



Adaptive Dynamic Visual Semantic Organizing

## Abschlussbericht

**Förderkennzeichen**

01 | WD01

**Projektleitung**

Prof. Dr. Hans Hagen / Prof. Dr. Andreas Dengel

**Projektlaufzeit**

01.03.2004 – 31.12.2006

**Mitarbeiter**

S. Agne, H. Barthel, M. Deller, A. Ebert, B. Klein, C. Reuschling



DFKI GmbH  
Trippstadter Str. 122  
D - 67663 Kaiserslautern

Tel.: 0631/205-75201  
Fax: 0631/205-75202

---

Das diesem Bericht zugrunde liegende Vorhaben wurde mit Mitteln des Bundesministers für Forschung und Technologie unter dem Förderkennzeichen 01 | WD01 gefördert. Die Verantwortung für den Inhalt dieser Veröffentlichung liegt beim Autor.

## Inhaltsverzeichnis

<b>1. Einleitung</b> .....	<b>3</b>
<b>2. Allgemeine System-Architektur</b> .....	<b>4</b>
<b>3. Strukturierung und Organisation von Inhalten</b> .....	<b>6</b>
3.1 Dokument-basierter persönlicher Informationsraum.....	7
3.2 Metadaten der Dokumente.....	8
3.3 Suchfunktionalität des @VISOR-Backend .....	9
3.3.1 Initiale Suchanfrage .....	10
3.3.2 Query Refinement.....	11
3.3.3 Korpusnavigation .....	11
<b>4. Kontext- und inhaltssensitive Visualisierung</b> .....	<b>12</b>
4.1 Immersion .....	14
4.2 Präattentive Visualisierung von Dokumentrelevanzen.....	20
<b>5. Multimodale Interaktion und Navigation</b> .....	<b>24</b>
5.1 Immersive 6DOF-Interaktion .....	24
5.2 Taktiler Feedback.....	28
5.3 Dimensional Congruence.....	32
<b>6. Finaler Demonstrator</b> .....	<b>33</b>
<b>7. Evaluierung und Demonstrationsszenarien</b> .....	<b>38</b>
7.1 Evaluierung Phase I.....	40
7.2 Evaluierung Phase II.....	41
7.3 Evaluierung Phase II.....	42
7.4 Evaluierung Phase IV .....	43
<b>8. Publikationen</b> .....	<b>46</b>

## 1. Einleitung

Ein Großteil der heutigen Ansätze für Mensch-Maschine-Schnittstellen (HCI – Human Computer Interfaces) fokussiert seine Untersuchungen auf Punkte, in denen Computer die menschliche Leistungsfähigkeit übertreffen. In der Regel wird dabei davon ausgegangen, dass sich die Benutzer an das Interface anpassen müssen und nicht umgekehrt. Diese Ansätze beachten dabei nicht, dass Computer zwar in Gebieten wie dem Durchsuchen und Strukturieren von großen Datenmengen sehr effizient eingesetzt werden können, der Mensch jedoch aufgrund seiner visuellen und haptischen Fähigkeiten Daten eher visuell anordnet und hierdurch Beziehungen zwischen verschiedenen Datenmengen (Metadaten) leichter erkennen kann.

Die Vision für die Forschungsarbeiten in @VISOR ist daher die Schaffung einer individuell anpassbaren virtuellen Welt, welche diese Wahrnehmungsfähigkeiten adressiert und die Barriere zwischen der menschlichen Aktivität und ihrem technischen Abbild überbrückt. Dabei ist der Begriff der Immersion – ein Synonym für einen unterbrechungsfreien und fließenden Arbeitsablauf basierend auf Elementen der multimodalen Interaktion und Navigation – von zentraler Bedeutung. Die beiden zentralen Fragestellungen in diesem Zusammenhang sind dabei:

- Wo sind die menschlichen Fähigkeiten denen des Computers überlegen (und umgekehrt)?
- Wie kann man die Vorteile in den Arbeitsweisen von Mensch und Maschine vereinen und somit die jeweiligen Nachteile mindern oder sogar völlig eliminieren?

Da die Untersuchungen von allgemeinen Informationsräumen den Projektrahmen sprengen würden, konzentrieren sich die Forschungsarbeiten auf individuelle, virtuelle und dokumentenbasierte Informationsräume. Diese Informationsräume sind mächtig genug, um alle anvisierten Ziele zu beschreiben und die jeweiligen Forschungsergebnisse zu evaluieren. Beispiele sind hier die effiziente Suche in Dokumenten und Dokumentenstapeln, das Organisieren und Katalogisieren von Dokumenten, dynamische Prioritätswechsel verbunden mit der Restrukturierung der entsprechenden Dokumentenräume usw. Der Demonstrator für @VISOR ist daher ein virtueller Schreibtisch, welcher einen virtuellen und dokumentbasierten 3D-Informationsraum beschreibt und eine innovative Unterstützung bei der Arbeit mit Dokumenten bietet.

Die Verbindung von Wissensstrukturen und Visualisierung ist zentrales Ziel von @VISOR und liefert Beiträge zur hochaktuellen Thematik der Arbeitsprozessoptimierung (Erreichung von mehr und höheren Zielen mit weniger Personal).

## 2. Allgemeine System-Architektur

Die @VISOR-System-Architektur ist aufgeteilt in vier verschiedene Forschungsaspekte: Strukturierung und Organisation von Inhalten, kontext- und inhaltsensitive Visualisierung, multimodale Interaktion zur Immersionserhaltung sowie Navigation in virtuellen Umgebungen (vgl. Abb. 1). Im Rahmen des vorliegenden Berichtes sind die Gebiete Interaktion und Navigation aufgrund ihrer gemeinsamen Wurzeln unter einer Überschrift zusammengefasst. Neben den erwähnten Forschungsaspekten spielt die Umsetzung der Ergebnisse eine entscheidende Rolle in @VISOR und wird im Unterpunkt Konzept, Evaluierung und Demonstrationsszenario näher beschrieben.

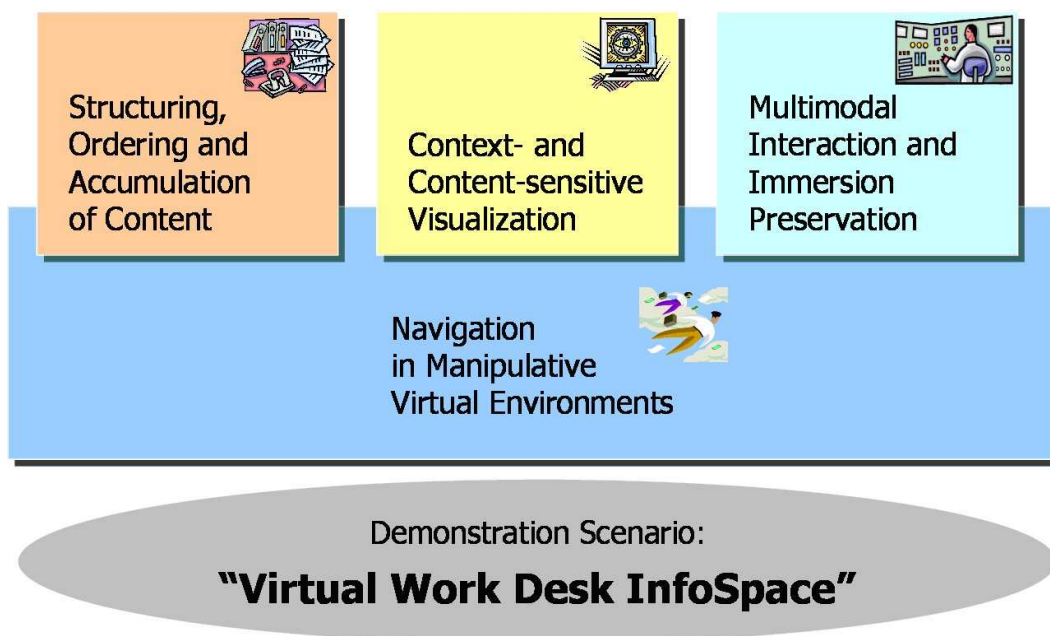


Abbildung 1: Forschungsaspekte

Abbildung 2 veranschaulicht in modularer Sichtweise das entwickelte Framework-Konzept für @VISOR und die Zusammenhänge der einzelnen Forschungsmodule. Im oberen Bereich werden die Inhalte und Metadaten analysiert und verarbeitet bzw. generiert. Im mittleren Bereich sind die verschiedenen Kontexte zu erkennen. Hervorzuheben ist hier insbesondere der Benutzerkontext, der auf einem individuellen Benutzermodell basiert und die personalisierten Ziele und Wünsche des aktuellen Benutzers beschreibt. Der untere Bereich zeigt die Visualisierungs-, Interaktions- und Navigationselemente, die für die Benutzerinteraktion, das haptische Feedback und das 2D/3D-Rendering auf bzw. mit verschiedensten Geräten verantwortlich sind. Diese liefern ihrerseits Parameter für die Modellierung der einzelnen geräte- und benutzerabhängigen Kontexte. Die Verwaltung und Koordination aller Module übernimmt die zentrale @VISOR-Kontrolleinheit (Control Unit).

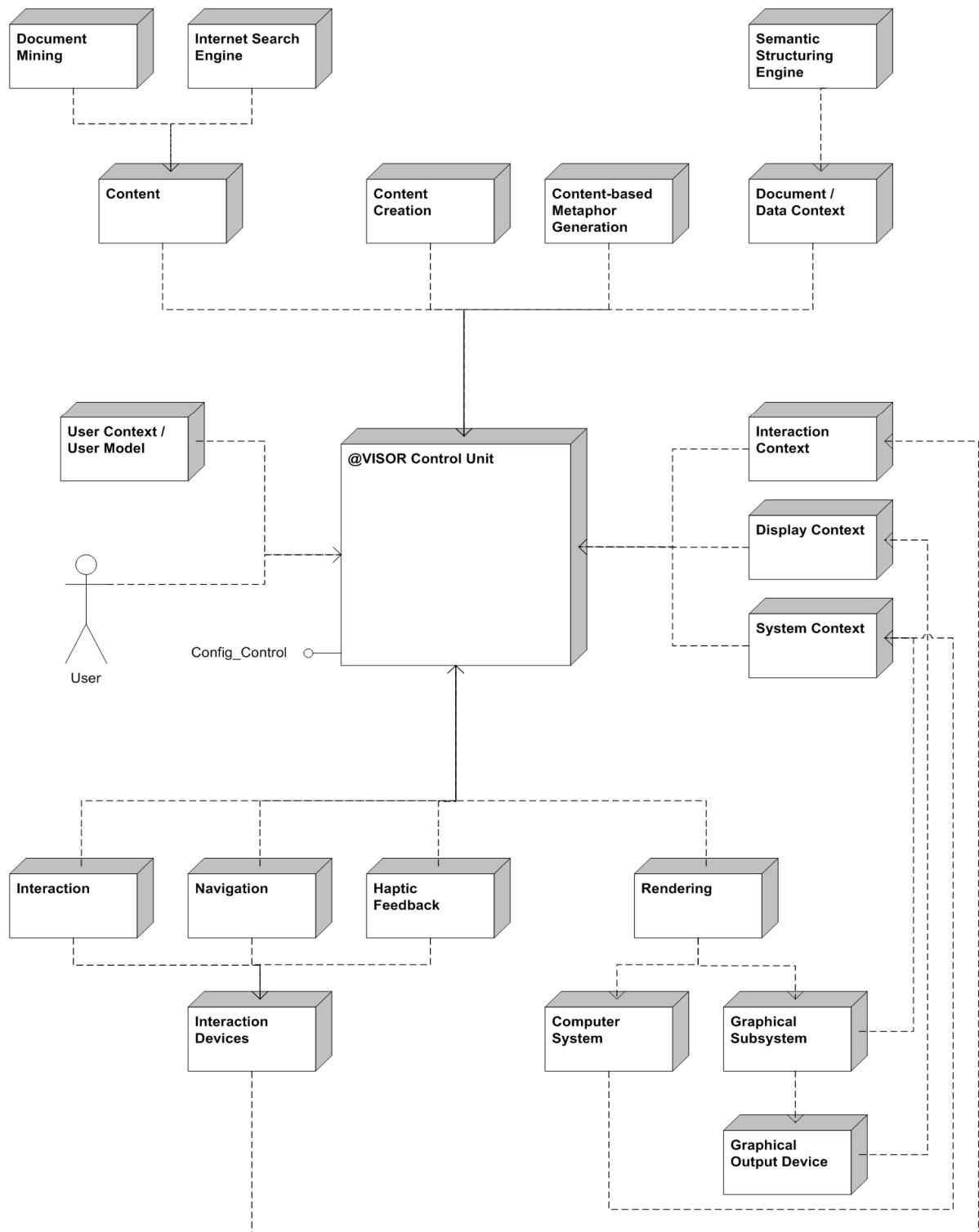


Abbildung 2: @VISOR-Framework.

Es ist offensichtlich, dass das Diagramm nur eine theoretische Sicht auf das @VISOR-Framework geben kann. Aus implementierungstechnischer Sicht werden aus Performance- und Effizienzgründen einzelne Module ohne den Umweg über die Control Unit direkt miteinander kommunizieren. Dies ist im Design der jeweiligen Modulschnittstellen entsprechend beachtet.

Die Strukturierung der folgenden Unterkapitel basiert auf den einzelnen Hauptforschungsthemen, die im Rahmen von @VISOR untersucht wurden und in deren Rahmen während der Projektlaufzeit verschiedene Teil-Demonstratoren entwickelt, evaluiert und verfeinert wurden, die schließlich in den Gesamtdemonstrator des Projekts einfließen.

### 3. Strukturierung und Organisation von Inhalten

Im Bereich der Strukturierung und Organisation von Inhalten war die Hauptaufgabe die Konzeption und Realisierung des @VISOR-Backends. Das Backend ist für zwei Hauptaufgaben zuständig. Zuerst müssen alle Dokumente, die zum persönlichen Informationsraum gehören, für eine schnelle Suche aufbereitet werden. Hierzu werden der textuelle Inhalt eines Dokuments sowie alle für eine Suche relevanten Metadaten in einem effizienten Index gespeichert. Des Weiteren stellt das Backend die eigentliche „Search Engine“ für das @VISOR-Frontend zur Verfügung, die sehr schnell sein muss.

Der dokument-basierte persönliche Informationsraum eines Benutzers konstituiert sich durch alle Dokumente auf seinem Rechner. Hierzu gehören Dokumente verschiedener Formate, aber auch seine E-Mails, die unter Umständen auf einem externen Server gespeichert sind. Eine detaillierte Beschreibung findet sich in Abschnitt 3.1.

Außer dem eigentlichen Text eines Dokuments werden zusätzlich noch alle relevanten Meta-Informationen zu den Dokumenten erfasst und indexiert. Einige sind explizit definiert, wie beispielsweise der Titel bei PDF-Dokumenten, andere werden geschätzt, wie z. B. die Seitenanzahl bei ASCII- oder HTML-Dokumenten. Zusätzlich werden für jedes Dokument wichtige beschreibende Wörter (Buzzwords/Keywords) auf Basis von informationstheoretischen Maßen erfasst. In Abschnitt 3.2 werden die vom @VISOR-Backend indexierten Metadaten detailliert beschrieben.

Die Möglichkeit zum Spezifizieren einer Suchanfrage stellt einen wesentlichen Teil der Gesamtfunktionalität des Backends dar. Je mehr Optionen ein Nutzer durch das System erhält, desto zielgenauere Anfragen kann er formulieren. Die aktuelle Version des @VISOR-Backends stellt u. a. folgende Funktionalitäten zur Verfügung: Schlüsselwortsuche, Phrasensuche, Boolesche Operatoren, Feldbezogene Anfragen, Wildcards, Phonetische Suche und Dokumentähnlichkeitssuche (vgl. Abschnitt 3.3).

### 3.1 Dokument-basierter persönlicher Informationsraum

Im Rahmen des Projektes @VISOR wurde der Fokus auf PDF-Dokumente gelegt. Dennoch wurde bei der Entwicklung des Backends darauf geachtet, dass weitere wichtige Dokumentformate unterstützt werden. Somit kann das Backend auch in industrienahen Projekten wieder verwendet werden. Im Folgenden sind die wichtigsten, vom @VISOR-Backend unterstützten Dokument-Formate beschrieben:

- **Adobe Portable Document Format (\*.pdf):** Die Firma Adobe hat dieses Dateiformat offen gelegt, um Drittanwendungen die Möglichkeit zu bieten, Dateien diesen Typs anzeigen zu können. Das PDF-Format erfreut sich so immer zunehmender Beliebtheit und wird oft zum Austausch von Dokumenten eingesetzt, die nicht weiter editiert werden sollen.
- **MS-Word Dokumente (\*.doc, \*.rtf):** Microsoft Word ist das wohl am weitesten verbreitete Dokumentenformat. Obwohl es sich hierbei um ein nicht-offenes, proprietäres Dateiformat handelt, muss dieses aufgrund der großen Anzahl an Nutzern unterstützt werden.
- **Open Office Dokumente (\*.odf):** Open Office ist eine Open Source-Alternative zum Microsoft-Office-Paket und erfreut sich immer größerer Beliebtheit. ODF steht hierbei für *Open Document Format*.
- **Einfache Textdateien (z.B. \*.txt):** Der 'Urtypus' des Dokumentformats, wenn man davon sprechen kann. In Textdateien können beliebige Dinge gespeichert werden, wobei es sich nicht immer um für den Menschen lesbaren Klartext handeln muss. Die Dateinamenendung '.txt' bezeichnet im Regelfall ein Dokument, das für den Nutzer verständliche Prosa enthält, weshalb Dateien dieses Typs ohne größeren Aufwand indiziert werden können. Bei Dateien beliebiger Dateinamenerweiterung kann beispielsweise mit Hilfe einer Sprachklassifikation festgestellt werden, ob für den Menschen verständlicher Fließtext darin enthalten ist.
- **Hypertext Markup Language (\*.html):** Mit der Popularität des Internets hat dieses Dateiformat weite Verbreitung gefunden. HTML wird meistens zur Darstellung von Internet-Webseiten eingesetzt, oft aber auch lokal z.B. zum Darstellen von Dokumentationen.
- **Postscript (\*.ps):** PostScript hat sich über die Jahre zu einem Standard in der Druckindustrie entwickelt, wird aber immer mehr von PDF verdrängt, das ebenfalls von Adobe entwickelt wurde. Grafiken und Druckseiten werden als Dateien im PostScript-Format angelegt, um sie auf den unterschiedlichsten Ausgabegeräten auszugeben.

