Räumliche Gestenerkennung und Natural User Interfaces mit Microsoft Kinect

Björn Oltmanns
Fachhochschule Kaiserslautern
mail@bjoernoltmanns.de

Denis Kruschinski
Fachhochschule Kaiserslautern
kontakt@deniskruschinski.de

Dieter Wallach
Fachhochschule Kaiserslautern
ERGOSIGN GmbH
dieter.wallach@fh-kl.de

Abstract
Der vorliegende Beitrag stellt Ansätze zur Realisierung von gestenbasierten User-Interfaces und Anwendungen auf Basis des 3D Sensors Microsoft Kinect in Verbindung mit Frameworks wie OpenNI und NITE vor. Nach einer kurzen Einführung in die genutzte Hard- und Software wird eine prototypische Applikation und die dieser zu Grunde liegenden Interaktionskonzepte skizziert. Hierbei kommen neben ein- und mehrhändigen Gesten insbesondere auch Konzepte wie virtuelle Cursor und Echtzeit Motion Capturing zur Analyse der Körperhaltung — lokal als auch im räumlichen Kontext — zum Einsatz. Im dritten Teil werden die vorgestellten Interaktionskonzepte schließlich diskutiert und auf Herausforderungen und Probleme — wie die Unterscheidung zwischen intentionalen und nicht-intentionalen Gesten — eingegangen.

Keywords:
/// Kinect, Natural User Interfaces
/// Natural Interaction
/// Räumliche Gesten

1. Einleitung

Im November 2010 stellte Microsoft den Kinect Sensor als Zubehör für die Spielekonsole Xbox 360 vor. Innerhalb kürzester Zeit wurde mit der Veröffentlichung von Frameworks wie OpenNI und NITE die Möglichkeit geschaffen, Kinect als Low-Budget-Grundlage zur plattformunabhängigen Entwicklung von Natural User Interfaces (NUI) einzusetzen. Mit dem Begriff NUI werden hierbei Interaktionsansätze angesprochen, die von Benutzern als natürliche Erweiterungen der eigenen Körperlichkeit empfunden werden, während die eigentliche Schnittstelle zu Applikationen weitgehend in den Hintergrund tritt. NUIs erlauben durch ihre intuitive Bedienbarkeit und schnelle Erlernbarkeit einen raschen Übergang vom Anfänger zum Experten. Unter „natural" ist hierbei insbesondere ein „natürliches Gefühl" bei der Verwendung solcher „Natural User Interfaces" zu verstehen.

2. Hardware und Frameworks

Um natürliche Interaktionskonzepte auf der Basis räumlicher Gesten zu ermöglichen, müssen Benutzer im Raum präzise erfassbar sein. Der Kinect Sensor ist hierzu mit einer Farbkamera, einem Infrarotlaser, einer Infrarotkamera sowie einem Mikrofonarray ausgerüstet und gestattet eine dreidimensionale Rekonstruktion des jeweiligen Raums. Hierzu projiziert der Kinect-Laser ein für menschliche Benutzer nicht sichtbares Muster in die Szene, vorhandene Objekte verzerren dieses Muster und werden von der Infrarotkamera aufgezeichnet um schließlich zu einer Tiefendarstellung transformiert zu werden. Diese Tiefendarstellung wird dann durch geeignete Algorithmen analysiert. Die im Rahmen der vorliegenden Arbeit eingesetzten Frameworks OpenNI und NITE bieten als Grundlage dieser Analyse folgende Optionen:

- Zugriff auf Rohdaten von Farb- und Infrarotkamera sowie auf das Tiefenbild;
- Erfassung von Benutzern und deren räumlicher Lokalisierung;
- Rekonstruktion einer elementaren Skelettstruktur von Benutzern und Analyse von deren Bewegungen;
- Erfassung der Hände von Benutzern als Grundlage der Gestenidentifikation;
- Eine begrenzte Anzahl an vordefinierten und anpassbaren räumlichen Gesten.

