

NINGU

THE NINJA WAY OF PRESENTING

JULIA ALEXANDRA PANEK & ALEXANDER ZANDELOV

INTERACTION ENGINEERING || PROF. DR. MICHAEL KIPP

HOCHSCHULE AUGSBURG
UNIVERSITY OF APPLIED SCIENCES

WS 2015/2016

INHALTSVERZEICHNIS

1. PROJEKTBESCHREIBUNG
2. EINFÜHRUNG
3. KONZEPT
4. IMPLEMENTIERUNG
 - 4.1. PLANUNG
 - 4.2. IMPLEMENTIERUNG DER WERKZEUGE
 - 4.3. IMPLEMENTIERUNG DER SKELETTERKENNUNG
 - 4.4. IMPLEMENTIERUNG DER GESTENSTEUERUNG
5. EVALUATION
6. LESSONS LEARNED
7. AUSBLICK

ABBILDUNGSVERZEICHNIS

1. PROJEKTBECHREIBUNG

„NINGU - The Ninja way of presenting“ beschreibt eine alternative, gestengesteuerte Interaktionsmöglichkeit, mit deren Hilfe Präsentationssoftware (wie beispielsweise MS Powerpoint) ohne weitere Hardware oder die bedient werden kann. Hierbei werden mit einer Microsoft Kinect Kamera die Gesten des Referenten erkannt und in Abhängigkeit von seiner Entfernung zur Präsentationsfläche (z.B. einer Wand, auf der die Folien dargestellt werden) interpretiert. Eine Aktion (Folie vor/zurück) wird je nachdem, ob und welche Geste in einem spezifischen Bereich vollzogen wird, ausgeführt.

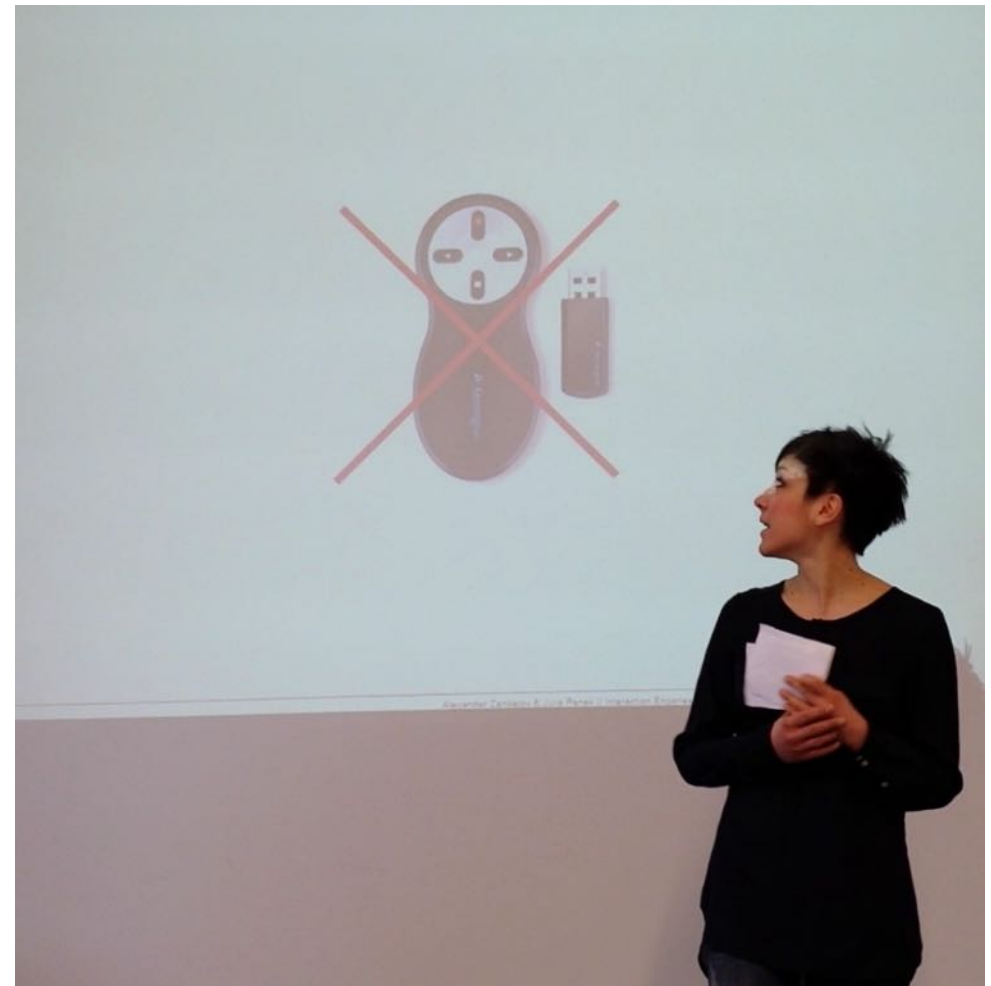


Abb.1

2. EINLEITUNG

Den massenhaften Zulauf einer von Steve Jobs präsentierten Keynote und die schon fast euphorischen Zwischenrufe während einer solchen galten sicherlich nicht nur der innovativen Hardware - das gezielte Spielen mit den Emotionen der Zuschauer, sein Wissen um die Bedeutung der Dramaturgie und nicht zuletzt seine sorgsam eingesetzte aber bis ins kleinste Detail durchdachten Körpersprache machten jede seiner Präsentation zu einem regelrechten Event.