- **E-Mail-Postfach via IMAP:** Die Abkürzung IMAP steht für Internet Message Access Protocol. Das IMAP-Protokoll erlaubt den Zugriff auf und die Verwaltung von empfangenen E-Mails. Dieses Protokoll bietet die Möglichkeit, über ein standardisiertes Interface die E-Mails eines Nutzers zu verarbeiten, ohne auf die jeweils proprietären Formate der Fülle an E-Mail-Clientapplikationen eingehen zu müssen. Dabei ist sowohl eine unverschlüsselte Verbindung als auch eine mit SSL verschlüsselte Verbindung zum IMAP-Server möglich.
- **Wikipedia via XML:** Zusätzlich wurden noch alle Artikel des deutschen Wikipedia<sup>1</sup> als eine Art allgemeine Wissensquelle importiert, um Large-Scale Evaluierungen und Performanztests des Backends zu ermöglichen. Insgesamt handelt es sich um 590.895 Artikel, die das @VISOR-Backend sehr effizient indiziert und somit für die Suche zur Verfügung stellt.

### 3.2 Metadaten der Dokumente

Der Mehrwert eines Systems zur Verwaltung des persönlichen Dokumenten-Informationsraums wird einem Nutzer erst dann ersichtlich, wenn keine allzu große administrative Hürde zum Erschließen dieses Informationsraums - für das System - erforderlich ist. Dieses gilt vor allem auch für die spätere Wartung bei der Nutzung des Systems. Überwiegt der subjektiv empfundene Eindruck des Mehraufwands den, ebenfalls subjektiv empfundenen, Nutzen, wird der Benutzer keinen Vorteil beim Verwenden eines solchen Systems sehen. Dies gilt insbesondere auch im Rahmen einer Evaluation. In diesem Kontext sind beim Erfassen der Metadaten vor allem solche von Relevanz, die von der Maschine automatisch erhoben werden können.

Die Metadaten können grundsätzlich in drei Kategorien unterteilt werden:

1. Metadaten, welche direkt aus dem Datenformat ersichtlich sind:

- Seitenzahl
- Sprache
- Herkunft / Institut / Autor / Land (sofern Bestandteil des jeweiligen Formats, ansonsten nicht automatisch erfassbar)
- Dokumenttyp, z.B. Word, Postscript, PDF, HTML (entspricht dem Datenformat)
- Dokumentart, z.B. Bericht, E-Mail, Paper (sofern Bestandteil des jeweiligen Formats, ansonsten teilweise analog zum Dokumenttyp, z.B. E-Mails in IMAP-Postfächern)
- Erstellungsdatum, Änderungsdatum, letztes Zugriffsdatum. (Bei dateibasierten Datenformaten)

---

<sup>1</sup> <http://de.wikipedia.org/>



2. Metadaten, welche aus den erhaltenen Dokumenten extrahiert bzw. berechnet werden:

- Thumbnail des Deckblattes / Logos
- Seitenzahl (über Heuristiken, z.B. bei E-Mails)
- Sprache (über entsprechende Klassifikationstechniken)
- Dokumentart, z.B. Bericht, E-Mail, Paper (über Heuristiken, allerdings stark domänenabhängig => es stellt sich zu Recht die Frage, ob eine solche Zuordnung in einem allgemeinen Kontext des persönlichen Informationsraums in ausreichender Qualität getroffen werden kann)
- Keywords (über Techniken zur Buzzword-Extraction)
- Ähnlichkeit zu einem Initialdokument (über document retrieval Techniken)
- Ähnlichkeit zu einer Anfrage

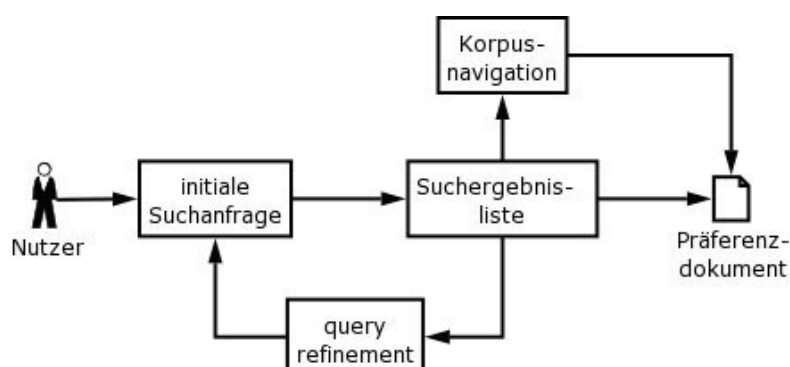
3. Metadaten, welche von Nutzern manuell erstellt werden:

- Annotationen / freie Kommentare / qualitative Bemerkungen / Erfahrungen / Bewertungen
- Dokumentart z.B. Bericht, E-Mail, Paper (sofern nicht vom Format unterstützt sowie nicht automatisch klassifizierbar)

### 3.3 Suchfunktionalität des @VISOR-Backend

Die Möglichkeit zur Spezifikation einer Suchanfrage stellt einen wesentlichen Teil der Gesamtfunktionalität des @VISOR-Backends dar. Je mehr Optionen vom System zur Verfügung stehen, umso flexibler kann die Navigation und Interaktion in @VISOR realisiert werden. Dabei kann zwischen Optionen zur initialen Suchanfrage (Query) und solchen zu einem nachfolgenden 'Verfeinern' der Anfrage (Query Refinement) unterschieden werden. Ferner können unter dem Aspekt des Orientierings Möglichkeiten zur Navigation durch den Korpus geschaffen werden, mit denen der Nutzer ebenfalls Dokumente seiner Präferenz finden kann. Dabei stellen Navigationsmöglichkeiten eine zusätzliche Form der Suchanfrage dar. Navigation in @VISOR geschieht ausgehend von einem Dokument in einer Suchergebnisliste.

Alle Suchanfragen mit ihren Zusammenhängen sind in der folgenden Abbildung illustriert.



### 3.3.1 Initiale Suchanfrage

In @VISOR wird eine initiale Suchanfrage mit Hilfe von 'Suchtermen' spezifiziert, die vom Nutzer des Systems angegeben werden. Als Ergebnis liefert das System die Dateien, die die angegebenen Suchbegriffe enthalten. Dabei werden die Dokumente der Ergebnisliste nach ihrer durchschnittlichen Relevanz zu den initialen Termen sortiert. Zu den einzelnen Termen der Suchanfrage können spezifische Angaben gemacht werden, wozu mehrere syntaktische Möglichkeiten gegeben sind.

Das @VISOR-Backend bietet folgende Funktionalität in Bezug auf die Syntax der Suchanfrage:

- **Wildcards:** Diese dienen als Platzhalter für eine beliebige Zeichenfolge und können somit eingesetzt werden, wenn der vollständige Suchterm nicht bekannt ist oder einfach generalisiert angegeben werden soll.  
*Syntax:* '\*' steht für kein oder mehrere beliebige, '?' für genau ein beliebiges Zeichen.
- **Initiales Boosting:** Dieser Begriff steht für die anfängliche Gewichtung einzelner Terme der Suchanfrage. Terme mit hohem Gewicht sind somit wichtiger für die Anfrage und umgekehrt. Dokumente, die als relevant zu einem hoch gewichteten Term erachtet werden, wandern somit in der Ergebnisliste nach oben.  
*Syntax:* '^' gefolgt von einer Zahl steht für die Gewichtung eines Terms. Dabei stehen Zahlen >1 für positive, Zahlen im Bereich [0..1] für negative Gewichtungen.
- **Initiale Attributselektion:** Eine initiale Attributselektion bezeichnet die Möglichkeit, selektiv in diesen Attributen zu suchen. Damit können z.B. Dokumente mit einer bestimmten Mindestseitenzahl oder mit einem ausgezeichneten Schlagwort aufgefunden werden.  
*Syntax:* Das Attribut wird durch ein dem Suchterm vorangestelltes 'Attributname:' ausgewählt. Sollen mehrere Terme angegeben werden, fasst man diese mit einfachen Hochkommata ('term1 term2') ein.

Zusätzlich zu der syntaktischen Vorgehensweise innerhalb von Suchanfragen existiert noch ein weiterer Suchmodus im @VISOR-Backend, der als Alternative zur Standardsuche gewählt werden kann.

- **Phonetix-Modus:** Dieser Modus kann gewählt werden, wenn lediglich der Klang, nicht aber die genaue Schreibweise eines Suchbegriffs bekannt ist. Mit Hilfe phonetischer Algorithmen werden die angegebenen Terme in Lautschrift gewandelt, die anstatt der Originalterme für die Suche eingesetzt werden. Es stehen die Algorithmen *Soundex*, *Metaphone* und *DoubleMetaphone* zur Auswahl.

### 3.3.2 Query Refinement

Im Abschnitt 3.3.1 wurde auf die Formulierung einer Query an das System eingegangen. Doch was tun, wenn die Ergebnisliste nicht den eigentlichen Bedürfnissen entspricht?

Die übliche Vorgehensweise eines Users besteht darin, die ursprüngliche Frage anders zu formulieren und sie nochmals an das System zu schicken. Dies geschieht so lange, bis der Nutzer das oder die Dokument(e) seiner Wahl gefunden hat, oder bis er das Gefühl hat, dass sich im Korpus kein Dokument befindet, welches zur eigentlichen Intention seiner Frage passt. Je nach Wissensstand des Nutzers zu den angefragten Themengebieten bzw. seines momentanen Erwartungshorizonts ist seine erste Anfrage eher generalisierter Natur. Der User will oder muss sich erst mal einen Überblick verschaffen, was zu diesem allgemeinen Thema im Korpus zu finden ist. Erst in zweiter Instanz stellt er – ausgehend von der zuerst erhaltenen Ergebnisliste – die eigentliche Frage, die spezifischer auf seine Intention abzielt.

Der Effekt des Neuformulierens einer Initialquery kann alternativ auch durch Möglichkeiten des Feedbacks zur Ergebnisliste erzielt werden: Der Nutzer sagt dem System das – ausgehend von der gestellten Anfrage an das System und der daraus resultierenden Ergebnisliste – er in bestimmten Aspekten gerne spezifischere, oder umgekehrt auch generalisiertere, Ergebnisse hätte.

*Anfrageverfeinerung in Bezug auf die Terme der initialen Suchanfrage:* Dabei handelt es sich um eine Form des in Abschnitt 3.3.1 bereits angesprochenen 'Initiales Boosting'. Der Nutzer erhält dabei die Gelegenheit, die Terme seiner ursprünglichen Anfrage einzeln zu gewichten – wobei sich die Reihenfolge der Dokumente in der Ergebnisliste entsprechend angleicht.

*Relevance Feedback:* Dabei wählt der Nutzer aus der Ergebnisliste eins oder mehrere Dokumente aus, die am ehesten seiner Intention entsprechen. Die Ergebnisse werden daraufhin aufgrund ihrer Ähnlichkeit zu diesen positiv bewerteten Dokumenten umsortiert.

### 3.3.3 Korpusnavigation

Die Interaktion mit dokument-basierten persönlichen Informationsräumen mit Hilfe von dreidimensionalen Benutzerschnittstellen erfordert die Möglichkeit zur Navigation durch die Menge aller zur Verfügung stehenden Dokumente. Dies dient in erster Instanz der Übersicht. Man kann erkennen, welche ähnlich beschaffenen Dokumente, ausgehend vom Referenzdokument, zur Verfügung stehen. Ferner ist es auch eine Möglichkeit, in den Dokumenten mehr oder weniger gezielt zu 'stöbern', was eine zusätzliche Funktionalität zur Suche darstellt.

Als Ausgangspunkt kann ein einzelnes Dokument, aber auch eine Menge von Dokumenten dienen. Das @VISOR-Backend berechnet dazu eine sortierte Liste von ähnlichen Dokumenten. Die Berechnung der Ähnlichkeit kann auf den Dokumentvektoren, aber auch auf allen erfassten Metadaten der Dokumente basieren. Darüber hinaus können auch explizit definierte Ähnlichkeiten (Relationen) zwischen Dokumenten zur Navigation genutzt werden.

## 4. Kontext- und inhaltsensitive Visualisierung

In der Literatur sind zum Stichwort *Information Visualization* viele verschiedene Methoden zur Darstellung von Daten zu finden, welche jedoch samt und sonders auf spezielle Anwendungsfälle abzielen. Die Auswahl der passenden Metapher obliegt dabei stets dem Benutzer, der bei einer falschen oder unpassenden Wahl ein verfälschtes Ergebnis erhält. Im @VISOR-Konzept wird dieses Problem durch eine automatische Auswahl der geeigneten Metapher nach Art der darzustellenden Daten umgangen. Durch die in @VISOR integrierte Ergänzung der Daten um Meta-Informationen ist eine schnelle automatische Auswahl der jeweilig geeigneten Metapher möglich, da in unterschiedlichen Kontexten die gleichen Daten durchaus mit unterschiedlichen Metaphern sinnvoll dargestellt werden.

Eine entsprechende Anwendung dieses Schemas findet sich in vereinfachter Form auf heutigen PCs in der Form von Dateinamenerweiterungen wieder. Diese ermöglichen je nach Sicht z.B. eine Zuordnung eines passenden Piktogramms bei der Verzeichnisauflistung oder die Auswahl einer passenden Anwendung bei der Ausführung der Datei. Im Rahmen von @VISOR wurde dieses Schema auf Dokumentinhalte erweitert und ermöglicht dabei eine wesentlich umfassendere Anpassbarkeit der Darstellung, was gerade durch die zusätzliche Erweiterung von zwei auf drei Dimensionen im Darstellungsraum einen integralen Bestandteil darstellt. Die reine Ersetzung des zweidimensionalen Desktops durch dreidimensional angeordnete, weiterhin zweidimensionale Teil-Desktops, wie sie die im Arbeitspaket CCSV-EXIS evaluierten heutigen Ansätze zu 3D-Desktops meist verwenden, ist dabei denkbar ungeeignet. Es wird lediglich eine Sichtverzerrung angewendet, jedoch keine weitere wirklich informationstragende Dimension hinzugefügt.

Neuere Ansätze auf diesem Gebiet gehen daher den Weg, ganze Umgebungen in 3D aufzubauen, welche nach Art moderner Computerspiele „begehbar“ sind. Innerhalb dieser Welten sind dann Interaktionselemente platziert, welche zugeordnete Funktionalitäten bereitstellen. Diese beschränken sich im Allgemeinen jedoch auf Web-Browsing, E-Mail und Medienwiedergabe (Videos, Musik). Ein Einbeziehen echter Dokumentenstrukturen fehlt diesen Systemen zumeist gänzlich. Auch werden dem Benutzer keine neuen Interaktionsfunktionalitäten angeboten, sondern die bekannten lediglich in den dreidimensionalen Raum übertragen.

Die genannten Evaluationsergebnisse beeinflussten in großem Masse die Feinplanung der ersten Schritte in @VISOR, insbesondere die Detailkonzeption des ersten visuellen Demonstrators. Die entsprechenden Teilkonzepte sind im Gesamtsystemdesign integriert, erste Teile wurden bereits früh zu Testzwecken implementiert. Um die „auf dem Papier entwickelten“ Metaphern intuitiv testen zu können, wurden parallel dazu Animationen händisch mittels Standard-Werkzeugen erstellt. Diese wurden anschließend im Hinblick auf ihre Eignung im Rahmen von @VISOR evaluiert.

Zusätzlich wurden vom Projektteam die benötigten Begriffe *Kontextsensitive Visualisierung* und *Inhaltssensitive Visualisierung* untersucht und definiert: Eine Visualisierung bzw. eine Visualisierungsapplikation ist genau dann kontextsensitiv, wenn sie vorgegebene Kontexte berücksichtigt und flexibel auf sich dynamisch ändernde Situationen proaktiv (ohne Notwendigkeit eines Benutzereingriffs) reagiert. Basierend auf dieser Definition wurden die zentralen Szenarien der kontextsensitiven Visualisierung identifiziert:

**Benutzerkontext:** In der heutigen Informationsgesellschaft stellen Individualität und Personalisierung entscheidende Faktoren für den Erfolg oder Misserfolg einer neuen Idee oder eines Produktes dar. Um beide Voraussetzungen zu erfüllen, muss der Benutzer durch ein Benutzermodell beschrieben werden, welches die individuellen Benutzereigenschaften repräsentiert, z. B. Annahmen, Interessen, Präferenzen, Fähigkeiten, Aufgaben usw. Diese Eigenschaften sind entweder explizit vom Benutzer vorgegeben (Benutzerprofil) oder implizit aus dem Benutzerverhalten abgeleitet. Der Kontext eines durch ein Benutzermodell beschriebenen Benutzers umfasst physikalische Informationen (Position im physikalischen oder virtuellen Raum und Bewegungen darin), soziale Kontexte (Stellung, Rollen, Berechtigungen), Zeit (Termine, Zeitvorgaben), individuelle Fähigkeiten und Interessen.

**System- und Interaktionskontext:** Auf dem Markt befindet sich heutzutage eine Vielzahl von sich hardwaretechnisch stark unterscheidenden Plattformen. Die einzelnen Fähigkeiten eines Rechners (wie Prozessorleistung, Speicherausbau, Grafikfähigkeiten, Betriebssystem usw.) lassen sich unter dem Begriff „Systemkontext“ zusammenfassen. Neben den statischen Parametern des Systemkontextes müssen in diesem Zusammenhang aber auch sich dynamisch ändernde Werte in Betracht gezogen werden. So ändern sich je nach Art und Anzahl der parallel oder (unbemerkt) im Hintergrund laufenden Programme die Auslastung des Systems und damit die für die jeweilige Applikation zur Verfügung stehenden Systemressourcen. Die dynamischen Systemparameter werden daher unter dem Begriff „Interaktionskontext“ zusammengefasst.