3. Entwicklung von Natural User Interfaces

Auf Basis des Kinect Sensors und der genannten Frameworks wurde eine interaktive Anwendung entwickelt, die mehrere kleinere Apps zur Erkundung zentraler Konzepte der räumlich-gestischen Interaktion umfasst. Nach dem Start dieser Anwendung wird der Benutzer zunächst durch einen Kalibrierungsprozess geführt, bei dem die Adaptierung des Frameworks an die Statur des Benutzers und die Handerfassung erfolgt.

Nach erfolgreicher Kalibrierung steht Benutzern ein durch räumliche Gesten gesteuertes Menü zur Verfügung, welches die Apps KINOTE, SENSE TV und SNOWHITE miteinander verknüpft. Die

Auswahl bzw. das Starten einer App erfolgt durch eine sagittale Handbewegung (nach vorne/hinten) als Push-Geste im Raum. Das Verlassen einer App und die Rückkehr zum Hauptmenü erfolgt jeweils durch die Ausführung einer transversalen (rechts/links) Wave-Geste.

KINOTE verbindet eine Slideshow-Anwendung mit einem virtuellen Presenter der eine Interaktion mit ein- und beidhändigen Gesten ermöglicht, die jeweils durch rekonstruierte Skelettdaten des Benutzers unterstützt werden. Durch Swipe-Gesten können Folien vor und zurückgeblättert werde; eine dem 2D Touch verwandte, zweihändige Zoom-Geste gestattet die vergrößerte Darstellung von Folienbereichen; mit einer greifenden Bewegung der rechten Hand kann hierzu ein Ausschnitt selektiert werden. Durch eine Swipe-Geste nach unten wird eine virtuelle Folienübersicht zur Direktauswahl aktiviert, und über eine Verschiebung der Hand und Push-Gesten bedient. Präsentationen können mit KINOTE vollständig ohne externe Fernbedienung auf der alleinigen Grundlage räumlicher Gesten gesteuert werden.

SENSE TV ist ein adaptiver User-Sensing Video Player, dessen Darstellungsmodus jeweils die Präsenz und räumliche Nähe von Benutzern berücksichtigt. Die Interaktion mit SENSE TV erfolgt durch Veränderung der Benutzerposition im Raum und Einhandgesten, beispielsweise zum Zurückspulen und Pausieren eines Videos. Der Videoplayer reagiert dabei auf die Abwesenheit von Benutzern, indem er das Video beispielsweise bei deren Verlassen des Raumes pausiert und die Wiedergabe bei Wiederkehr fortsetzt. Nähert sich der Benutzer dem Sensor bzw. Bildschirm, werden zusätzliche Informationen zum Video eingeblendet.

Die App SNOWHITE ist ein Augmented Reality-Spiegel zur interaktiven Kleideranprobe. SNOWHITE integriert in Echtzeit Realbild- und Skelettdaten. Benutzer können Kleidungsstücke aus verschiedenen Kategorien durch „Berührung" virtueller Kontrollelemente auswählen, die mit Hilfe von Skelettdaten an den Benutzer angepasst und superpositioniert über dem Real-Bild dargestellt. Hierbei ist auch die Aufnahme von Fotos dieser „virtuellen Modeschau" möglich. Die Selektion der virtuellen Kontrollelemente, die über den Bilddaten des Sensors dargestellt werden, erfolgt durch ein nachfolgend erläutertes Pointer Mapping auf der Basis von Skelettdaten der Hände des Benutzers. **[Abb. 1]**

4. Interaktionskonzepte mit Kinect

Im Folgenden werden räumlich-gestische Interaktionskonzepte beschrieben, welche sich mit Hilfe des Kinect Sensors und der zuvor angeführten Frameworks umsetzen lassen. Die Ansätze werden anhand der realisierten Lösungen zu KINOTE, SENSE TV und SNOWHITE verdeutlicht und um zusätzliche theoretische Betrachtungen ergänzt.

