell die entsprechende Bewegung, in dem es kurze Animation mit einer Reihe von Bewegungsoperatoren vermischt und manipuliert. In Abbildung 9 wird am Beispiel des Motion Models *Gehen* gezeigt, wie aus drei Grundbewegungen (Start Walk, Walk Cycle, Stop Walk) mit den Operatoren *Blend* (Mischen von Animationen), *Loop* (endlicher oder unendliche Wiederholung) und *TimeShift* (zeitlicher Versatz) die Animation erzeugt wird. Die Wurzel des Operatorbaumes bildet die daraus resultierende Bewegung. Wird während der Ausführung eines Motion Models ein Parameter geändert (z.B. Erhöhung der Geschwindigkeit beim *Gehen*), so spiegelt sich dies in der Veränderung des entsprechenden Operatorbaumes wider.

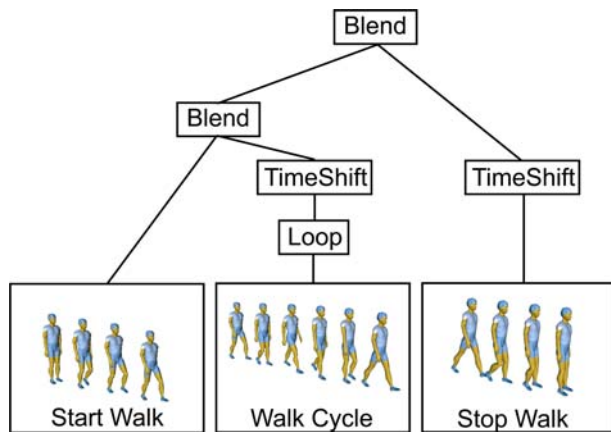


Abbildung 9: Operatorbaum eines Motion Models

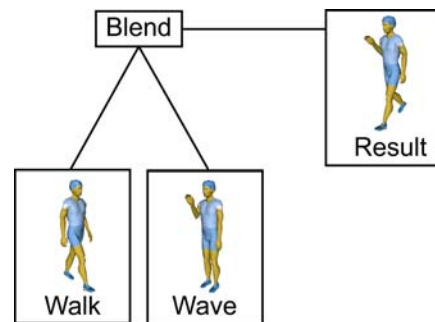


Abbildung 10: Mischen von Motion Models

Um gleichzeitig unterschiedliche Bewegungen auszuführen (z.B. *Winken* und *Gehen*) werden die von den Motion Models resultierenden Operatorbäume wiederum mit den Bewegungs-Operatoren miteinander kombiniert (vgl. Abbildung 10). Da gewisse Bewegungen nicht miteinander kombinierbar sind (etwa gehen und stehen) entscheidet die Reihenfolge und die Priorität eines Motion Models darüber, ob es verzögert oder gar nicht ausgeführt wird.

2.6. User Interface und API

Das von der cutup GmbH entwickelte User Interface ist das katalytische Element, durch das sich für den Anwender das Potenzial eines Zusammenspiels der verschiedenen Einzelkomponenten von mqube erschließt.

Mit einem einzigen atomaren Befehl verbindet der User ein von der Objektverfolgung erfasstes reales Objekt ("Realoid") mit einem virtuellen Objekt ("Virtualoid") zu einer "Interaction Unit". Dieser "Enslavement" genannte Vorgang macht einen der wichtigsten Vorzüge von mqube greifbar: die Ausnutzung der robusten Analogie zwischen räumlichen Operationen im realen und im virtuellen Raum. Auf dieser Grundlage entstehen bspw. Animationspfade, indem mqube aufzeichnet, wie der User eine Interaction Unit durch den Raum bewegt (Abbildung 11).