**Situationskontext:** Der Situationskontext ist eng mit dem Benutzerkontext verbunden und beschreibt die aktuelle Situation des Benutzers. Zentrale Parameter stellen dabei Orts- und Umgebungsmerkmale dar, welche insbesondere die Frage, wo sich der Benutzer gerade befindet, beantworten. Gerade im Hinblick auf die heute immer wichtiger werdende Mobilität stellt die Berücksichtigung der aktuellen Situation eines Benutzers einen äußerst zentralen Faktor für die Darstellung von Informationen dar. Location Based Services (LBS) stellen standortbezogene Dienste zur Verfügung und bieten dem Benutzer spezifische, auf seinen jeweiligen Raumkontext zugeschnittene Dienste an.

**Darstellungskontext:** Neben den für die Performance interessanten Hardware- und Auslastungsaspekten ist vor allem die Art des Ausgabemediums für Visualisierungsanwendungen von großer Bedeutung. Je nach Art, Größe und Auflösung muss die darzustellende Information verschieden aufbereitet werden. Als Präsentationstypen können z. B. Texte, 2D-Grafiken, Animationen, Audioausgaben, Videos oder 3D-Darstellungen zur

Anwendung kommen. Die Frage „Wie stelle ich etwas dar?“ wird durch den Darstellungskontext beschrieben. Gerade bei mobilen Anwendungen erschwert die stark begrenzte Darstellungsfläche in Verbindung mit den geringen Auflösungen der Bildschirme die adäquate Präsentation der darzustellenden Information. Die hierdurch notwendige Aufteilung des Inhalts in kleine Präsentationseinheiten führt zudem zu komplexen Navigationsstrukturen. Im Gegensatz zu den mobilen Anwendungen erfordert die Visualisierung von Informationen auf 3D-Displays die Entwicklung und den Einsatz hochperformanter (i.d.R. GPU-basierter) Algorithmen, welche hohe Detaillierungsstufen und Auflösungen zusammen mit interaktiven Frameraten ermöglichen.

**Datenkontext:** Die von einer Visualisierungsapplikation darzustellenden Daten hängen im Allgemeinen von den Wünschen und Vorgaben des Benutzers ab und machen eine individuelle Anpassung und Ausrichtung der verwendeten Daten an den jeweiligen Benutzer erforderlich. In der Regel sind diese Parameter jedoch nicht im Voraus bekannt, sondern werden erst zur Laufzeit definiert. Die benötigten Daten können also nicht vorberechnet werden. Allgemein übliche Algorithmen zur Datengenerierung setzen jedoch häufig auf komplexe Simulationsprozesse, welche zwar qualitativ hochwertige Visualisierungen ermöglichen, jedoch aufgrund des immensen Berechnungsaufwandes bei interaktiven Anwendungen keine Anwendung finden können. Der Datenkontext ist offensichtlich stark von Benutzer- und Situationskontext abhängig.

Analog zur Definition der kontext-sensitiven Visualisierung kann die inhaltssensitive Visualisierung definiert werden: Eine Visualisierung bzw. eine Visualisierungsapplikation ist genau dann inhaltssensitiv, wenn sich die visuelle Repräsentation der Informationen sowie die damit verbundenen Interaktions- und Navigationsstrukturen dynamisch an die jeweiligen Inhalte adaptieren.

Zusammenfassend lässt sich sagen, dass die kontext- und inhaltssensitive Visualisierung das entscheidende Bindeglied zwischen Dokumentenmanagement und Visualisierung darstellt.

## 4.1 Immersion

Ein zentrales Ziel des Projektes @VISOR ist die Schaffung und Erhaltung einer immersiven virtuellen Umgebung. Hierzu wurde ein holistischer, kontext- und inhaltssensitiver Ansatz zur Informationsvisualisierung eingeführt, welcher die automatische Adaption an verfügbare Hardwarekonfigurationen beinhaltet. Durch effiziente Visualisierungsmetaphern und die Möglichkeit, moderne stereoskopische Anzeigeeinheiten zu benutzen, wurde eine starke Immersionswirkung erreicht.



Abbildung 3: Visualisierung auf einem 3D-Display (SeeReal Ci).

Ein wichtiger Schritt, um eine immersive Visualisierung auf jedem gewünschten Rechner zu erreichen, bestand in der Konzipierung und Implementierung eines Frameworks, welches die Features der verwendeten Hardware analysiert und basierend auf den gegebenen technologischen Möglichkeiten die immersive Wirkung der Visualisierung entsprechend maximiert. Natürlich werden die besten Ergebnisse für das Gefühl des "Eintauchens" durch einen echten dreidimensionalen Eindruck der virtuellen Umgebung erreicht, was jedoch eine stereoskopische Sicht der Szene voraussetzt. Um zu dieser stereoskopischen Vision zu gelangen, muss die virtuelle Szene aus zwei verschiedenen Sichtpunkten (je einer für die beiden virtuellen Augen des Benutzers) berechnet werden. Diese generierten Sichten müssen dann exakt für das entsprechende Auge des Benutzers verfügbar gemacht werden, ohne die Visualisierung für das jeweils andere Auge zu beeinflussen.

Es gibt dabei mehrere Möglichkeiten, das Ziel der stereoskopischen Trennung mit spezieller Hardware zu erreichen, welche sich in Form, Verfügbarkeit und Kosten jedoch stark unterscheiden. Auch ist der computergrafische Weg zur Erzeugung und Aufbereitung der stereoskopischen Darstellung auf den verschiedenen Geräten in der Regel unterschiedlich, so dass dies ebenfalls vom Rendering-Framework gehandhabt werden muss.

Um jeweils den bestmöglichen Grad an Immersion zu garantieren, wurde im Projektzeitraum eine performante und flexible Renderengine entwickelt, welche eine Onlineänderung der stereoskopischen Anzeigenmethode erlaubt, ebenso wie die applikationsunabhängige, dynamische Anpassung der für den stereoskopischen Effekt maßgeblichen Parameter. Dies wird von der Renderengine unabhängig zur Verfügung gestellt, ohne dass die tatsächliche Visualisierungsanwendung die stereoskopischen Sichten selbst generieren oder verwalten muss. Die Anwendung instantiiert hierzu ein oder mehrere stereoskopische Fenster und liefert diesen eine gewöhnliche OpenGL-Zeichermethode, um die Szene zu visualisieren. Zusätzlich können dem Fenster Anfangswerte für die den Stereoeffekt beeinflussenden

Variablen wie Augenabstand und fokaler Abstand übergeben werden. Die Schaffung der stereoskopischen Sichten sowie die Verwaltung der besagten Variablen werden von der Fensterklasse eigenständig behandelt, können aber auch von der Visualisierungsanwendung kontrolliert werden. Außerdem können Methoden für die Szeneninitialisierung und die Behandlung von Messages der erstellten Fenster zugewiesen werden. Die generierten Windows unterstützen automatisch die von der Hardware vorgegebenen stereoskopischen Funktionalitäten durch Wahl eines geeigneten Displaymodes. Alternativ kann der gewünschte Modus auch online vom Benutzer gewählt und umgeschaltet werden. Auf diese Weise ist auch möglich, gleiche oder verschiedene Szenen auf mehreren Anzeigegeräten mit unterschiedlichen stereoskopischen Einstellungen gleichzeitig darzustellen. Die Bandbreite der unterstützten stereoskopischen Geräte reicht dabei von einfachen, aus dem Gaming-Bereich bekannten Shutterbrillen, über spezialisierte 3D-Monitore und Head Mounted Displays, bis hin zu Powerwalls.

Basierend auf der beschriebenen Fensterklasse wurden während der Projektlaufzeit verschiedene neue Visualisierungsmetaphern für Dokumente und Dokumentcluster konzipiert und in Demonstratoren umgesetzt.

Zunächst wurden dazu verschiedene Metaphern entwickelt, um das blättern („browsen“) durch virtuelle Dokumente zu visualisieren. Diese wurden anschließend in Form von modularen Einzeldemonstratoren implementiert und evaluiert. Die entwickelten Ansätze beinhalteten dabei sowohl deutlich an die Realität angelehnte Visualisierungen als auch verschiedene stark abstrakte Repräsentationen von Dokumentinhalten (siehe Abbildungen Abb. 4 und 5).

Auch für das Browsen durch Gruppen von Dokumenten („Clustern“) wurden verschiedene Visualisierungsmetaphern erarbeitet und als Demonstratoren umgesetzt (Abb. 6 und 7).

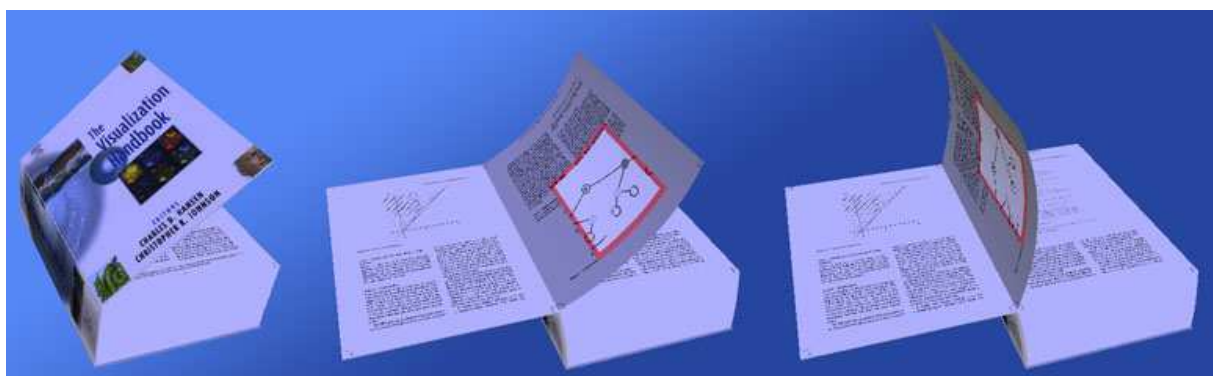


Abb. 4: Durchblättern eines 3D-Buches wie in der Realität.



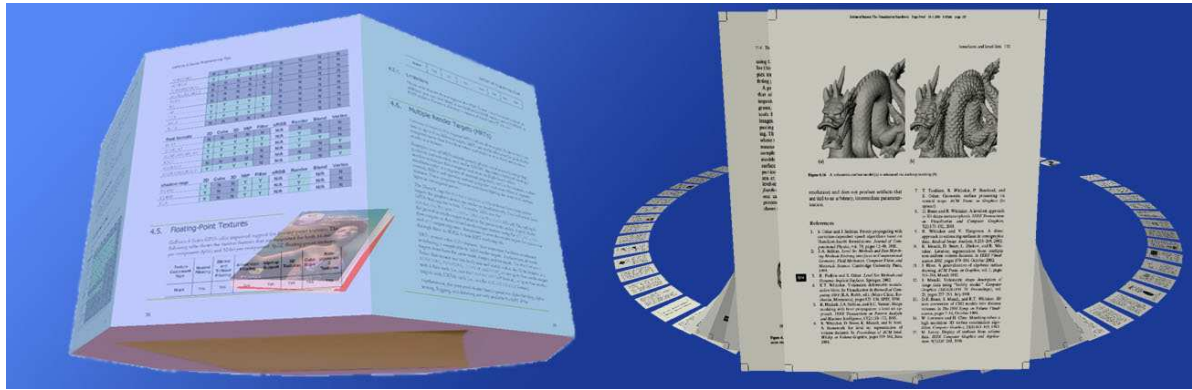


Abb. 5: Durchblättern eines Dokumentes mittels abstrakter Darstellungsformen.

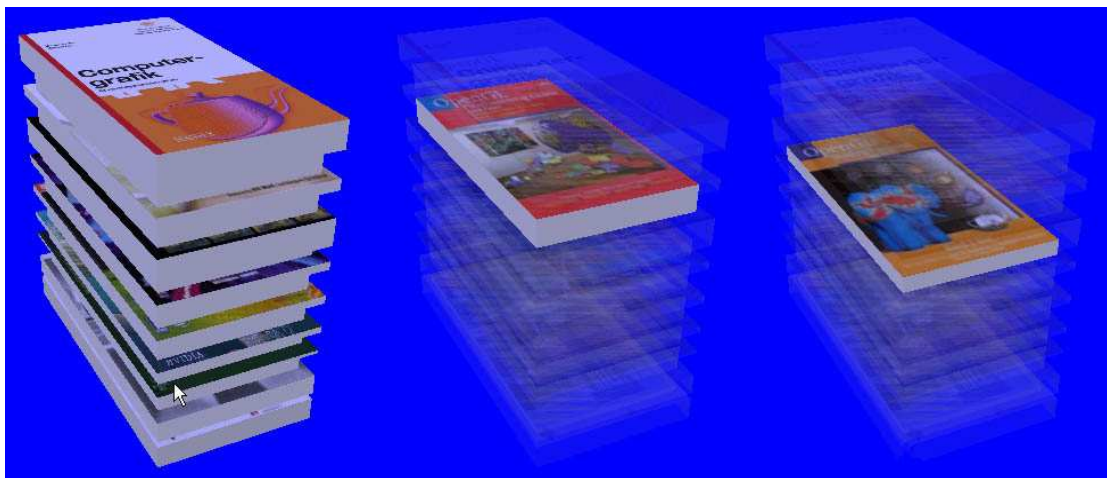


Abb. 6: Durchsuchen eines realistisch visualisierten Dokumentenstapels mittels Transparenzen.



Abb. 7: Durchsuchen eines realistisch visualisierten Dokumentenstapels mittels Herausrotieren.

Um ganze Dokumenträume darstellen zu können, wurden außerdem unterschiedliche Möglichkeiten des Clustering von Dokumenten untersucht. Hierzu wurde eine erste Anknüpfung zwischen der Visualisierungsumgebung „DocuWorld“ und dem Dokumenten-Management realisiert. Hiermit ist es möglich, einfache Suchanfragen zu stellen (Textsuche über verschiedene Dokumentabschnitte: Autor, Titel, Thema, Stichwörter, Volltext) und somit die Wichtigkeit von Dokumenten in Bezug auf diese Anfrage automatisch zu bewerten. Aufgrund der Trefferhäufigkeit und Relevanz der Treffer für die einzelnen Suchanfragen lassen sich Gruppen bilden und zusammen mit ihren Beziehungen untereinander darstellen. Die folgende Abbildung zeigt eine Suchanfrage (Angabe: fünf Keywords); hierbei muss jedoch beachtet werden, dass die Übersichtlichkeit aufgrund der fehlenden 3D-Information in der Abbildung signifikant verschlechtert ist.

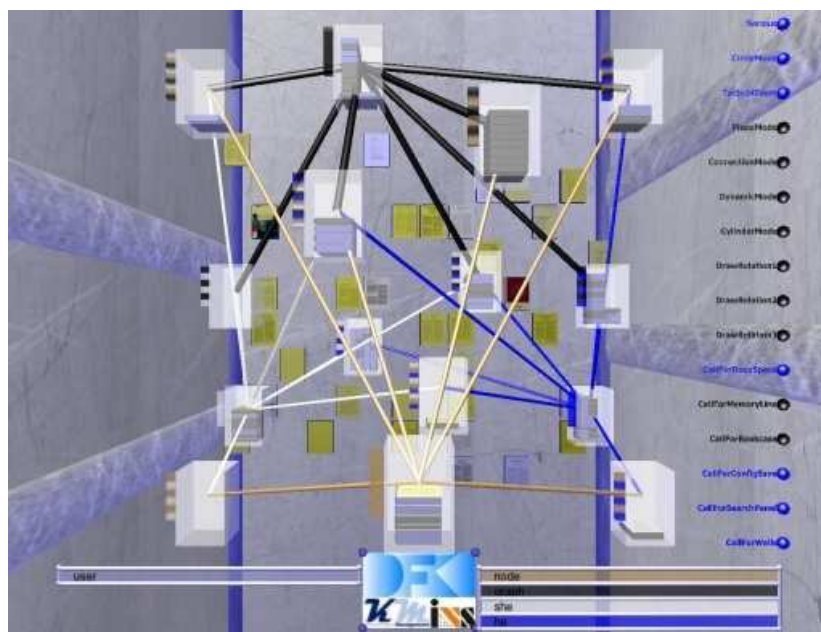


Abb. 8: 3D-Clustering von Dokumenten.

Eine sehr intuitive Alternative besteht in der räumlichen Positionierung der Dokumente basierend auf deren Gesamtrelevanz zur aktuellen Anfrage. Je wichtiger ein Dokument, desto näher erscheint es dem Betrachter, je unwichtiger, desto weiter ist es entfernt. Dokumente ohne Bezug zur Anfrage verbleiben im Hintergrund der virtuellen Umgebung. Es sei auch hier angemerkt, dass die Übersichtlichkeit und Zuordnung der Dokumentrelevanz aufgrund der fehlenden 3D-Information in der Abbildung signifikant verschlechtert ist.

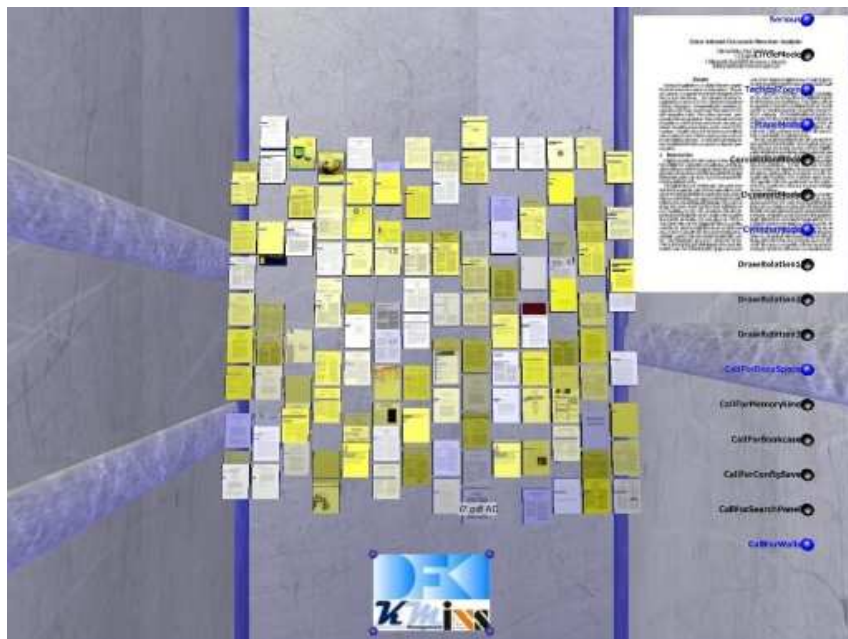


Abb. 9: Visualisierung der Relevanz von Dokumenten durch räumliche Anordnung.