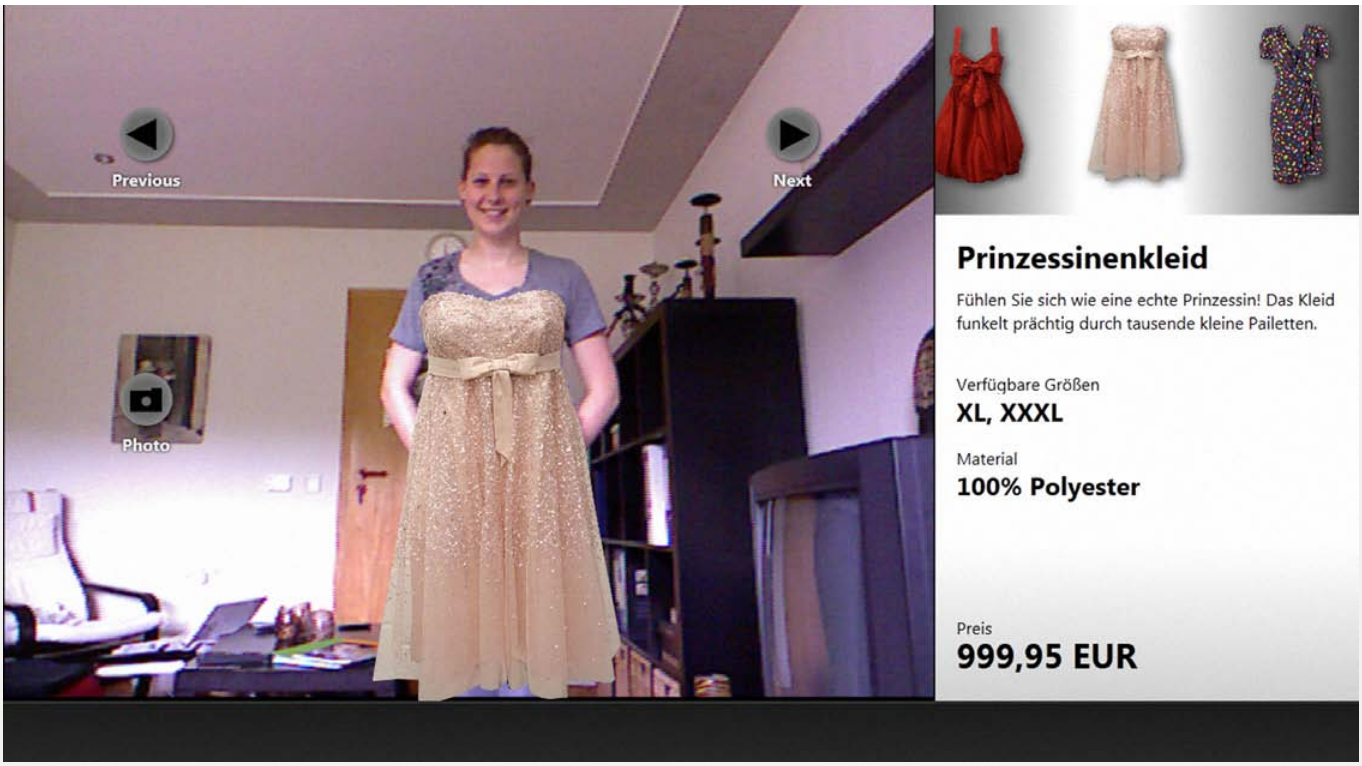

Abb. 1.
SNOWHITE, Augmented Reality
Kleideranprobe auf Skelettdatenbasis

4.1.
Realisierung räumlicher Gesten

Räumliche Gesten im eigentlichen Sinne stellen das wohl am stärksten mit Kinect assoziierte Interaktionskonzept dar. Bei räumlichen Gesten handelte es sich wohldefinierte Bewegungen von einer bzw. beiden Händen eines Benutzers. Die Frameworks OpenNI und NITE verfügen über eine Anzahl von vordefinierten Gesten:

- Die transversale Wave-Geste, eine Winkbewegung der Hand, wurde bei den oben skizzierten Anwendungen zum Verlassen der jeweiligen App als eine Art „Goodbye"-Geste eingesetzt.
- Swipes sind transversale (rechts/links) oder longitudinale (oben/unten) Wischgesten, ähnlich ihren Verwandten aus 2D Touch Systemen. Swipes eignen sich zum Wechseln zwischen einzelnen Sichten und wurde zum Vor- bzw. Zurückblättern von Folien bei KINOTE verwendet.
- Push, eine sagittale (nach vorne/hinten), drückende Vorwärtsbewegung der Hand wurde zum Selektieren von Elementen – etwa beim Auswählen von Menüelementen genutzt.
- Steady, ein ruhiges Halten der Hand.
- Die Circle Geste, einer Kreisbewegung der Hand