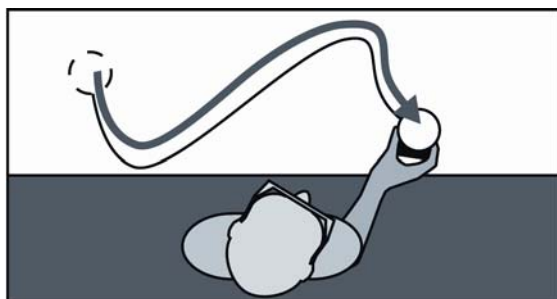


Abbildung 11: Aufzeichnung von Bewegungspfaden durch Interaction Units

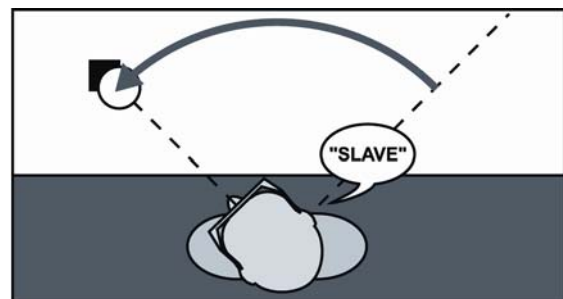


Abbildung 12: Befehlseingabe über Viewpointer und Sprachsteuerung

Um ein Enslavement herbeizuführen, platziert der User einen Realoid in der Nähe eines Virtualoids, schaut den Virtualoid an und sagt das Wort "Slave". Das System erkennt, welcher Virtua-

loid gemeint ist, indem es die Kopf- und gegebenenfalls Augenbewegungen des Anwenders analysiert ("Viewpointer"). Der eigentliche Befehlsimpuls wird über eine Sprachsteuerung abgegeben, die mit einem extrem eingeschränkten und damit leicht erlernbaren Befehlssatz arbeitet (Abbildung 12). So bestimmt der User über Enslavement-Verhältnisse, ohne dazu Realoids, mit denen er eventuell arbeitet, loslassen zu müssen.

Die Dimensionalität von Listen, Hierarchien, Graphen, usw., (eingesetzt bspw. für Parameterausprägungen) ist nicht primär raumzeitlich determiniert. Solche Strukturen soll man möglichst auf einen Blick erfassen und schnell und verlässlich bearbeiten können. Hierzu wurden verschiedene, freihändig bedienbare Menü-Lösungen entworfen. Sie sind auf den Blick des Users ausgerichtet, so dass dieser seine Stellung während der Benutzung nicht ändern muss.

Die Bedarfsanalyse im Umfeld der theatertechnischen Vorproduktion ergab eine Reihe sowohl expliziter als auch impliziter Anforderungen an das User Interface. Zudem soll mcube auch in anderen Bereichen (z.B. Architektur, Städtebau) einsetzbar sein. Daher wurden grundsätzliche und generische Lösungen erarbeitet zu Themen wie Gruppierung, Befehlsverkettungen zwischen Virtualoids und Sequenzierung (Zeitschienenverwaltung).

Eine objektorientierte API, die strukturell an der User Interface-Architektur orientiert ist, macht die verschiedenen Interface-Komponenten als einheitliches Subsystem von mcube für Anwendungsentwickler zugänglich. Darüber hinaus bietet sie einen Rahmen zur Integration anwendungsspezifischer Inhalte.

3. Erfahrungen und Bewertungen

Der derzeitige Stand des Vorhabens ermöglicht zurzeit noch keine Bewertung der Erfahrungen zukünftiger Anwender. Bei den durchgeführten Befragungen unter potentiellen Anwendern aus der Event- und Theaterbranche fand ein solches System jedoch starkes Interesse.

Bezüglich der Verwertung des Systems werden unterschiedliche Strategien angewandt werden. Zum einen ist es angestrebt, den endgültigen Prototypen der Mixed Reality Stage in ein Produkt zu überführen und als solches nach Möglichkeit gemeinschaftlich zu verwerten. Darüber hinaus stellt das entwickelte System die Basis für eine Vielzahl anderer Anwendungsgebiete dar. Dabei kann eine Verwertung sowohl kommerziell als auch wissenschaftlich verfolgt werden. Schließlich ist darüber hinaus die Verwertung der entwickelten Einzelkomponenten (AR System, Benutzer-Tracking, Objekt-Tracking, Charakter-Animation) im Rahmen anderer Produkte oder Forschungsprojekte möglich.

4. Ausblick

Seitens des AR Systems wird in Kürze die Unterstützung mehrerer Benutzer möglich sein, so dass unterschiedliche Kooperationstechniken erprobt werden können. Mit der Realisierung des ersten Prototypen eines kabellosen 6-DOF Head Tracking Systems, werden dann mehrere Benutzer erstmals gemeinsam interaktiv Manipulationen an der Mixed Reality Stage vornehmen und die Choreographien planen können. Dabei wird es aufgrund neuer User Interface Techniken basierend auf Spracheingabe und kamerabasierten Objektverfolgungstechniken möglich sein, trotz eines verschwindenden Computers, die Vorteile einer direkten Verbindung aller Komponenten einschließlich der Licht- und Bühnensteuerung zu nutzen.