Ausgangspunkt für alle bereits beschriebenen Darstellungsmöglichkeiten ist der so genannte Virtual Workdesk (siehe Abb. 10), der eine natürliche Darstellung und intuitive Interaktion mit der Dokumentensammlung ermöglicht. Hierzu wurden umfassende immersive Metaphern entwickelt und zur Evaluierung in einem ersten Prototypen des virtuellen Workdesks integriert. Dieser beinhaltet u.a. die folgenden Interaktions- und darauf abgestimmten Visualisierungsmetaphern:

- Visualisierung von Post-Its
- Visualisierung der damit verlinkten Cluster
- Visualisierung der kompletten Dokumentensammlung (Aktenschrank)
- Visualisierung aktiver Tasks (Pinboard)
- Visualisierung abgelegter Tasks (Ablagefächer)
- Visualisierungsmetapher für Dokumentbrowsing abhängig von Mono- oder Stereo-Darstellung (für 2-Monitor-Lösung)
- Anlegen und Verbinden von Dokumentclustern
- Anlegen von Dokumentensammlungen (d.h. verlinkte Cluster)
- Anlegen von Post-Its auf Pinboard und Öffnen der damit verlinkten Cluster bzw. Sammlungen
- Taskwechsel (Ablegen von aktiven Tasks, Reaktivieren abgelegter Tasks)
- Auswahl (Suche) von Dokumenten aus Dokumentenpool für den Workdesk
- Blättern durch Stapelvorschau
- Blättern durch Stapel
- Blättern durch Dokumente



Abb. 10: Frühes Konzept des Virtual Workdesk.

## 4.2 Präattentive Visualisierung von Dokumentrelevanzen

Aufmerksamkeit ist ein entscheidender Faktor bei der Betrachtung von Suchaufgaben. Besonders in informationsreichen virtuellen Umgebungen wie dem @VISOR-Frontend ist es unerlässlich, dem Benutzer einen schnellen und intuitiven Eindruck zu vermitteln, welche Teile der Umgebung wichtig für seine momentane Aufgabe sind. Dabei ist jedoch unbedingt zu vermeiden, dass durch die Hervorhebung einzelner Dokumente andere Charakteristika der Umgebung verfälscht werden oder gar verloren gehen. Der Benutzer muss relevante Information schnell finden, ohne seinen Zugriff auf andere Information dabei zu sehr zu behindern. Dies ist besonders wichtig bei der Suchstrategie des „Orientierung“.

Als viel versprechender Ansatz wurde dafür die Benutzung präattentiver Merkmale identifiziert. Unter präattentiver Wahrnehmung versteht man das Phänomen, dass (visuelle) Reize aufgenommen und vom Gehirn verarbeitet werden, noch bevor eine bewusste Wahrnehmung dieser Reize erfolgt. Diese Eigenschaft spielt eine entscheidende Rolle beim Konzept der „Guided Search“. Diese unterteilt den Prozess einer visuellen Suche in zwei Stadien, präattentive und aufmerksame Wahrnehmung. Wie der Name andeutet, findet die präattentive Wahrnehmung statt, bevor bewusste Aufmerksamkeit auf ein Objekt gerichtet wird. Im Allgemeinen gilt eine Wahrnehmung als präattentiv, wenn sie innerhalb einer Zeitspanne von 200 Millisekunden stattfindet. Dieser Zeitrahmen basiert auf der Tatsache, dass das menschliche Auge mindestens 200 Millisekunden braucht, um eine Augenbewegung einzuleiten. In anderen Worten, präattentive Wahrnehmung beschreibt die

Information, die der Mensch mit einem einzigen Blick aufnehmen kann. Visuelle Merkmale, die das Finden von Objekten während dieses Prozesses unterstützen, werden „Preattentive Cues“ genannt. In der nächsten, bewussten Phase der visuellen Suche werden die in der ersten Phase gefundenen Objekte dann seriell nach weiteren visuellen Merkmalen untersucht.

Aufgrund dieser Eigenschaften eignen sich präattentive Hinweise gut, um in der @VISOR WorkDesk Umgebung die Aufmerksamkeit des Benutzers schnell auf die relevantesten Dokumente einer Suchanfrage zu lenken, auch wenn hunderte von Dokumenten gleichzeitig dargestellt werden. Gerade aber beim Betrachten von Dokumentvisualisierungen in einer an visuellen Merkmalen reichen virtuellen Umgebung wie @VISOR ist es wichtig zu beachten, dass diese präattentiven Hervorhebungen nicht die anderen visuellen Eigenschaften der Dokumentrepräsentationen beeinflussen.

Aus diesem Grund wurden im Rahmen des Projektes mehrere verschiedene präattentive Merkmale daraufhin untersucht, wie stark sie die restliche Visualisierung beeinflussen bzw. von der Visualisierung beeinflusst werden. Die alternativen Möglichkeiten wurden als prototypische Demonstratoren implementiert und evaluiert. Dabei wurden sowohl die Auswahl der wichtigsten Dokumente, ihre Reihenfolge und die dafür benötigte Zeit betrachtet. Stark zusammengefasst ergaben sich bei der Evaluierung für die einzelnen Merkmale folgende Eigenschaften:

- **Objektausrichtung:** Hier wurden zwei Alternativen untersucht: drehen des Dokumentes nach links oder rechts und „klappen“ des Dokuments nach hinten. In beiden Fällen führte das Ändern der Ausrichtung zu einer schlechteren Erkennbarkeit des Dokuments. Das Drehen von Dokumenten führte außerdem bei großer Anzahl von Dokumenten zu visueller Überlastung und ließ nicht intuitiv erkennen, ob stärker oder weniger stark gedrehte Dokumente relevanter waren. Beim Klappen war zwar intuitiv erkennbar, dass stärker dem Benutzer zugeneigte Dokumente wichtiger waren, jedoch waren gekippte Dokumente kaum noch erkennbar.
- **Farbe und Intensität:** Wegen der Vielzahl visueller Charakteristika in einer informationsreichen Umgebung wie @VISOR konnten die präattentiven Merkmale Farbe und Intensität nicht getrennt, sondern nur in Kombination evaluiert werden. Dabei zeigte sich, dass Farbe ein sehr starker präattentiver Hinweis ist, der in Kombination mit Intensität auch die Möglichkeit einer quantitativen Abstufung bietet. Allerdings ist eine solche Abstufung nur bei starken Relevanzunterschieden erkennbar, da das menschliche Auge feinere Abstufungen nicht korrekt trennen kann. Als wichtigster Nachteil erwies sich, dass die Benutzung von Farbe sich einerseits stark auf die restliche Visualisierung auswirkt, da natürlich von sich aus farbige Dokumente stark verfälscht werden. Umgekehrt wurden, je nach Wahl der Farbe, auch nicht relevante Dokumente als relevant erkannt, wenn bei diesen Dokumenten im Vornherein diese Farbe vorherrschte.

- **Größe:** Die Größe von Dokumenten erwies sich als weiteres stark präattentives Merkmal. Ein weiterer Vorteil war hier, dass wichtige und daher vergrößerte Dokumente leichter erkennbar waren. Allerdings war es auch hier für die Testpersonen schwer, feinere Relevanzabstufungen zu unterscheiden. Auch führte das Ändern der Größe dazu, dass Dokumente umgeordnet werden mussten oder ein größerer Abstand zwischen Dokumenten gewählt werden musste, was zusätzlichen Darstellungsraum verbrauchte.
- **Animation:** Zur Benutzung von Animation als präattentives Merkmal wurden zwei Varianten gewählt: Wichtige Dokumente „pulsieren“ mit höherer Frequenz bei höherer Relevanz oder wichtige Dokumente „hüpfen“ auf und ab. In beiden Fällen zeigte sich, dass Animation bei einer großen Anzahl Dokumente einen sehr störenden Effekt auf die Visualisierung hat. Auch war eine präattentive Erkennung von quantitativer Abstufung nicht möglich, da zunächst die Frequenz der Animation bewusst betrachtet werden musste.
- **Stereoskopische Tiefe:** Als letztes Merkmal wurde die stereoskopische Tiefe von Objekten betrachtet. Dabei wurden relevantere Dokumentrepräsentationen näher an den Benutzer heran bewegt, während sich weniger wichtige Dokumente vom Benutzer weg in den Hintergrund bewegten. Auch stereoskopische Tiefe stellte sich als starker präattentive Hinweis heraus. Zusätzlich ist die Vorstellung, dass sich wichtigere Dokumente näher beim Benutzer befinden, sehr intuitiv, während der unterschiedliche Abstand als Abstufung der Wichtigkeit auch gut wahrgenommen wurde. Auch wird die Benutzung stereoskopischer Tiefe nicht von anderen Eigenschaften der Visualisierung beeinflusst. Umgekehrt hat man bei stereoskopischer Tiefe zwar das gleiche Problem der Überdeckung anderer Dokumente wie bei der Benutzung von Größe, da nähere Objekte natürlich größer erscheinen; Jedoch kann durch die Verwendung von stereoskopischer Tiefe eine stärkere Abstufung mit weniger Platzaufwand erreicht werden. Im Kontext von @VISOR hat die Nutzung von stereoskopischer Tiefe den zusätzlichen Vorteil, dass sich dadurch ein sehr intuitiver Weg ergibt, die Relevanz von Dokumenten manuell zu korrigieren, indem das Dokument einfach nach vorne gezogen oder nach hinten geschoben wird.

Aufgrund der Ergebnisse der Evaluierung wurde im finalen @VISOR-Demonstrator stereoskopische Tiefe als Relevanzhinweis gewählt, optional in Verbindung mit einer Farb- und Intensitätsfilterung zur Vermeidung von visueller Überladung. Darauf wird bei der Beschreibung des finalen Demonstrators näher eingegangen.

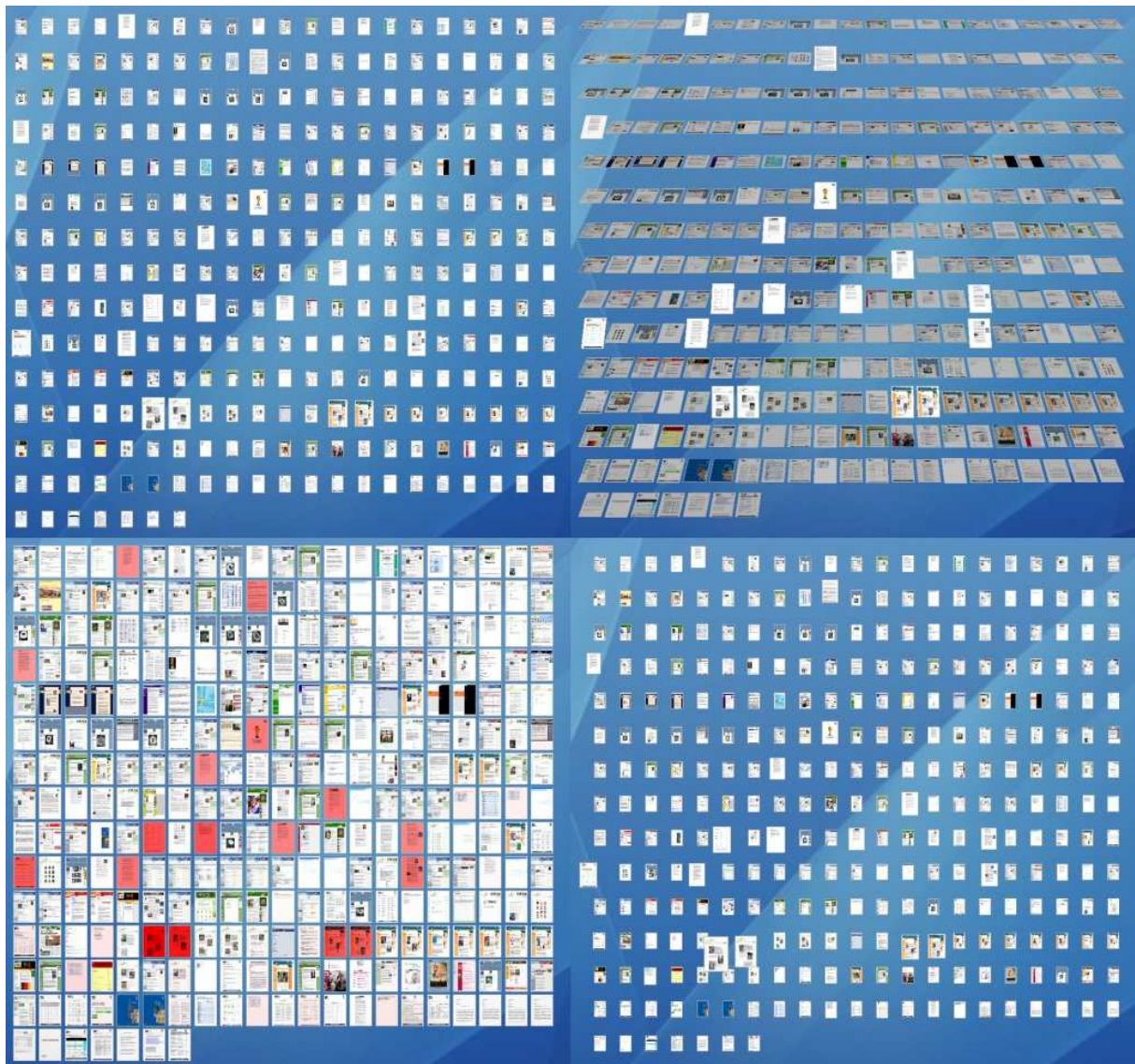


Abb. 11: Das gleiche Suchergebnis dargestellt mit verschiedenen präattentiven Merkmalen: Größe, Orientierung, Farbe/Intensität, und stereoskopische Tiefe.

## 5. Multimodale Interaktion und Navigation

In den die Interaktion und Navigation behandelnden Arbeitspaketen wurde im Projektzeitraum zunächst ein Überblick über bestehende Techniken zur Navigation in zwei- und dreidimensionalen Räumen erstellt. Insbesondere wurden verschiedene Methoden zur Interaktion mit und Manipulation von Objekten in 3D-Räumen auf ihre Anwendbarkeit und ihren Nutzen im Rahmen des Projektes @VISOR untersucht. Besonderes Augenmerk wurde dabei vorerst auf Techniken zum Auswählen und Ergreifen von Objekten gerichtet. Hierbei wurde auf der Grundlage bestehender Vergleiche der Techniken entschieden, dass für die Interaktion in @VISOR eine Kombination von einfachen *Arm Extension*-Techniken, bei denen der Benutzer mit Hilfe eines getrackten Datenhandschuhs die Objekte direkt auswählen und manipulieren kann, sowie einer besonderen Form von *Ray Casting* am Erfolg versprechendsten erscheint. Hierbei wird vom Augpunkt des Betrachters ein Strahl über den Auswahlcursor geschickt und das erste Objekt gewählt, das dieser Strahl schneidet. Aus diesem Grund bietet sich diese Methode an, um 3D-Objekte mit Hilfe eines 2D-Eingabegeräts wie einer Maus auswählen zu können.

Weiterhin wurden verschiedene im Rahmen von @VISOR benötigte Metaphern betrachtet und zentrale Thematiken wie das Blättern durch Dokumente oder Dokumentenstapel für eine detaillierte Evaluierung ausgewählt. Zu diesem Zweck wurden für Versuche mit vorhandener Hardware rudimentäre Evaluierungsumgebungen für vorerst zwei verschiedene Datenhandschuhe entwickelt. Zum einen wurde für einen 5DT-Datenhandschuh mit 16 Finger-Sensoren und ohne vorhandenen Tracking-Mechanismus eine Umgebung zur Erkennung statischer Gesten implementiert. Hierbei wurden zwei Ansätze getrennt verfolgt: Zum einen die Vorabdefinition von Gesten aufgrund prekärer Fingerstellungen und zum anderen das Lernen einer vom Benutzer ausgeführten Geste mit anschließendem Erkennen anhand geeigneter Ähnlichkeitsanalysen. Zusätzlich wurde eine Evaluierungsumgebung für einen P5-Datenhandschuh mit 5 Biegesensoren für die Finger, kombiniert mit einem optischen 6 DOF-Trackingsystem, implementiert. Nachdem zunächst verschiedene Glättungsfilter zur Vermeidung von Abtastungsfehlern und eine einfache Gestenerkennung aufgrund der Biegesensoren entwickelt wurden, lagen die nächsten Arbeitsschritte in der Auswahl, Umsetzung und Evaluierung verschiedener Metaphern für die Interaktion und Navigation beim Umgang mit Dokumenten in virtuellen Umgebungen.

### 5.1 Immersive 6DOF-Interaktion

Wie bereits erwähnt, ist einer der Schwerpunkte bei der Entwicklung des @VISOR-Projekts das Finden adäquater Methoden und Techniken zur Interaktion mit Dokumenten und Dokumenträumen. Ein hervorstechendes Kriterium zur Untersuchung und Evaluation geeigneter Interaktionsmetaphern ist der Immersionsgrad, der mit unterschiedlichen Interaktionsgeräten erreichbar ist. Der Zweck eines hohen Grades an Immersion ist das Potential, ein als "flow" bezeichnetes Phänomen herbeizuführen. Flow wird bevorzugterweise durch die dabei erfahrenen Symptome beschrieben, nämlich dem Verschmelzen von



Handlung und Bewusstsein, dem Ausblenden alles Anderen in den Hintergrund, dem Verlust des Zeitgefühls und dem Erreichen herausfordernder Ziele ohne Anstrengung. Eine Voraussetzung für das Erreichen von flow ist die Abwesenheit von Barrieren, die eine bewusste Aufmerksamkeit verlangen, wie das Bedienen von Kontrollelementen eines Programms.