Neben den zuvor beschriebenen „offenen" Gesten, existieren im Framework NITE noch zwei „geschlossene" Gesten, die sogenannte false-positives für bestimmte Anwendungsfälle erheblich reduzieren können. Unter einem false-positive versteht man die nicht-intentionale Auslösung einer Geste im Sinne einer Fehlinterpretation einer Benutzerbewegung durch das System. Bei den „geschlossenen" Gesten handelt es sich um ein- und zweidimensionalen Slider mit jeweils N oder NxN Elementen. Solche Slider bieten sich insbesondere zur Realisierung von Menüs an und wurden beispielsweise für den oben angesprochenen Auswahlscreen (vgl. Abbildung 2) zum Starten einer App eingesetzt. Elemente des Startscreens werden durch Handbewegungen vorselektiert und in einen Hover-Zustand versetzt, anschließend können diese mit einer Push-Geste ausgewählt werden. **[Abb. 2]**

Gesten werden im NITE Framework durch eine proprietäre Mustererkennung realisiert. Sie sind grundsätzlich nur von einer als Primärpunkt registrierten Hand eines einzelnen Benutzers ausgeführt werden. In der beschriebenen Anwendung wurde während der Kalibrierung eine Wave-Geste benutzt, um den Primärpunkt auf der ausführenden Hand zu registrieren.

Zur Realisierung einer möglichst „natürlichen" Interaktion sollten räumliche Gesten zur Interaktion mit einem User Interface möglichst ikonisch im Sinne eines nachvollziehbaren Mappings auf reale physische Gesten eingesetzt werden. In diesem Sinne bilden etwa Swipe-Gesten eine Blättern in einem Katalog oder ein manuelles Wechsel von Folien angemessen ab.

4.1.1.
Intentionale vs. nicht-intentionale Gesten

Als Herausforderung bei der Realisierung der skizzierten Apps stellte sich die zuverlässige Diskriminierung zwischen intentionalen- und nicht-intentionalen Gesten heraus. Räumliche Gesten verfügen über keinen physischen „Schalter" der diese aktiviert oder deaktiviert. Bei touchbasierten Interfaces stellt der physische Kontakt mit einem virtuellen Control dieses Schalterelement dar – räumliche Gesten erweisen sich indes als in diesem Sinne immer aktiv(ierend).

Zur Erläuterung kann die robuste Identifikation einer Swipe-Geste dienen, die als gleichförmige transversale oder longitudinale Bewegung definiert ist. Ohne die Berücksichtigung einschränkender