Daraus ergibt sich, dass eine Mensch-Maschine-Schnittstelle das Herbeiführen von flow unterstützen kann, indem es die Barrieren zwischen Mensch und Computer vermindert. Unter diesen Gesichtspunkten wurden mehrere verfügbare Interaktionsgeräte hinsichtlich ihres Potentials untersucht, als eine immersive und intuitive Schnittstelle zwischen dem Benutzer und dem @VISOR-Framework zu dienen. Spezielles Augenmerk wurde dabei auf die Evaluation von Geräten gerichtet, die eigens zur Interaktion mit dreidimensionalen Umgebungen entwickelt wurden, sowie die neuen Interaktionsmetaphern, die sich durch zusätzliche Freiheitsgrade ergeben. Um diese neuen Möglichkeiten besser demonstrieren und evaluieren zu können, wurden einfache Demonstratoren für technische Versuchszwecke implementiert. Ziel war dabei, einen sanften Übergang zwischen Mensch und Computer zu realisieren, so dass der Benutzer mit der Anwendung auf eine natürliche und leicht zu erlernende Weise interagieren kann, wie er es von seinem Schreibtisch her gewohnt ist.

Der natürlichste Weg zur Manipulation der Umgebung ist natürlich das Benutzen der Hände. In den meisten Fällen erfolgt diese Interaktion ohne darüber nachzudenken und ohne andere Aufgaben dafür unterbrechen zu müssen. Der vielversprechendste Ansatz zur Minimierung der kognitiven Belastung, welche für das Erlernen und die Verwendung einer Benutzerschnittstelle in einer virtuellen Umgebung aufgebracht werden muss, ist daher eine Gestenerkennung, welche den Benutzer in einer natürlichen Art und Weise mit der Applikation interagieren lässt.

Ein Ziel in @VISOR war daher eine zuverlässige Echtzeitgestenerkennung, die auf jedem einigermaßen aktuellen Arbeitsplatz-PC lauffähig ist und sich leicht in normale Anwendungen integrieren lässt, ohne dabei übermäßig viel Rechenleistung zu verbrauchen. Wie in bekannten Ansätzen auch, verwenden wir eine Menge von Gesten, die durch Ausführen der jeweiligen Gesten gelernt worden ist. Jedoch definieren wir für unser System keine Gesten als Bewegung über einen gewissen Zeitraum, sondern als eine Sequenz von Handhaltungen, die an bestimmten Positionen mit bestimmten Orientierungen der Hand des Benutzers aufgenommen werden.

Die Data Glove-Hardware, die wir für die Gestenerkennungs-Engine verwenden, ist ein P5 Glove von Essential Reality, ein eigentlich für den Spielmarkt entwickelter Datenhandschuh. Der P5 wurde dabei aus zwei Gründen gewählt: Zum einen aufgrund seines niedrigen Preises von etwa 50 € - dies zeigt, dass brauchbare Interaktionshardware für virtuelle Umgebungen mittlerweile auch für den normalen Verbraucher erschwinglich ist. Zum anderen können wir durch die Verwendung der Consumer-Hardware zeigen, dass unsere Erkennungs-Engine leistungsfähig und flexibel genug ist, um eine zuverlässige

Gestenerkennung selbst bei Verwendung von preisgünstiger Spiele-Hardware an Stelle von sonst üblicher teurer Spezialhardware zu ermöglichen.

Die @VISOR-Gestenerkennungs-Engine ist in zwei Komponenten unterteilt: die Datenerfassung und den Gestenmanager. Die Datenerfassung läuft als separater Thread und überprüft kontinuierlich die vom Handschuh erhaltenen Daten auf mögliche Übereinstimmungen mit dem Gestenmanager. Da wie bereits erwähnt der P5 Consumer-Hardware darstellt, sind die von diesem gelieferten Positions- und insbesondere Orientierungsinformationen in der Regel stark verrauscht. Entsprechend wurde eine Methode entwickelt und vorgeschaltet, welche die erhaltenen Daten geeignet filtert und glättet, um eine für die Gestenerkennung ausreichende Übereinstimmung mit den bekannten Handhaltungen zu ermöglichen. Die von der Datenerfassung gewonnenen Informationen über Position und Orientierung werden in Form von Events direkt an die Hauptapplikation weitergereicht. Diese kann aufgrund der mitgelieferten Daten direkt auf bestimmte Ereignisse reagieren. Im Falle des @VISOR-Demonstrator beinhaltet dies z.B. das Nachführen der virtuellen Hand in der Visualisierung sowie das Umschalten in die einzelnen Arbeitsbereiche (siehe Kapitel 6), wenn die virtuelle Hand des Benutzers die Ränder der virtuellen Sicht berührt.



Abbildung 12: Immersives Drag'n'Drop.

Gleichzeitig werden die gewonnenen Daten an den Gestenmanager weitergereicht, der diese mit der gerade benutzten Gestenbibliothek abgleicht und prüft, ob die momentane Handhaltung des Benutzers mit einer gespeicherten Geste übereinstimmt. Dabei wird sowohl die Orientierung der Hand als auch die Biegung der einzelnen Finger berücksichtigt. Findet der Gestenmanager eine ausreichend genaue Übereinstimmung (der benötigte Grad der Übereinstimmung kann dynamisch geregelt werden), so sendet er ein weiteres Event an die Hauptapplikation. Dieses Event beinhaltet einen Zeitstempel, Position und Orientierung der Hand, sowie den Namen der erkannten Geste. So kann die Applikation flexibel auf jede Abfolge von Gesten reagieren und die betreffende Aktion auslösen.

Obwohl das @VISOR-Backend (vergl. Abschnitt 3.3) verschiedene Arten der Wichtigkeitsbewertung einzelner Dokumente zulässt, ist eine zusätzliche manuelle Einstufung der Wichtigkeit für den aktuellen Kontext des Benutzers unerlässlich. Aus diesem Grund wurde ein neues Visualisierungs- und Interaktionskonzept entwickelt, mit dessen Hilfe der Benutzer Dokumente nach der Relevanz für seine momentane Aufgabe einstufen kann, ohne dabei die vom System vorgeschlagene Ordnung zu zerstören.

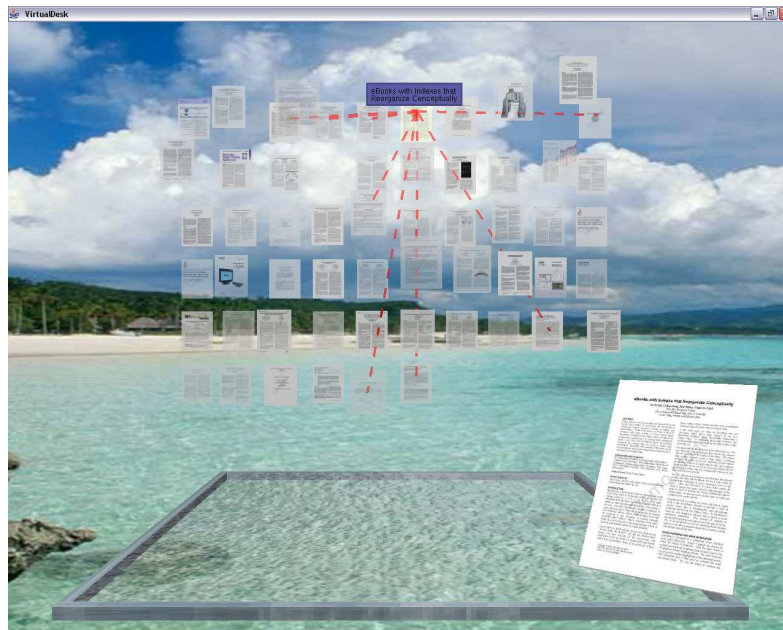


Abb. 13: Virtual Workdesk mit Ähnlichkeitsvisualisierung und manueller Einstufung der Wichtigkeit.

Der Demonstrator für dieses Konzept (Abb. 13) bestand während des DocumentPlane-Modus im Wesentlichen aus einer virtuellen Arbeitsplatte, auf der der Benutzer Dokumente ablegen, anordnen und zu Dokumentenstapeln zusammenfassen kann. Über der Platte schweben die Miniaturansichten von Dokumenten, die das Ergebnis einer Suchanfrage an das System darstellen. Die Anordnung der Dokumente geschieht zunächst auf einer senkrecht zum Benutzer stehenden Ebene im Raum mit der gleichen Reihenfolge, wie sie in einer 2D-Umgebung wäre. Das relevanteste Dokument befindet sich am oberen linken Rand der Dokumentgruppe, wobei die Relevanz nach rechts und nach unten (in dieser Reihenfolge) abnimmt. Zusätzlich wird das relative Alter der Dokumente durch den Grad ihrer Transparenz visualisiert. Ältere Dokumente werden zunehmend ausgeblendet, um neuere, aktuellere Dokumente in ihrer Wichtigkeit hervorzuheben. Natürlich lässt sich diese Metapher auch auf ein beliebiges, vom Benutzer festgelegtes Alter anwenden, so dass Dokumente mit zunehmendem Altersunterschied zum festgelegten Alter transparenter werden.

Führt der Benutzer nun mit der Maus oder – im Falle der Benutzung eines Datenhandschuhs – mit seiner virtuellen Hand über ein Dokument, werden sowohl nähere Informationen zum gewählten Dokument als auch Ähnlichkeiten zu anderen Dokumenten des Suchergebnisses angezeigt. Die Strichenge der Linien repräsentiert dabei den Ähnlichkeitsgrad zum gewählten Dokument.

Der Benutzer kann nun das betreffende Dokument "angreifen" und – je nach seiner Einschätzung der Wichtigkeit – zu sich hin ziehen oder von sich weg schieben. Die ähnlichen Dokumente werden dabei, abhängig vom Grad der Ähnlichkeit, ebenfalls in ihrer räumlichen Position nach vorne oder hinten korrigiert. Sobald der Benutzer das fokussierte Dokument wieder loslässt, wird es an der aktuellen Position verankert, um eine versehentliche Veränderung durch Bewegen eines ähnlichen Dokuments zu verhindern. Hat der Benutzer auf diese Art einige der im Suchergebnis befindlichen Dokumente bewertet, so ergibt sich für die weiteren Ergebnisdokumente bereits eine vorläufige Bewertung, basierend auf der vom User vorgenommenen Einstufung, wobei die ursprüngliche Ordnung in x/y-Richtung unverändert bleibt. Zusätzlich kann der Benutzer einzelne Dokumente aus der Gruppe herausgreifen und eine Kopie des Dokuments auf seiner Arbeitsplatte positionieren oder zu einem bereits vorhandenen Dokument oder Stapel hinzufügen, um die Ergebnisse seiner Suche auf eine für ihn hilfreiche Weise zu gruppieren.

## 5.2 Taktiler Feedback

Ein anderer Betrachtungsansatz für eine multimodale Interaktion und speziell für die Navigation durch Dokumente ist die Berücksichtigung von haptischem Feedback, um dem Benutzer zusätzliche Informationen über seine virtuelle Umgebung und den Inhalt von Dokumenten zuzuführen. Beispielsweise kann ein Benutzer, der ein Dokument auf der Suche nach speziellen Informationen durchblättert, durch einen geeigneten haptischen Effekt darauf aufmerksam gemacht werden, dass er sich gerade in der Nähe von für ihn relevanten Informationen befindet. Es wurden verschiedene Evaluationen durchgeführt, um die Auswirkung verschiedener Effekte auf die Fähigkeit eines Benutzers zu testen, eine bestimmte Stelle innerhalb eines Dokuments zu finden.

Visuelles Interaktionsfeedback ist bei vielen Anwendungen im Bereich der virtuellen Realität der Standard. Daneben gibt es weitere Möglichkeiten, dem Benutzer Rückmeldung über den aktuellen Systemzustand und Vorgänge innerhalb der Simulation zu vermitteln. Kombiniert man verschiedene Ausgabeformen, so erhält man eine multimodale Interaktion, die mehrere Sinne des Anwenders anspricht. Im Projektzeitraum wurden Konzepte entwickelt, um auch den Tastsinn (haptisches Feedback) beim Interaktionsfeedback ansprechen zu können. Haptisches Interaktionsfeedback kann durch geeignete haptische Ausgabegeräte wie beispielsweise einen Force-Feedback-Joystick erzeugt werden und soll dem Benutzer helfen, Aufgaben in einer virtuellen Umgebung besser und effizienter erledigen zu können. Haptisches Feedback kann dabei unterstützend zu visuellem Feedback eingesetzt werden.

Um von dem verwendeten Force-Feedback-Gerät weitgehend unabhängig zu sein, bietet sich auf Windows Plattformen DirectX, genauer DirectInput, an. Im @VISOR-Kontext wurde zur Untersuchung der Force-Feedback-Möglichkeiten ein Joystick eingesetzt, da die Anwendung unkompliziert ist und die meisten Benutzer zumindest mit gewöhnlichen Joysticks bereits Erfahrung haben. Mit einem FF-Joystick können alle notwendigen grundlegenden Kräfte simuliert werden – es kann somit also evaluiert werden, ob diese Effekte vom Benutzer als nützlich empfunden werden oder nicht. Allerdings sind die FF-Effekte nicht so intensiv wie bei anderen Geräten und auch auf den Joystick selbst beschränkt, der wiederum ein stationäres Gerät ist. Dieser Umstand schränkt den Bewegungsspielraum des Anwenders ein, was jedoch im @VISOR-Kontext (Single-User-Workspace) keine negativen Auswirkungen hat.

Ein mögliches Force-Feedback-Navigations-Szenario für den dreidimensionalen Dokumentenraum ist der *Pilot's View*, bei welcher sich der Anwender wie in einem Flugzeug durch diesen virtuellen Raum bewegen kann. Dabei werden für die einzelnen Objekte oder Gruppen jeweils gegebene Relevanzen vorausgesetzt, welche sich beispielsweise über eine Suche und deren Auswertung im @VISOR-Backend (vergl. Abschnitt 3.3) definieren. Hierfür wurde das Konzept des Relevanzfeldes (engl. *field of relevance*) entwickelt. Dieses Relevanzfeld ist an dem physikalischen Phänomen der Gravitation angelehnt und definiert für jeden Punkt im virtuellen Raum einen Grad der Relevanz und einen Anziehungsvektor. Statt der Massen, die sich bei der Gravitation anziehen, sind es bei Relevanzfeldern die Dokumente im Raum, die den Anwender anziehen. Je größer die Relevanz des jeweiligen Dokuments, desto größer ist entsprechend auch die Anziehungskraft, die auf den Benutzer ausgeübt wird. Der Faktor *Relevanz* legt dabei fest, wie stark der Benutzer vom entsprechenden Dokument angezogen wird. Der „beste“ Treffer bei der Suchanfrage erhält entsprechend die größte Relevanz und übt auch die größte Anziehung aus. Das *Wirkungsfeld* beschreibt, ab wann der Benutzer von einem Dokument angezogen wird. Hier ist es sinnvoll, eine Kugel mit einem bestimmten Radius um den Mittelpunkt des Dokuments zu definieren. Der *Abschwächungsfaktor* legt zudem fest, inwieweit sich die Anziehungskraft in Abhängigkeit des Abstands vom Objekt-Mittelpunkt abschwächt. Abbildung 14 zeigt, wie man sich das Relevanzfeld im dreidimensionalen Raum vorstellen kann. Die Dokumente sind in diesem Entwurf als einfache Quader dargestellt, deren Tiefe die Seitenzahl veranschaulicht. Zur besseren Darstellbarkeit liegen in diesem Beispiel alle Dokumente in einer Ebene. Ein Querschnitt aus dem Raum ist als Gittermodell zu sehen und zeigt, wie die einzelnen Dokumente den Raum um sich herum krümmen, je nachdem wie relevant diese sind. Diese aus der Physik bekannte Veranschaulichung der Raumkrümmung passt auch gut auf das Konzept des Relevanzfeldes, da man sich gut vorstellen kann was passiert, wenn man in die Nähe eines solchen Dokuments kommt und angezogen wird.

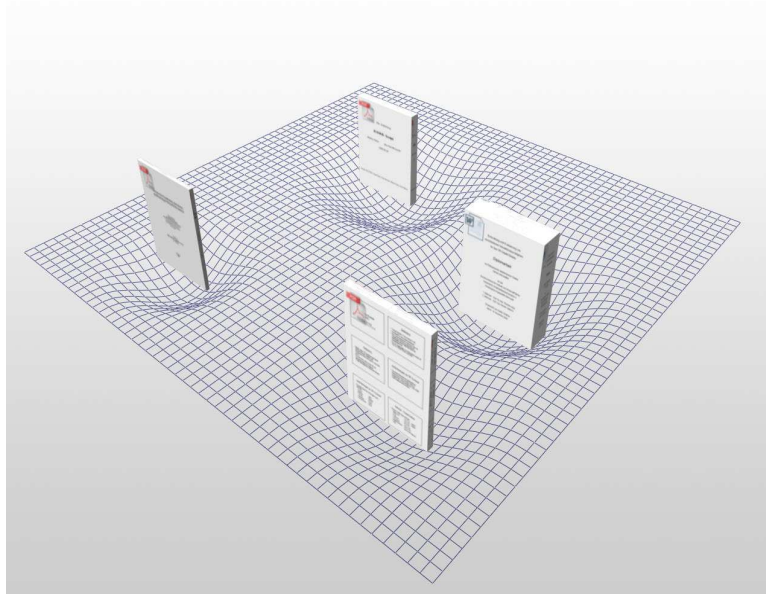


Abb. 14: Konzept des Relevanzfeldes im 3D-Dokumentenraum.

Diese Anziehungskraft wird nun von einem FF-Joystick simuliert. Fliegt der Anwender durch den Dokumentenraum und kommt in den Einflussbereich eines Dokuments, so bewegt sich der Joystick in die Richtung des Anziehungsvektors. Diese Kraft wird immer stärker, je näher man am Objekt ist. Der Benutzer muss eine gewisse Gegenkraft ausüben, um sich wieder von dem Dokument zu entfernen. Dem Anwender soll so klargemacht werden, dass je wichtiger das Dokument ist, desto mehr Kraft muss er aufwenden, um sich davon zu entfernen.

Hat ein (Text-)Dokument mehrere Seiten, die nicht mehr alle auf einmal dargestellt werden können, so muss für den Benutzer die Möglichkeit geschaffen werden, durch das Dokument zu blättern. Eine Variation des Blätterns wurde in @VISOR-Projekt so implementiert, dass der Benutzer als Geste den Daumen nach oben streckt und die Hand dann nach links oder rechts dreht, um eine Seite zurück oder vor zu springen. Durch den Datenhandschuh kann diese Bewegung erkannt werden und das Dokument wird eine Seite nach vorne oder zurück geblättert. Dieses Konzept funktioniert prinzipiell sehr gut, hat aber einen entscheidenden Nachteil: Wenn mehrere Seiten geblättert werden sollen, so muss der Benutzer seine Hand immer wieder in die Ausgangsstellung bewegen, um anschließend wieder eine Seite zu blättern. Ein kontinuierliches Blättern ist also mit dieser Technik unmöglich.

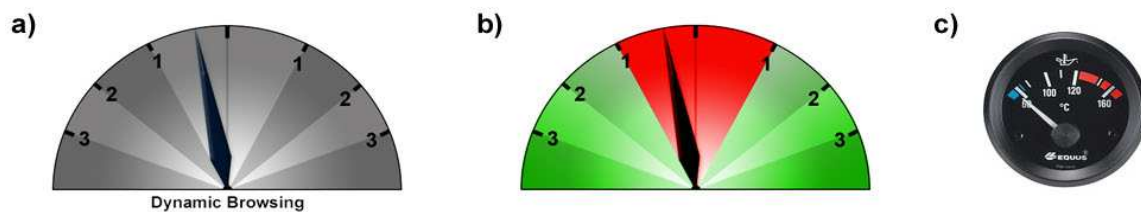


Abb. 15: Visuelles Feedback beim Dynamic Browsing (a,b) angelehnt an Messinstrumente (c).

Daher wurde das Konzept *Dynamic Browsing* entwickelt: auch hier wird als Geste, welche die Aktion des Blätterns einleitet, der nach oben zeigende Daumen verwendet. Allerdings gibt es beim Dynamic Browsing eine Abstufung, wie weit die Hand anschließend nach links bzw. rechts bewegt wird. Der Drehwinkel der Hand legt fest, wie viele Seiten pro Sekunde geblättert werden. Dies heißt zum einen, dass einerseits die Geschwindigkeit beeinflusst werden kann (Anzahl der Seiten pro Sekunde) und andererseits auch ein kontinuierliches Blättern ermöglicht wird, da in jeder Sekunde die eingestellte Anzahl an Seiten geblättert wird. Um den Benutzer bei dieser Navigationsaufgabe innerhalb des Dokuments zu unterstützen, ist ein visuelles Feedback sinnvoll. Abbildung 15 zeigt eine mögliche, an gängige Anzeigen von Messgeräten angelehnte Visualisierung, die dem Benutzer helfen kann. Der Benutzer bekommt hierdurch eine visuelle Rückmeldung darüber, wie weit seine Hand bereits gedreht ist und muss seinen Blick dazu nicht vom Anzeigegerät abwenden.

Die Technik des Dynamic Browsing ist nicht auf die Anwendung mit einem Datenhandschuh beschränkt, sondern kann auch mit anderen Eingabegeräten verwendet werden. Insbesondere kann diese Technik mittels Force-Feedback erweitert und dann mit einem entsprechenden Joystick sehr effizient benutzt werden. Das haptische Feedback liefert dem Benutzer zusätzliche Informationen zur Unterstützung bei der Orientierung innerhalb des Dokumentenraumes. Für die Einbindung des Force-Feedback-Effekts sind zwei Anwendungsmöglichkeiten denkbar: Beim „normalen“ Blättern durch ein Dokument kann eine Gegenkraft erzeugt werden, die linear ansteigt, je weiter der Joystick nach außen bewegt wird. Somit ist es für den Anwender schwerer, mehrere Seiten auf einmal zu blättern (analog zu der realen Welt, in der es auch aufwendiger ist, mehrere Seiten auf einmal umzublättern). Somit kann der Benutzer eine direkte Assoziation herstellen: je mehr Gegendruck er ausüben muss, desto mehr Seiten werden auf einmal geblättert. Eine zweite Anwendungsmöglichkeit des Force-Feedback-Effekts ergibt sich, wenn man ein Dokument durchblättert, in dem zuvor eine Textsuche durchgeführt wurde. Durch die Suchanfrage entstehen im Dokument relevante und weniger relevante Bereiche, je nachdem wie groß die Trefferzahl für den eingegebenen Suchbegriff ist. Befindet sich der Benutzer in einem Bereich des Dokuments, in dem eine hohe Relevanz vorliegt, so wird eine Gegenkraft erzeugt, wenn der Anwender im Dokument weiterblättern möchte. Es soll also für den Nutzer spürbar gemacht werden, dass er sich in einem relevanten Bereich bezüglich der Suchanfrage aufhält und die Gegenkraft verdeutlicht, dass nun dieser Bereich verlassen

wird. So merkt der Benutzer intuitiv, wo die für ihn wichtigen Textpassagen sind, auch ohne den Bildschirm nach den als relevant markierten Stellen abzusuchen. Diese Force-Feedback-Technik ist eine Abwandlung der Technik des Relevanzfeldes, die sich in diesem Falle auf die eindimensionale Operation des Blätterns bezieht. Weiterhin könnte die relevanteste Seite des Dokuments (beispielsweise diejenige mit den meisten Treffern) durch ein Rütteln des Joysticks hervorgehoben werden. So wäre selbst bei schnellem Blättern zu erkennen, wo die Suchanfrage einen „exakten“ Treffer produzierte.

### 5.3 Dimensional Congruence

Ein für das Gesamtkonzept von @VISOR äußerst wichtiges Prinzip der „Dimensional Congruence“. Dieser Begriff bezieht sich speziell auf visuelle Interfaces und virtuelle Umgebungen, bei denen situationsbedingt eine Darstellung in 2D oder in 3D wünschenswert ist. Dabei sollten sowohl Darstellung als auch Interaktion mit der Umgebung jeweils an den gerade durchgeführten Task angepasst sein. In anderen Worten: In Situationen, in denen eine 3D-Visualisierung vorteilhaft ist, sollte eine solche Darstellung verbunden mit passender 3D-Interaktion erfolgen, jedoch sollte eine Visualisierung, die nur zwei Dimensionen erfordert, auch in einer solchen Umgebung visualisiert und die Interaktion entsprechend angepasst werden. Dies war bei @VISOR in besonderem Maße der Fall. Für die natürliche Interaktion mit mehreren Dokumenten sowie für die intuitive Anordnung von Dokumenten entsprechend ihrer Relevanz zeigte sich die Darstellung der Umgebung in stereoskopischem 3D als sehr vorteilhaft. Jedoch ergeben sich gerade beim Umgang mit Dokumenten oft Situationen, für die eine Darstellung in 2D nicht nur ausreichend, sondern auch vorteilhafter ist. Speziell bei der Arbeit an einem 3D-Monitor, wie er bei @VISOR verwendet wurde, wird das Darstellen stereoskopischer Tiefe ohne zusätzliche Hardware mit einem Verlust an Auflösung erkauft, was das Lesen von Dokumenten unangenehm erschwert. Auch führt das Darstellen von Text auf einer anderen stereoskopischen Tiefe als der tatsächlichen Entfernung des Displays bei längerer Betrachtung zu unnötiger Belastung der Augen.

Aus diesem Grund wurden im @VISOR-Projekt die unterschiedlichen Aufgaben und Interaktionen des Benutzers daraufhin untersucht, ob diese besser in 3D oder in 2D funktionieren, und der finale Demonstrator entsprechend angepasst. So ergab sich beim Aufbau, dass sowohl die Darstellung als auch die Interaktion auf zwei dimensional kongruente Arten durchgeführt wurden. Für stereoskopische Darstellung und Interaktion kamen dabei die bereits vorgestellte Kombination aus 3D-Display und 6DOF-Datenhandschuh zum Einsatz. Für naturgemäß zweidimensionale Aufgaben wie Lesen und Bearbeiten von Dokumenten sowie Texteingabe, wurde der Aufbau um via Drahtlosnetzwerk verbundenen Tablet PC erweitert. Dieser ermöglicht nicht nur das Darstellen von hoch aufgelöstem Text, der Benutzer kann dargestellte Dokumente auch wie normale Dokumente physikalisch aufheben und in einer für ihn bequemen Haltung lesen. Zusätzlich ermöglicht die Stiftsteuerung des Tablet-PC eine schnelle und intuitive Eingabe von Text. Dabei hat der Stift den weiteren Vorteil, dass seine Benutzung durch einen getragenen Datenhandschuh nicht behindert wird (vgl. Abbildung 16).



## 6. Finaler Demonstrator



Abb. 16: Aufbau des finalen Demonstrators: 3D-Bildschirm, Datenhandschuh und Tablet-PC.

In der letzten Projektphase von @VISOR wurden die erarbeiteten Konzepte und gewonnenen Erkenntnisse in einem finalen Demonstrationsaufbau umgesetzt. Wie bereits im Kapitel über Dimensional Congruence beschrieben, bestand der Hardware-Aufbau des Szenarios aus Display- und Interaktionsgeräten sowohl für den zweidimensionalen als auch für den dreidimensionalen Einsatz. Zur Darstellung von 3D-Inhalten wurde ein stereoskopisches Cn-Display von SeeReal verwendet, die Interaktion mit der auf diesem Display dargestellten virtuellen Umgebung erfolgte mit einem P5 Datenhandschuh. Einzelne Dokumente sowie Eingabefelder für Text wurden auf einem zusätzlichen 2D Interface auf einem per Netzwerk verbundenen Toshiba Tablet PC dargestellt.

Der wesentliche Teil des Demonstrationsszenarios spielt sich in einer virtuellen Arbeitsumgebung auf dem stereoskopischen Display ab. Der virtuelle Arbeitsplatz besteht aus drei Hauptbereichen, zwischen denen der Benutzer durch einfaches Bewegen seiner Hand in die entsprechende Richtung umschalten kann. Die drei Arbeitsbereiche sind dabei im virtuellen Raum egozentrisch um die Position des Benutzers herum angeordnet, so dass ein Wechsel zwischen den Arbeitsbereichen fließend und intuitiv erfolgt, indem einfach die virtuelle Sicht des Benutzers gedreht und auf den aktiven Arbeitsbereich gerichtet wird.



Abb. 17: Der Dokumenten- und Suchbereich.

Im Anfangszustand wird der Hauptarbeitsbereich, der so genannte Dokumenten- und Suchbereich angezeigt. Hier werden alle indizierte Dokumente zunächst als Draufsicht in einer Ebene angezeigt. Die x- und y-Position der Dokumente bleibt dabei während der gesamten Interaktion unverändert, so dass der Benutzer außer dem visuellen Wiedererkennen auch sein räumliches Gedächtnis nutzen kann, um ein Dokument jederzeit schnell wieder zu finden (Abb. 17). Am unteren Rand der Darstellung ist ein Teil des zweiten Bereiches, des so genannten Arbeitsplattenbereiches, zu sehen. Diese geringe Überlappung der Arbeitsbereiche dient zu Orientierung des Benutzers, in welche Richtung er seine Hand bewegen muss, um zum entsprechenden Bereich zu gelangen. Ebenso verhält es sich mit dem dritten Bereich zur Rechten des Benutzers, der Pinnwand.

Der normale Ablauf eines typischen Tasks beginnt in der Regel im Dokumentenbereich. Hier kann der Benutzer durch Aktivieren eines Widgets im oberen Bereich der Darstellung eine Suchanfrage starten. Dabei wird für die Dauer der Suchanfrage die Interaktion mit dem Datenhandschuh ausgesetzt und ein entsprechendes 2D-Interface auf dem Tablet-PC aktiviert, in dem der Benutzer ein oder mehrere Suchanfragen eingeben kann. Die stereoskopische Darstellung reagiert auf die getätigten Anfragen direkt, indem Dokumente aus der Ebene heraus nach vorne oder hinten verschoben werden, basierend auf der Relevanz des betreffenden Dokuments für die Gesamtsuche. Zusätzlich erscheinen auf der linken Seite der Dokumentenebene Repräsentationen der einzelnen Suchanfragen, so dass einerseits die erfolgten Suchanfragen zurückverfolgt werden können (History), andererseits aber auch mit den einzelnen Suchtermen interagiert werden kann. Sobald der Benutzer mit der Suchanfrage zufrieden ist, kann er das 2D-Interface durch einen Klick mit dem Stift deaktivieren und gelangt zurück zur Datenhandschuh-Steuerung. Hier kann er nun auf

verschiedenste Weisen mit den dargestellten Dokumentrepräsentationen interagieren. Bewegt er seine virtuelle Hand zu einem Dokument, so werden zusätzliche Metadaten in einem Hilfsfenster (3D-Popup) angezeigt. Zusätzlich werden von dem gewählten Dokument ausgehende Verbindungen zu Dokumenten mit ähnlichem Inhalt angezeigt. Auf diese Weise kann der Benutzer für seinen momentanen Suchkontext interessante Dokumente finden, auch wenn diese aufgrund seiner initialen Suchanfrage als wenig relevant eingestuft wurden.

Zusätzlich kann er die Einstufung jedes Dokuments auch manuell korrigieren, indem er das Dokument mit seiner virtuellen Hand ergreift und nach vorne zieht oder nach hinten schiebt. Abhängig von den im Benutzerkontext gewählten Einstellungen bewegen sich dabei auch Dokumente ähnlichen Inhalts nach vorne und hinten, so dass der Benutzer den Suchkontext individuell anpassen kann. Diese und andere Einstellungen des Benutzerkontexts können über ein Einstellungsmenü geändert werden, welches als Widget im linken unteren Bereich der Dokumentenansicht repräsentiert wird. Zeigt der Benutzer auf dieses Widget, öffnet sich ein Ringmenü mit verschiedenen Einstellungsmöglichkeiten. Der Nutzer kann seine Hand in Richtung des entsprechenden Ringbogens bewegen, um einen Menüpunkt auszuwählen. Ähnliche Kontextmenüs kann der Benutzer zu jedem Dokument und Dokumentenstapel aufrufen. Eine weitere Möglichkeit zur Interaktion im Dokumentenbereich besteht in der Option, eine Filterebene zur Verminderung visueller Überhäufung einzublenden. Diese Filterebene steht senkrecht zur Blickrichtung des Benutzers, so dass weiter hinten liegende, Unwichtige Dokumente in stark verminderter Intensität dargestellt werden. Der Benutzer kann die Empfindlichkeit des Filters auf intuitive Art einstellen, indem er einfach die Filterebene nach vorne oder hinten bewegt.

Besonders wichtig für @VISOR ist die Möglichkeit des Benutzers, Dokumente zu gruppieren, diese Gruppen räumlich anzuordnen und schnell zwischen verschiedenen Tasks wechseln zu können. All diese Funktionen werden durch den zweiten Arbeitsbereich, den Arbeitsplattenbereich, zur Verfügung gestellt. Hier steht dem Benutzer eine virtuelle Tischplatte zur Verfügung, auf der er Dokumente und Dokumentgruppen anordnen kann, wie er es von der realen Welt gewöhnt ist. Er kann einzelne oder mehrere Dokumente aus dem Dokumentenbereich greifen und auf die Tischplatte ziehen, wo diese zu als Dokumentenstapel dargestellten Gruppen zusammengefasst werden (Abb. 18). Je nach Einstellung des Benutzerkontexts können dabei auch automatisch alle zum gewählten Dokument ähnlichen Dokumente mit auf dem Stapel abgelegt werden. Zusätzlich kann der Benutzer auch eine oder mehrere der Suchterm-Repräsentationen greifen und auf die Tischplatte ziehen, um alle zu einem bestimmten Begriff gefundenen Dokumente zusammenzufassen. Die Auswahl mehrerer Suchterme ergibt dabei ein logisches Oder der gewählten Suchanfragen. Auch mit den so gebildeten Stapeln kann der Benutzer interagieren, wie er es von realen Stapeln gewohnt ist. Er kann existierende Stapel durch einfaches Ziehen auf der Tischplatte anordnen und gruppieren, er kann Stapel durchblättern und er kann Stapel durch zusätzliche Dokumente ergänzen, indem er Dokumente oder Stapel auf einen bestehenden Stapel ablegt.

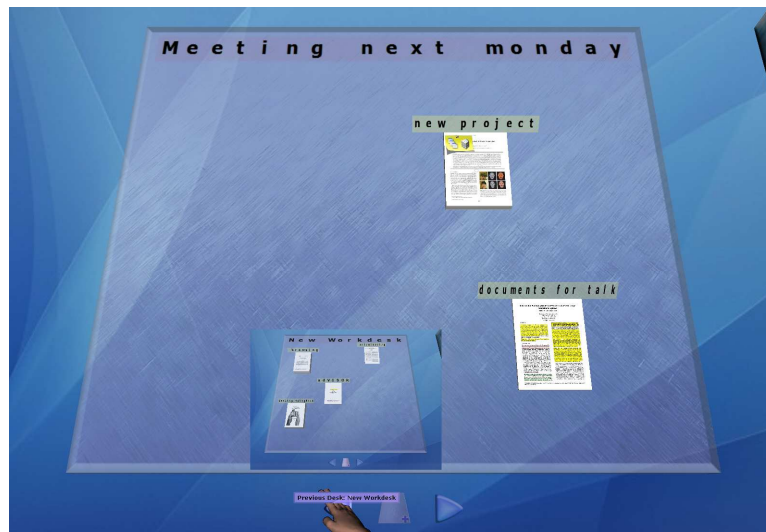


Abb. 18: Der WorkDesk-Bereich.