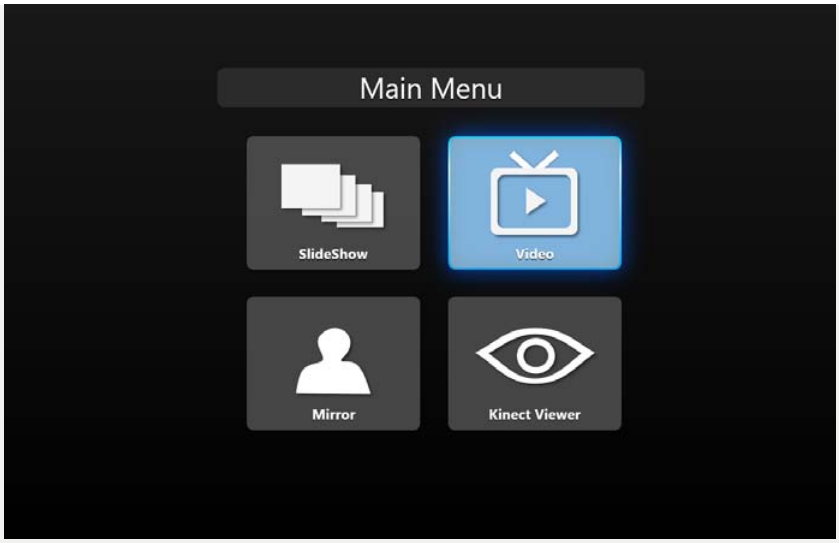

Abb. 2.
Gestengesteuerter 2x2 Slider des Hauptmenüs

Bedingungen ließe sich KINOTE kaum nutzen ohne fortlaufend unabsichtlich zwischen Folien hin und her zu springen. Was geschieht wenn eine Benutzerin nach einem Swipe nach links (zum Blättern auf die nächste Folie) ihre Hand nach rechts zurück in eine Ruhestellung bewegt? Wie kann die Bewegung in die Ruhestellung von einem Swipe nach rechts, der ein ungewolltes Rückblättern auslösen könnte, unterschieden werden? Sollte hierbei eine „Ruhezeit" zwischen zwei Gesten definiert werden?

Zur robusten Diskriminierung zwischen intentionalen und nicht-intentionalen Gesten können beispielsweise folgende Restriktionen eingeführt werden:
- Definition eines Zeitfensters innerhalb dessen die Bewegung der Swipe-Geste von Beginn zu deren Abschluss erfolgen muss.
- Definition einer maximalen Winkelabweichung aus der Bewegungsachse heraus.
- Definition einer Mindest- und Maximalgeschwindigkeit für die Durchführung einer Bewegung.

Bereits einfache räumliche Gesten bringen hierbei jedoch eine überraschende Komplexität mit sich, die einer allgemeinen Festlegung von statischen Parametern der Handbewegung entgegen stehen und den Einbezug weiterer Benutzerdaten wie dessen Körperhaltung oder auch die Einführung von virtuellen Schalterkonzepten zur zuverlässigen Gestenerkennung nahe legen. Neben der Feinabstimmung der Parameter elementarer Gesten wurden im Zuge der Entwicklungsarbeiten daher zusätzliche folgende Mechanismen analysiert:
- Definition von reservierten Gesten als Trigger: Die jeweilige Anwendung verbleibt in einem Ruhezustand bis zur Auslösung einer solchen Trigger-Geste, erst nach Aktivierung erfolgt eine Auswertung von Steuerungsgesten. Wird über einen definierten Zeitraum keine Geste identifiziert, oder der Benutzer als nicht aktiv oder gar abwesend erkannt (siehe Abschnitt User Sensing), geht das System in einen Ruhezustand zurück.
- Einbezug des räumlichen Kontextes: Gesten können nur in einem definierten virtuellen Raum vor dem Benutzer ausgeführt werden. Als Beispiel kann hier die Festlegung einer virtuellen Ebene in einer festgelegten räumlichen Ausdehnung vor dem Benutzer genannt werden. Diese Ebene muss zur Ausführung von Gesten mit den Händen durchstoßen werden. Zur Realisierung dieses Konzeptes ist ein alleiniges Tracking von Handkoordinaten nicht mehr hinreichend, vielmehr muss der Torso eines Benutzers zur Berechnung der Ebene erfasst werden.

Die darüber hinausgehende Kombination von räumlichen Gesten mit nicht-spatialen Hand- oder Fingergesten wurde im Rahmen der bisherigen Arbeiten zur Anwendungsentwicklung nicht betrachtet, da diese mit den genutzten Frameworks nicht oder nur mit sehr weitgehenden Erweiterungen realisierbar wären. So böte beispielsweise die von Wigdor und Wixton (2011) vorgeschlagene Pinch-Geste einen angemessenen Ausgangspunkt zur Abgrenzung von nicht-intentionalen Gesten. Dabei wird die durchzuführende Interpretation als Geste durch ein Zusammendrücken der Fingerspitzen während der Ausführung der selbigen jeweils explizit signalisiert. Ähnlich zeigt die Arbeit von Gawron, Głomb, Miszcza und Puchała (2011), wie Fingergesten algorithmisch mit Hilfe von Eigenvektoren aus den Daten eines Datenhandschuhen rekonstruiert werden können. Segers und Connan (2009) zeigt hohe Erkennungsraten für eine Auswertung von statischen Gesten auf Basis von 2D Bilddaten, jedoch unter Einschränkung auf eine Betrachtungsrichtung.

Grundsätzlich steigt die Wahrscheinlichkeit einer nichtintentionalen Auslösung mit der Anzahl zu einem Zeitpunkt prinzipiell verfügbaren Gesten (false positives), was eine Limitierung der zu betrachtenden Interaktionsgesten nahe legt. Die false-negatives lassen sich beispielsweise durch den Einsatz virtuellen Affordances und eines aussagekräftigen visuellen Feedbacks reduzieren. Bei der vorgestellten Applikation wurden daher Icons eingesetzt um entsprechende Interaktionsgesten zu signalisieren.

4.2. Pointer Mapping

Neben den oben angeführten komplexeren Algorithmen der spatialen Gestenerkennung lassen sich auch Konzepte realisieren, die auf dem direkten Einbezug der Handkoordinaten beruhen. So lassen sich bei dem sogenannten Pointer Mapping die Koordinaten einer oder beider Hände auf Bildschirmkoordinaten übertragen und z. B. als virtueller Cursor interpretieren. Diese Zuordnung sollte von lokalen Koordinaten eines dynamischen Handkoordinatensystems auf Bildschirm- oder UI-Koordinaten erfolgen. Hierzu werden Benutzer in einem virtuellen Koordinatensystem platziert, das sich mit ihnen durch den realen Raum bewegt.

Für das Auslösen von Aktionen mit Hilfe solcher virtueller Cursor sind verschiedene Ansätze denkbar:
- Eine verlässliche Methode ist das sogenannte Hovering mit sich „aufladenden" Controls. Der Benutzer hält hierbei den (virtuellen) Cursor für eine definierte Zeit über ein User Interface Control. Das UI Control lädt sich dann für die Dauer der Platzierung des Cursor über dem Element auf, bis schließlich eine Auslösung erfolgt. Die Aufladung sollte jeweils auf den Elementen visualisiert werden um Benutzern ein angemessenes Feedback zu geben.
- Eine weitere Möglichkeit zur Auslösung einer Aktion stellen räumliche Gesten dar. Als Beispiel kann eine Push-Geste angeführt werden, die durch eine Vorwärtsbewegung der Hand – als Durchbrechen der zweidimensionalen Ebene – repräsentiert wird.
- Ebenso sind multimodale Konzepte denkbar. Bei Verwendung des Kinect Sensors bieten sich hierzu Sprachkommandos an, die sich jeweils

auf Elemente unter dem virtuellen Cursor beziehen.

Die zuvor beschriebenen Konzepte machen es notwendig, Interface Controls für das Pointer Mapping relativ groß anzulegen um eine hohe Verlässlichkeit zu erreichen, da die Präzision von Handbewegungen im freien Raum stark eingeschränkt ist und sensorbedingte Toleranzen zu berücksichtigen sind.

4.3. Skeleton Tracking

Während das Pointer Mapping eine zweidimensionale Projektion der Hände darstellt, beschreibt das Skeleton Tracking eine dreidimensionale Interaktion unter Rückgriff auf den Körper des Benutzers. Mit dem Kinect Sensor und den Frameworks OpenNI und NITE ist in diesem Sinne eine Variante des Echtzeit Motion Capturing von Benutzern möglich. Die betrachteten Frameworks gestatten die (rudimentäre) Rekonstruktion der Skelettstruktur eines Benutzers in Form von 15 repräsentativen Punkten, den sogenannten Joints. Position und Orientierung von Kopf, Nacken, Torso Mittelpunkt, Schultern, Hüfte, Hände, Ellenbogen, Knie und Füßen stehen damit für weiterführende Auswertungen zur Verfügung. Voraussetzung des Skeleton Tracking ist jedoch eine Kalibrierung auf einen Benutzer. Hierdurch ist zum Beispiel die direkte Steuerung von virtuellen Charakteren realisierbar. Ebenso ist die Anpassung von virtuellen Elementen an die Statur des Benutzers möglich, wie dies im virtuellen Spiegel SNOWHITE realisiert wurde.