Zusätzlich zu seiner Funktion als Ablage und Ordnungsbereich repräsentiert der virtuelle WorkDesk aber auch den momentanen Task des Benutzers. Der Benutzer kann einen neuen Task beginnen, indem er ein entsprechendes Widget am unteren Rand des Arbeitsplattenbereichs aktiviert. Dabei wird der aktuelle WorkDesk einschließlich der erstellten Stapel und Suchanfragen abgespeichert, der Dokumentenbereich auf den Ausgangszustand zurückgesetzt und ein neuer, leerer WorkDesk angezeigt. Zwischen existierenden, automatisch mit Screenshots versehenen Tasks kann der Benutzer durch zusätzliche Widgets schnell und komfortabel wechseln (Abb. 18).

Unverändert von Änderungen des Tasks bleibt dabei der dritte Arbeitsbereich, die Pinnwand. Diese dient als persistente Ablage von Dokumenten, Dokumentengruppen und Notizen zu Dokumenten. Diese werden als an die Pinnwand angeheftete Post-Its visuell repräsentiert. Diese Post-Its werden erstellt, indem der Benutzer ein Dokument oder einen Stapel auf die Pinnwand zieht, ein Post-It per Kontextmenü an ein Dokument oder einen Stapel heftet, oder ein Bookmark in einem geöffneten Dokument anlegt. Die Post-Its werden dabei sowohl auf der Pinnwand als auch auf den betreffenden Dokumenten und Stapeln angezeigt. Einzelne Post-Its können frei auf der Pinnwand angeordnet werden, so dass der Benutzer auch hier sinnvolle Gruppierungen erstellen kann (Abb. 19). Der Text eines solchen Post-Its wird in die folgende Suche einbezogen, so dass mit diesem Post-It verknüpfte Dokumente bei entsprechenden Suchanfragen ein höheres Ranking erhalten und schnell wieder gefunden werden können. Zusätzlich können Post-Its auch auf die Arbeitsplatte gezogen werden, wo ein Stapel mit allen mit dem Post-It verknüpften Dokumenten gebildet wird. Auf diese Weise kann der Benutzer auch bereits gebildete Suchanfragen und Dokumentenstapel in verschiedenen Tasks nutzen, da die Pinnwand ihren Zustand unabhängig von Taskwechseln beibehält.

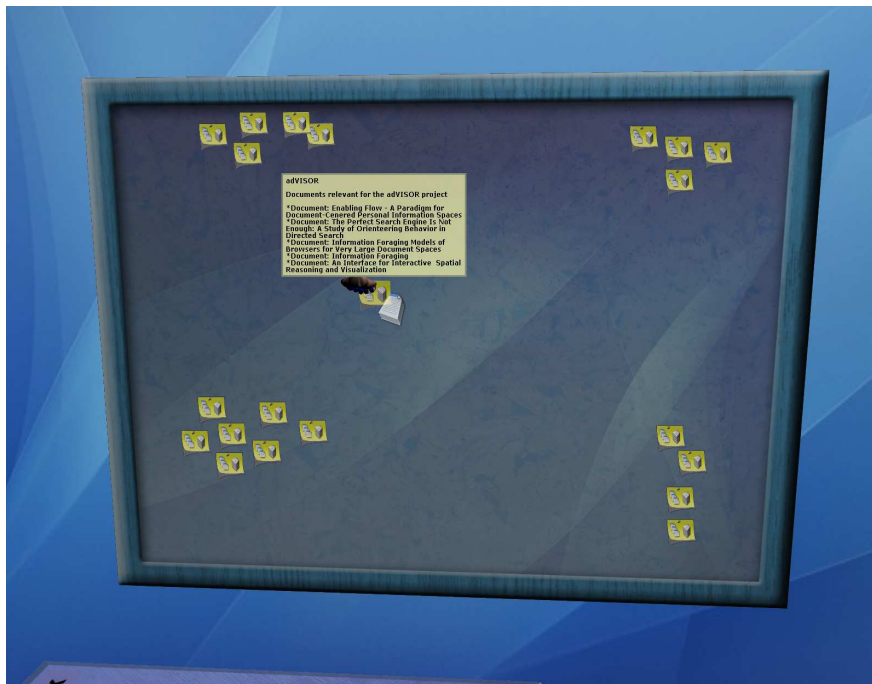


Abb. 19: Die virtuelle Pinnwand.

Schließlich hat der Benutzer natürlich auch die Möglichkeit, in Stapeln und einzelnen Dokumenten zu blättern, sowie diese zu bearbeiten. Das Öffnen von Stapeln und Dokumenten geschieht dabei durch eine entsprechende Geste (Greifen des Dokuments, anschließendes Öffnen der Hand mit der Handfläche nach oben) oder durch Benutzung des Kontext-Ringmenüs. Wird ein Stapel zum Blättern geöffnet, so wird der Stapel neben die Arbeitsfläche bewegt. Die Dokumente des Stapels bewegen sich der Reihe nach zum Benutzer hin, dann auf einen neuen Stapel auf der anderen Seite der Arbeitsfläche (Abb. 20). Dabei kann der Benutzer mit den Dokumenten des Stapels interagieren wie mit Einzeldokumenten, er kann also die Dokumente zum Lesen öffnen oder Dokumente aus dem Stapel herausziehen und auf andere Stapel auf der Arbeitsplatte ablegen.

Zum Blättern durch Dokumente hat der Benutzer zwei Möglichkeiten: Zum einen kann er eine schnelle Vorschau des Dokuments in der stereoskopischen 3D-Umgebung aufrufen. Dabei werden die Seiten des Dokuments auf den Seitenflächen eines oktagonalen Prismas abgebildet. Damit ergibt sich zum einen eine natürliche Focus & Kontext-Ansicht, da der Benutzer außer der aktuellen Seite im Mittelpunkt auch die vorhergehende und nachfolgende Seite im Blick hat. Zum anderen ergibt sich aus dieser Darstellung eine intuitive Blättermetapher: Zum Blättern in eine Richtung wird einfach die betreffende Seite des Prismas „angetippt“. Zusätzlich kann das Prisma auf die Kopfseite gedreht werden, wo alle Seiten des Dokuments in einer Übersicht dargestellt werden. Durch Antippen einer Seite wird direkt zu dieser Stelle des Dokuments gewechselt.



Abb. 20: Blättern in Stapeln und einzelnen Dokumenten.

Die zweite Methode zum Blättern besteht darin, das Dokument für längeres Lesen direkt auf dem sekundären Display, also dem Tablet PC zu öffnen. Geschieht dies, wird die Interaktion mit dem Datenhandschuh ausgesetzt, während der Benutzer das Dokument in voller Auflösung lesen und auch bearbeiten kann. Hat der Benutzer die Bearbeitung des Dokuments beendet, kann er durch einen entsprechenden Button in der Bedienoberfläche des Tablet PC zur 3D-Umgebung zurückwechseln. Änderungen im Dokument werden dabei direkt in der 3D-Repräsentation übernommen.

## 7. Evaluierung und Demonstrationsszenarien

Im Einklang mit den Empfehlungen des SAB wurde eine intensive Evaluierung des @VISOR-Konzeptes und der angedachten technologischen Umsetzung ausgearbeitet und durchgeführt. Der @VISOR-Evaluierungsplan besteht aus vier Hauptphasen, die im Folgenden tabellarisch dargestellt sind. Ziel aller Phasen ist dabei die Schaffung einer projektbegleitenden Qualitätssicherung.

### Phase I: Zu Beginn des Forschungsprojektes

<b>Ziel</b>	<p><b>Untersuchung der Benutzerbedürfnisse unter Berücksichtigung der Projektziele laut Projektantrag</b></p> <ul style="list-style-type: none"> <li>• Was ist dem Benutzer wichtig?</li> <li>• Welche Defizite sieht er in den bestehenden Lösungen?</li> <li>• Was mögen bzw. lehnen die Benutzer ab bezogen auf bestehende Lösungen?</li> <li>• Welche Features und Funktionalitäten sind für alle Benutzerrollen wichtig (Schnittmenge)?</li> <li>• Stimmt das Konzept mit den Ergebnissen der ersten Evaluation überein?</li> </ul>
-------------	--

<b>Methoden</b>	<ul style="list-style-type: none"> <li>• <b>Feature Focus</b> – Features und Funktionen werden den Testpersonen vorgestellt (verbal oder mit Skizzen). Sie werden aufgefordert diese zu bewerten und nach Wichtigkeit zu klassifizieren.</li> <li>• <b>Comparative Evaluation</b> – die Testpersonen werden aufgefordert bestimmte Tätigkeiten an existierenden Anwendungen auszuführen und diese zu bewerten und miteinander zu vergleichen.</li> </ul>
<b>Material</b>	<ul style="list-style-type: none"> <li>• Beschreibungen (und Skizzen) der geplanten Features und Funktionen</li> <li>• Fragebogen zur Bewertung und Klassifizierung</li> <li>• Zugang zu ähnlichen bereits existierenden Lösungen bereitstellen</li> </ul>

### Phase II: Am Ende der Konzeptions- und Designphase der Kernfunktionen

<b>Ziel</b>	<p><b>Überprüfung des Konzeptes und der aktuellen Entwicklungsarbeit</b></p> <ul style="list-style-type: none"> <li>• Überprüfung der geplanten Features und Funktionen</li> <li>• Entsprechen die angebotenen Features und Funktionen den Benutzerwünschen?</li> <li>• Wie intuitiv verständlich sind die Metaphern bzw. die Darstellung allgemein und die Benutzerinteraktion?</li> <li>• Feedback zum graphischen Design</li> <li>• Überprüfung der Navigation</li> <li>• Wie intuitiv verständlich sind die dynamischen Effekte?</li> </ul>
<b>Methoden</b>	<ul style="list-style-type: none"> <li>• <b>Design Walkthrough</b> – Evaluation anhand einer geleiteten Tour mit Hilfe von Screenshots oder Papierprototypen. Die Testpersonen werden nach ihrer Meinung zum Prozessablauf, „overall look and feel“, Navigation, Terminologie, Farben und Graphik gefragt.</li> <li>• <b>Content Aggregation</b> – Evaluation anhand von sinnvollen Aufgabenstellungen. Die Testpersonen sollen anhand von einfachen Prototypen Arbeitsaufträge ausführen. Dabei wird jeder Arbeitsschritt von Beobachtern protokolliert oder mit Video aufgezeichnet. (Evtl. kann danach ein Erinnerungstest durchgeführt werden.)</li> </ul>
<b>Material</b>	<ul style="list-style-type: none"> <li>• Prototypen als Screenshots oder als Skizzen (paper mockups)</li> <li>• Aufgabenkatalog</li> <li>• Protokollbogen/Videokamera</li> <li>• Bewertungsfragebogen</li> </ul>

### Phase III: Während der Implementierungsphase

<b>Ziel</b>	<p><b>Test anhand von Demonstratoren</b></p> <ul style="list-style-type: none"> <li>• Welche Features und Funktionen müssen zusätzlich berücksichtigt werden?</li> <li>• Wie intuitiv verständlich sind die Metaphern bzw. die Darstellung allgemein und die Benutzerinteraktion?</li> <li>• Überprüfung graphisches Design und Navigation</li> <li>• Werden alle benötigten Meta-Informationen bereitgestellt?</li> </ul>
<b>Methoden</b>	<ul style="list-style-type: none"> <li>• <b>Content Aggregation</b> – (siehe Phase II)</li> <li>• <b>Comparative Evaluation</b> – (siehe Phase II; die Demonstratoren werden nun in den Vergleich mit einbezogen)</li> </ul>
<b>Material</b>	<ul style="list-style-type: none"> <li>• Demonstratoren</li> <li>• Aufgabenkatalog</li> <li>• Bewertungsfragebogen</li> </ul>

**Phase IV: Am Ende des Projektes**

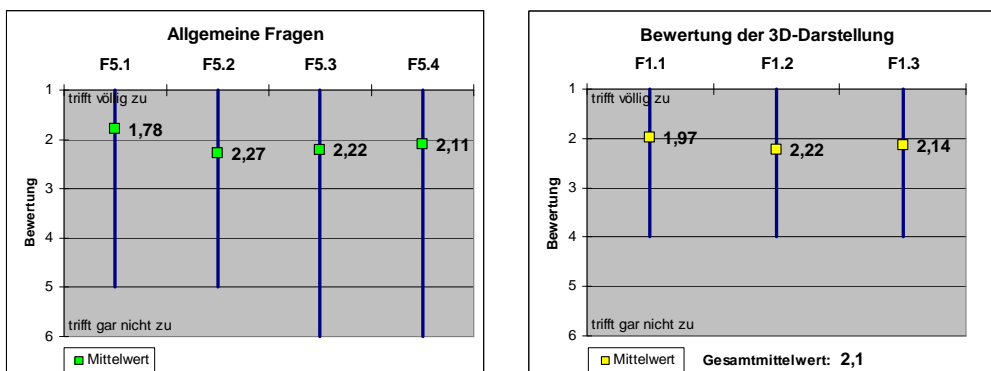
Ziel	<b>Summative Evaluation</b> <ul style="list-style-type: none"> <li>Überprüfung der Gesamtfunktionalität</li> <li>Evaluierung in einer echten Dokumentenumgebung</li> </ul>
Methoden	<ul style="list-style-type: none"> <li><b>Final User Tests</b> – Alle Funktionen werden von Testpersonen anhand eines Testkataloges getestet. Fehler werden protokolliert und zeitnah bereinigt; Noch vorhandene Funktionalitätslücken und Weiterentwicklungsideen/-wünsche werden festgehalten.</li> </ul>
Material	<ul style="list-style-type: none"> <li>Testkatalog</li> <li>Protokoll der Reflexionsergebnisse</li> </ul>

**7.1 Evaluierung Phase I**

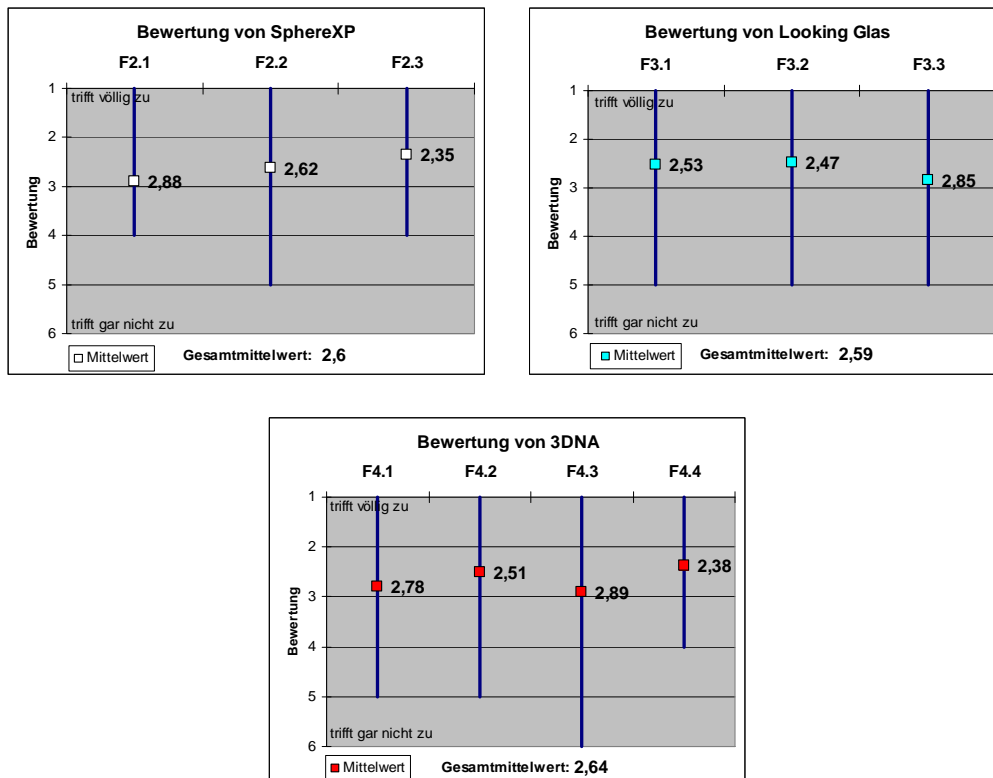
In Phase 1 wurden 46 Personen befragt, ca. 30% waren weiblich. Die Personen hatten unterschiedliche Vorkenntnisse und Berufe (Schüler, Studenten, Lehrer, Sekretärinnen, Wissenschaftler), so dass die gestellten Anforderungen an eine Rechnerumgebung als „stark unterschiedlich“ erwartet wurden. Neben allgemeinen Fragen zum Umgang mit Dokumenten wurden insbesondere die dreidimensionale Darstellung auf 3D-Displays sowie existierende Desktops den Probanden als Evaluierungsgrundlage an die Hand gegeben.

Im Gegensatz zu den Erwartungen waren keine signifikanten Unterschiede zwischen den Benutzergruppen erkennbar. Die dreidimensionale Darstellung schneidet in der Evaluierung insgesamt gut ab, ablehnende Kommentare lassen sich eher auf den aktuellen technischen Stand der Display-Hardware als auf die 3D-Visualisierung zurückführen. Der Wunsch nach besserer Unterstützung bei der Arbeit ist unverkennbar groß, es lassen sich diesbezüglich auch hier keine großen Meinungsverschiedenheiten feststellen. Bei den existierenden Desktops mit Pseudo-3D-Darstellung wird *Looking Glas* etwas schlechter bewertet als *SphereXP*. Beim dritten Kandidaten *3DNA* scheiden sich die Geister – insbesondere jüngeren Testteilnehmern gefällt dieses System durch die Anlehnung an 3D-Spiele gut.

Eine Übersicht über die Auswertungsergebnisse geben die folgenden Diagramme:







## 7.2 Evaluierung Phase II

Die Phase II diente zur nochmaligen intensiven Überprüfung des Konzeptes und der aktuellen Entwicklungsarbeit. Diese Phase ist der formativen Evaluierung zuzuordnen, d.h. der fortlaufenden, projektbegleitenden Bewertung der Ergebnisse.

Zur Demonstration und Evaluierung wurden zwei erste Prototypen für den „Virtual Workdesk“ sowie die abstrakte Suche und Visualisierung von Dokumentenräumen „DokuWorld“ umgesetzt. Die Teilnehmer erhielten hierzu eine geführte Tour durch die Applikation, mit besonderem Augenmerk auf den Vergleich von 2D- und 3D-Darstellung. Bei der Auswahl der Probanden wurde darauf geachtet, dass diese keine speziellen Kenntnisse im Bereich der Informatik haben (z.B. Sekretärinnen, Student(inn)en außerhalb der Informatik, usw. Insgesamt haben an der Phase II 14 Personen teilgenommen, hiervon waren ca. 60% weiblich. Die Auswertung der Fragebögen führte zu den folgenden zentralen Ergebnissen:

- Die neuartigen Visualisierungs- und Interaktionsmetaphern wurden von 67% als gut bewertet
- Die Klarheit und Ausdruckskraft der räumlichen Darstellung der Dokumente wurde von 80% herausgehoben
- 75% bestätigten, dass die Suche in @VISOR sehr intuitive ist
- 80% fanden das Arbeiten mit Dokumenten in @VISOR als intuitiv

- 60% kritisierten das Design als zu verspielt und zu bunt
- Bei den Befragungen zeigte es sich deutlich, dass der momentane Stand der Technik im Bereich der Hardware unbefriedigend ist: das 3D-Display wurde von 57% akzeptiert, der Datenhandschuh nur von 54%. Beim Display ist insbesondere der kleine 3D-Bewegungsbereich (Sweep Spot) und die im Vergleich zu 2D-Geräten geringer Auflösung von Nachteil. Beim Handschuh ist die Bequemlichkeit stark eingeschränkt.

### 7.3 Evaluierung Phase II

Wie schon die Phase II zählt auch die Phase III zur Kategorie der formativen Evaluierungen. Im Gegensatz zur Phase II wurde hier jedoch ein völlig anderer Evaluierungsansatz gewählt, der auf dem ersten vom Projektteam organisierten und durchgeführten HCIV-Workshop von den Experten vorgeschlagen wurde.

Als Benutzertyp wurden die so genannten „Geeks“ (bzw. „early adopters“) ausgewählt, d.h. Personen, welche bereits frühzeitig und ohne äußere Zwänge neue Produkte einsetzen bzw. auch sich noch im Entwicklungsstadium befindliche Software verwenden und testen. Hierzu wurden 25 Studenten der Vorlesung Computergrafik am Fachbereich Informatik der Technischen Universität Kaiserslautern herangezogen. Diesen wurde dabei keinerlei Software demonstriert, sondern stattdessen die Ideen und Visionen anhand von Beschreibungen, Bildern und Videos präsentiert.

Die folgenden Punkte geben einen Auszug aus den gewonnenen Ergebnissen:

Die wichtigen Dokumente werden vorne (72%) und in der Mitte (40%) erwartet, 20% suchten die wichtigste Informationen auf der linken Seite des Bildschirms.

Die Frage nach einer intuitiven Geste für die Bewegung eines virtuellen Dokumentenstapels auf dem Bildschirm wurde von 65% der Befragten mit „Drag'n'Drop-Geste durch Greifen des entsprechenden Objektes“ beantwortet, was exakt der Umsetzung im @VISOR-System entspricht. 16% antworteten auf diese Frage, dass sie eine Steuerung mit der Maus bevorzugen würden.

Den Probanden wurde ohne weitere Erklärung ein kurzes Video von der in @VISOR implementierten Selektionsgeste gezeigt und nach der Bedeutung dieser Geste gefragt. 52% gaben hier die erwartete Antwort. 24% der Befragten vermuteten jedoch, dass es sich hierbei um die Geste zur Löschung von Objekten handelt. Hier ist die Vermutung, dass die gewisse Ähnlichkeit der Geste mit dem Abfeuern einer Pistole in Egoshootern für diese Interpretation verantwortlich ist. Insgesamt zeigte sich aber, dass die gewählte Geste durch diese Fehlinterpretationsmöglichkeit nicht intuitiv genug ist.

## 7.4 Evaluierung Phase IV

Die vierte Evaluierung stellt eine summative Evaluierung dar, d.h. die Probanden erhielten einen Gesamtdemonstrator mit realistischer Datengrundlage sowie konkrete Aufgaben, die sie zu lösen hatten. Zur Einarbeitung in den Gesamtdemonstrator wurde vom Projektteam ein Video erstellt, welches den Probanden die wesentlichen Funktionalitäten und Abläufe im @VISOR-System veranschaulichte. Die Testpersonen konnten die Erklärungen dann parallel am Live-System nachvollziehen. Der zeitliche Aufwand der Einarbeitungsphase betrug 20-30 Minuten, die Durchführung der Aufgaben bis zu 60 Minuten pro Person.

Grundlage dieser Evaluierungsphase bildete das folgende Storyskript, welches neben einer Einbettung in eine realistische Alltagsaufgabe die Beschreibung der einzelnen Tasks, welche die Probanden durchzuführen hatten, enthält:

*Reporter Bernd Meier soll einen Bericht über die Fußball-WM 2006 schreiben. Die wichtigsten Informationen sind für ihn zunächst wer Weltmeister wurde, welche Mannschaften das Finale bestritten und wie das Endspiel ausgegangen ist. Deshalb sucht er in den Dokumenten auf seinem Computer als erstes nach dem Begriff „Endspiel“ und ordnet die gefundenen Dokumente ein. Er ist aber mit den Ergebnissen der Suche noch nicht zufrieden und sucht deshalb direkt noch nach „Weltmeister“ und „Finale“. Daraufhin bildet er zwei weitere Gruppen von Dokumenten, nämlich alle Dokumente, die allen drei Suchkriterien entsprechen, und eine weitere, die nur die insgesamt besten Treffer der Suche enthält.*

- *Suchen Sie Dokumente zu den Begriffen „Endspiel“, „Weltmeister“ und „Finale“.*
- *Ordnen Sie die gefundenen Dokumente in drei Gruppen „enthält Endspiel“, „enthält alle drei“ und „beste Treffer“.*

*Dann macht sich Herr Meier daran, die „Endspiel“-Gruppe nach für ihn interessanten Dokumenten zu durchsuchen. Er arbeitet sich durch die Dokumente und ordnet sie neu ein in Dokumente über das Endspiel 2006 und eine weitere Gruppe, die Informationen über Endspiele in der WM-Geschichte enthält. Er denkt sich, dass diese Informationen vielleicht später noch nützlich sein könnten und markiert diese Dokumentgruppe, um später darauf zugreifen zu können.*

- *Gruppieren Sie die Dokumente in der Gruppe „Endspiel“ in Dokumente zum Endspiel 2006 und Dokumente zu älteren WM-Endspielen.*
- *Markieren oder merken Sie sich diese Gruppe.*

*Außerdem kommt ihm beim Betrachten der geschichtlichen Dokumente der Gedanke, dass ein Abschnitt über das „Wunder von Bern“ gut als ein Zusatz in seinen Bericht passen würde. Er lässt also seine bisherigen Funde zunächst im jetzigen Zustand und wendet sich einer neuen Aufgabe, der Suche nach dem Wunder von Bern zu. Auch in diesem Zusammenhang interessieren ihn zunächst die gleichen Begriffe „Endspiel“, „Weltmeister“ und „Finale“.*

diesmal jedoch speziell im Zusammenhang mit dem Wunder von Bern. Auch von diesen Dokumenten legt er sich eine eigene Gruppe an.

- *Beginnen Sie eine neue Suche nach den Begriffen „Endspiel“, „Weltmeister“ und „Finale“, beziehen Sie sich aber diesmal speziell auf das Endspiel in Bern.*
- *Legen Sie eine Gruppe mit den besten Treffern an.*

*Dabei fällt ihm auf, dass er nicht genau weiß, wie die Weltmeister-Mannschaft von Bern aussah. Also sucht er im Zusammenhang mit Bern nach „Mannschaft“ und durchsucht die gefundenen Dokumente nach Fotos der Spieler. Nach einiger Zeit wird er fündig und markiert die Namen der Spieler im Dokument, außerdem legt er sich ein Lesezeichen für die Seite im Dokument an, das er mit „Mannschaft von Bern“ betitelt. Als er die Fotos der Spieler betrachtet, fällt ihm auf, dass es ja auch einen Kinofilm über das Wunder von Bern gab. Er erweitert seine Suche um den Begriff „Filme“ und findet nach kurzer Zeit ein Dokument mit dem Titel „Sepp Herberger & die Finalisten“, das Fotos der Hauptdarsteller enthält. Auch hier setzt er sich ein Lesezeichen, diesmal mit dem Titel „Hauptdarsteller Wunder von Bern“.*

- *Suchen Sie in ihren Dokumenten nach dem Begriff „Mannschaft“ und versuchen Sie, ein Dokument mit Bildern der Helden von Bern zu finden.*
- *Markieren Sie die Namen der Spieler im Dokument.*
- *Markieren oder merken Sie sich die betreffende Dokumentseite mit der Anmerkung „Mannschaft von Bern“.*
- *Suchen Sie zusätzlich nach „Filme“ und versuchen Sie, ein Dokument mit Bildern der Darsteller des Films „Das Wunder von Bern“ zu finden.*
- *Markieren oder merken Sie sich die betreffende Dokumentseite mit der Anmerkung „Hauptdarsteller Wunder von Bern“.*

*In diesem Moment betritt sein Chefredakteur das Büro und teilt Herr Meier mit, dass er an einem Bericht über die Technik und Sicherheit von WM-Stadien schreibt und von Herrn Meier dringend passende Dokumente braucht. Herr Meier legt also seine momentane Aufgabe auf Eis und beginnt eine neue Recherche. Er sucht in seinen Dokumenten nach „Stadien“ und bekommt eine große Liste mit Dokumenten geliefert. Diese Liste arbeitet er durch und bewertet die Dokumente nach ihrer Ähnlichkeit zum Thema Aufbau und Technik von WM-Stadien. Danach legt er eine Gruppe mit den wichtigsten Dokumenten an, um diese seinem Chefredakteur zu übermitteln. Daraufhin kehrt er zu seiner ursprünglichen Aufgabe zurück.*

- *Beginnen Sie eine weitere Suche nach Dokumenten, die den Begriff „Stadien“ enthalten.*

- *Legen Sie eine Gruppe mit Dokumenten an, die sich mit Aufbau und Technik der WM-Stadien beschäftigen.*

*Vier Wochen später soll Herr Meier wegen des Erfolges seines Artikels über die WM 2006 einen weiteren Artikel über die Geschichte der WM schreiben. Er erinnert sich, dass er während seiner Recherchen eine Gruppe mit passenden Dokumenten angelegt hat, und auch die Bilder der Helden von Bern würde er gerne in den Bericht einbauen. Er macht sich also auf die Suche nach der Mannschaft von Bern und der Dokument-Gruppe, die er über die WM-Geschichte angelegt hat...*

- *Finden Sie die Gruppe mit Dokumenten wieder, die Sie anfangs zum Thema WM-Geschichte gefunden hatten.*
- *Finden Sie das Dokument wieder, das die Bilder der Helden von Bern enthält.*

Die Ergebnisse der 4. Evaluierungsphase zeigen, dass die gemachten Entwicklungen von den befragten Personen durchweg sehr gut beurteilt wurden. Die Ergebnisse lagen allesamt zwischen 1.3 und 2.7 auf eine Skala von 1 (sehr gut) bis 5 (schlecht). Die folgenden Punkte geben einen Überblick über eine Auswahl von Detailergebnissen:

- Einbeziehung der Notizen in die Suche: 1.3 [Standardvarianz: 0.48]
- Kontext-sensitive Suche: 1.55 [0.31]
- Berechnung und Visualisierung von Ähnlichkeiten: 1.45 [0.49]
- Verwendung des Stiftes zur Texteingabe: 2.7 [0.95]
- Beim notwendigen Kontextswitch zwischen 2D und 3D berichteten 80% der Teilnehmer über keinerlei Probleme.
- Eine Entscheidung, ob das Durchblättern eines Dokumentes in 2D oder 3D besser funktioniert, konnte nicht gefällt werden. Es zeigte sich, dass gerade dieser Punkt extrem benutzerabhängig ist.
- Subjektive Aussagen der Probanden waren beispielsweise: "...the 3D arrangement is cool!" und "...that's nice, I did not expect it to work that well!"

Wie schon in den vorangegangenen Evaluierungsphasen war der momentane Entwicklungsstand der Hardware der allgemein gesehen größte Kritikpunkt. Hier ist jedoch anzunehmen, dass die Industrie hier rasche Fortschritte machen wird. So existiert beispielsweise bei der Firma SeeReal bereits ein 3D-Display-Prototyp, welcher 2D und 3D parallel und ohne Auflösungsverluste anzeigen kann.

## 8. Publikationen

- [1] Achim Ebert. *Context-sensitive Visualization*. PhD dissertation, University of Kaiserslautern, 2004.
- [2] Achim Ebert, Jochen Ehret, Lars Schuchardt, Heidrun Steinmetz, and Hans Hagen. *Scalable Information Visualization on Mobile Devices for Efficient Information Management*. IASTED International Conference on Visualization, Imaging, and Image Processing (VIIP), 2004.
- [3] Jochen Ehret, Achim Ebert, Lars Schuchardt, Heidrun Steinmetz, and Hans Hagen. *Context-adaptive Mobile Visualization and Information Management*. IEEE Visualization, 2004.
- [4] Bertin Klein, Stefan Agne, Achim Ebert, and Michael Bender. *Enabling Flow — A Paradigm for Document-centered Personal Information Spaces*. IASTED International Conference on Artificial Intelligence and Soft Computing (ASC), 2004.
- [5] Gerd Reis, Dennis Wagner. *Hardware-beschleunigter, interaktiver, zeitdynamischer Volumenrenderer*. Virtuelle und Erweiterte Realität, GI-Fachgruppe VR/AR, 2004.
- [6] Gerd Reis, Dennis Wagner. *Hardware Accelerated, Interactive, Time Dynamic Volume Rendering*. IASTED International Conference on Visualization, Imaging, and Image Processing (VIIP), 2004.
- [7] Achim Ebert, Matthias Deller, and Michael Bender. *An Immersive Gesture-controlled Interface for Virtual Document Information Spaces*. IASTED International Conference on Visualization, Imaging, and Image Processing (VIIP), 2005.
- [8] Matthias Deller, Michael Bender, Dennis Wagner, Achim Ebert, Stefan Agne and Bertin Klein. *Efficient Immersive Visualization of Virtual Document Spaces*. IASTED International Conference on Visualization, Imaging, and Image Processing (VIIP), 2005.
- [9] Achim Ebert, Michael Bender, Matthias Deller, Hans Hagen, Stefan Agne and Bertin Klein. *@VISOR - Adaptive Dynamic Visual Semantic Organizing*. Interactive Demo, IEEE Visualization 2005 conference, Minneapolis, USA, 2005.
- [10] Tristan Miller and Stefan Agne. *Attention-based information retrieval using eye tracker data*. In Proceedings of the Third International Conference on Knowledge Capture (K-CAP05), September 2005.
- [11] Katja Einfeld, Achim Ebert, Stefan Agne, and Bertin Klein. *DocuWorld --- A 3D User Interface to the Semantic Desktop*. 1st Workshop on The Semantic Desktop – Next Generation Personal Information Management and Collaboration Infrastructure, International Semantic Web Conference, 2005

- 
- [12] Markus Stäudel, Bertin Klein, and Stefan Agne. *Pen-based acquisition of real world annotations for document information spaces*. In: Proceedings of Semantic Desktop Workshop at the ISWC, Galway, Ireland, November 6, volume 175, 2005.
- [13] Stefan Klink, Koichi Kise, Andreas Dengel, Markus Junker, and Stefan Agne. *Document information retrieval*. In: Bidyut Baran Chaudhuri, editor, Digital Document Processing. Major Direction and Recent Advances, pages 1–27. Springer-Verlag London Ltd, 2006. ISBN 1846285011.
- [14] Katja Einfeld, Stefan Agne, Matthias Deller, Achim Ebert, Bertin Klein, and Christian Reuschling. *Dynamic visualization and navigation of semantic virtual environments*. In: Proceedings of the 10th International Conference on Information Visualization (IV 2006), pages 569–574, London, United Kingdom, 5-7 July 2006. IEEE Computer Society.
- [15] Torsten Bierz: *Interaction Technologies for Large Displays - An Overview*. In: Visualization of Large and Unstructured Data Sets, GI-Edition Lecture Notes in Informatics (LNI), Vol. S-4, 2006.
- [16] Stefan Agne, Christian Reuschling, and Andreas Dengel. *DynaQ — Dynamic Queries for Electronic Document Management*. In: Proceedings of the First International EDM Workshop (EDM 2006) – The Electronic Document Management in an Enterprise Computing Environment, pages 56–59, Hong Kong, China, 17 October 2006.
- [17] Matthias Deller, Achim Ebert, Michael Bender, and Hans Hagen. *Flexible Gesture Recognition for Immersive Virtual Environments*. In: Proceedings of the 10th International Conference on Information Visualization (IV 2006), pages 569–574, London, United Kingdom, 5-7 July 2006. IEEE Computer Society.
- [18] Andreas Dengel, Stefan Agne, Bertin Klein, Achim Ebert, and Matthias Deller. *Human-centered interaction with documents*. In: Proceedings of the First International Workshop on Human-Centered Multimedia, ACM Multimedia 2006, pages 1–8, Santa Barbara, USA, 2006.
- [19] Andreas Kerren, Achim Ebert, and Jörg Meyer (eds.): *Human-centered Visualization Environments*. Volume 4417 of LNCS Tutorial, Springer, 2007.
- [20] Sebastian Baumgärtner, Achim Ebert, and Matthias Deller. *Dimensional Congruence for Interactive Visual Data Mining and Knowledge Discovery*. Eurographics/IEEE VGTC, Symposium on Visualization (EuroVis), Norrköping, Sweden, 2007.
- [21] Sebastian Baumgärtner, Achim Ebert, Matthias Deller, and Stefan Agne. *2D Meets 3D: A Human-Centered Interface for Visual Data Exploration*. In: Extended Abstracts Proceedings of the 2007 Conference on Human Factors in Computing Systems, CHI 2007, pages 2273-2278, San Jose, California, USA, 2007
-