Mit dem Skeleton Tracking stehen sowohl weitergehende Möglichkeit zur Realisierung komplexerer Gesten, als auch Ansätze zur Unterscheidung zwischen intentionalen und nicht-intentionalen Gesten zur Verfügung. So können neben den Händen auch die Körperhaltung, Position und Orientierung des Benutzers im Raum analysiert werden. Im Rahmen der umgesetzten Apps wurde hierauf aufbauend eine komplexe zweihändige „Zoom"-Geste auf Skelettbasis realisiert. Hierzu werden beide Hände im freien Raum voneinander entfernt oder aufeinander zu bewegt. Das Schalterelement stellt hierbei die Stellung der Arme in Relation zum Körper des Benutzers dar. Wird eine Hand gesenkt und die führende Hand weiter vor dem Körper gehalten, kann von einem Zoom- in einen Panning-Modus gewechselt werden. Auf diese Weise erlaubt beispielsweise KINOTE die Verschiebung des Bildausschnittes einer Folie durch die Bewegungen der führenden Hand.

Abb. 3.
Einblendung von Zusatzinhalten in SENSE-TV

Werden die zuvor vorgestellten Konzepte des Skeleton Trackings und des Pointer Mappings mit Bilddaten der Realität kombiniert, so lassen sich weitere Interaktionskonzepte erschließen.

4.4. Augmented Reality

Durch die Kombination von realen Bilddaten des Sensors mit den rekonstruierten Benutzer- und Skelettdaten aus den Frameworks werden benutzerzentrierte Anwendungen der erweiterten Realität realisierbar. Im Gegensatz zum Pointer Mapping, wird hierbei in der Regel kein lokales Koordinatensystem verwendet, sondern eine zweidimensionale Projektion von Raum- auf Bildkoordinaten durchgeführt. Konzeptuell wurde dies in SNOWHITE durch eine Kombination des Pointer Mappings und Skeleton Trackings realisiert, bei welchem die Hände des Benutzers virtuelle Cursor in der zweidimensionalen Projektion darstellen.

Hierbei lassen sich virtuelle Controls im Bild anlegen, welche z. B. durch „Berührung" mit der Hand – ggf. kombiniert mit einer Push-Geste oder dem Prinzip der Aufladung – aktiviert werden. Für die Positionierung der Controls oder anderer interaktiver Elemente kann zwischen einer festen Positionierung im Bild oder einer an den Benutzer angepassten Positionierung unterschieden werden. Im virtuellen Spiegel SNOWHITE wurden die Controls zur Auswahl von Kleidungsstücken fest positioniert, entsprechend ist auch die Position des Benutzers zur deren Bedienung im Raum festgelegt.

4.5. User Sensing

Bei SENSE TV wurde die Identifikation anwesender Benutzer in das Interaktionskonzept einbezogen, sowie deren Position im Raum als Interaktionsvariante erschlossen. Für ein einfaches User Sensing ist im Vergleich zum Skeleton Tracking keine Kalibrierung notwendig. Bewegungszentrum und Benutzerumriss sind für sich in den Erfassungsbereich des Sensors bewegende Benutzer durch das Framework leicht erfassbar. SENSE TV reagiert auf die Abwesenheit von Benutzern durch Pausieren und blendet distanzabhängig Zusatzinformationen zu laufenden Filmen ein. **[Abb. 3]**

5. Ausblick

Mit dem Kinect Sensor steht eine äußerst kostengünstige technische Grundlage zur Realisierung von Natural User Interfaces auf der Basis räumlich-gestischer Interaktionskonzepte zur Verfügung. Die im Rahmen dieses Beitrags vorgestellten Apps und die darin explorierten Interaktionsmechanismen geben einen ersten Einblick in die Möglichkeiten und Herausforderungen von Natural User Interfaces auf Basis von Kinect und können als Ausgangspunkt zur Entwicklung robuster räumlicher Interaktionsgesten gesehen werden.

Literatur

1. Gawron, P., Głomb, P., Miszczak, J. P. & Puchała, Z. (2011). Eigengestures for natural human computer interface. arXiv:1105.1293v1
2. Segers, V. & Connan, J. (2009). Real-time gesture recognition using eigenvectors. Proc. Southern Africa Telecommunication Networks and Applications Conference (SATNAC 2009). 363-366. Swaziland
3. Widgor, D., Wixon, D. (2011). Brave NUI World: Designing Natural User Interfaces for Touch and Gesture. Burlington: Morgan Kaufmann