Dass eine exaltierte Körpersprache auch zu viel des Guten sein kann, hat Job's Antipode Steve Ballmer 2011 eindrucksvoll bewiesen. Was für eine Figur wir auf einer Bühne, in einem Konferenz- oder im Seminarraum während eines Vortrags abgeben ist also von großer Wichtigkeit. Ob der Zuhörer mit uns ist, mit uns mitgeht, uns zuhört und sich von uns fesseln und emotional berühren lässt (und letztendlich treffen wir auch höchstrationale Entscheidungen immer auch auf Basis unserer Emotionen) hängt also zu einem gewissen Grad auch von unserer Darstellung ab.



Abb.2: Steve Ballmer

Gerade, wenn es darum geht das Publikum von einem selbst (z.B. während eines Pitches) oder von dem eigenen Produkt zu überzeugen, hängt unsere Glaubwürdigkeit von unserer Non-verbale Kommunikation, in dem Fall der Körpersprache, ab. Gestengesteuerte Präsentations-Kontrollmechanismen wurden in den vergangenen Jahren bereits mehrfach konzipiert und implementiert. Allen gemein ist jedoch die Tatsache, dass die Bewegungsabläufe sehr ausladend scheinen und dem Zuschauer jederzeit vor Augen führen, wer direkt für den Ablauf und demzufolge für die gesamte Performance verantwortlich ist. Eine alternative Gestensteuerung zu entwickeln, die nahezu unsichtbar vor dem Publikum erfolgt, sich also natürlichen Bewegungsabläufen bedient, war das angestrebte Ziel der Autoren.



Abb.3

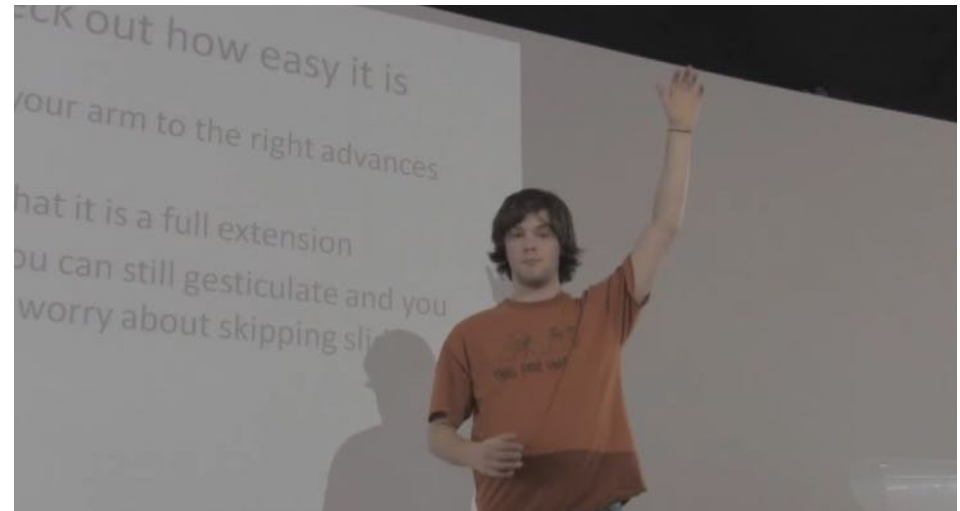


Abb.4

2. KONZEPT

Da sich eine folienbasierte Präsentation auf die Kernfunktionen Folie-vor und Folie-zurück beschränkt, wurde in der Prototypphase auf diese beiden Funktionalitäten der Fokus gesetzt. Dazu bedienten sich die Autoren des Prinzips der Aktivitätszonen und definierten 3 Bereiche zwischen der Kinect-Kamera und der Präsentationsfläche, innerhalb derer sich unterschiedliche, subtil vorführbare Gesten ausführen lassen.

In der ersten Zone, die sich unmittelbar an bis etwa 80cm vor der Präsentationsfläche befindet (Abb. 5) wurden zwei Entry-Zonen implementiert, in die der Präsentierende durch das bloße Betreten eine Funktion (rechts für Folie-vor und links für Folie-zurück) auslöst. Zudem erfolgt eine Auslösung durch die Vollführung der in Abb. 6 und 6a skizzierten Geste.

Zone 2 und 3 weisen keine Entry-Zonen auf. (Abb. 7 und 8, 8a) Bevor mit der Realisierung des Projektes begonnen werden konnte, mussten sich die Autoren für einen geeigneten Entwicklungsprozess entscheiden.

Dieser definiert die Vorgehensweise, nach der die Umsetzung erfolgen sollte. Für das Abschlussprojekt erschien ein agiler Entwicklungsprozess als sinnvollste Alternative. Hierbei diente das Vorgehensmodell Scrum als Vorbild. Das iterative Durchlaufen der Projektphasen sowie stetige Rücksprachen miteinander sind einige der Merkmale, die den agilen Entwicklungsprozess auszeichnen.

Durch die relativ kurzen Iterationszyklen wurde eine flexible Umsetzung der Anforderungen ermöglicht, sodass neue Erkenntnisse schnell in das Projekt einfließen konnten.

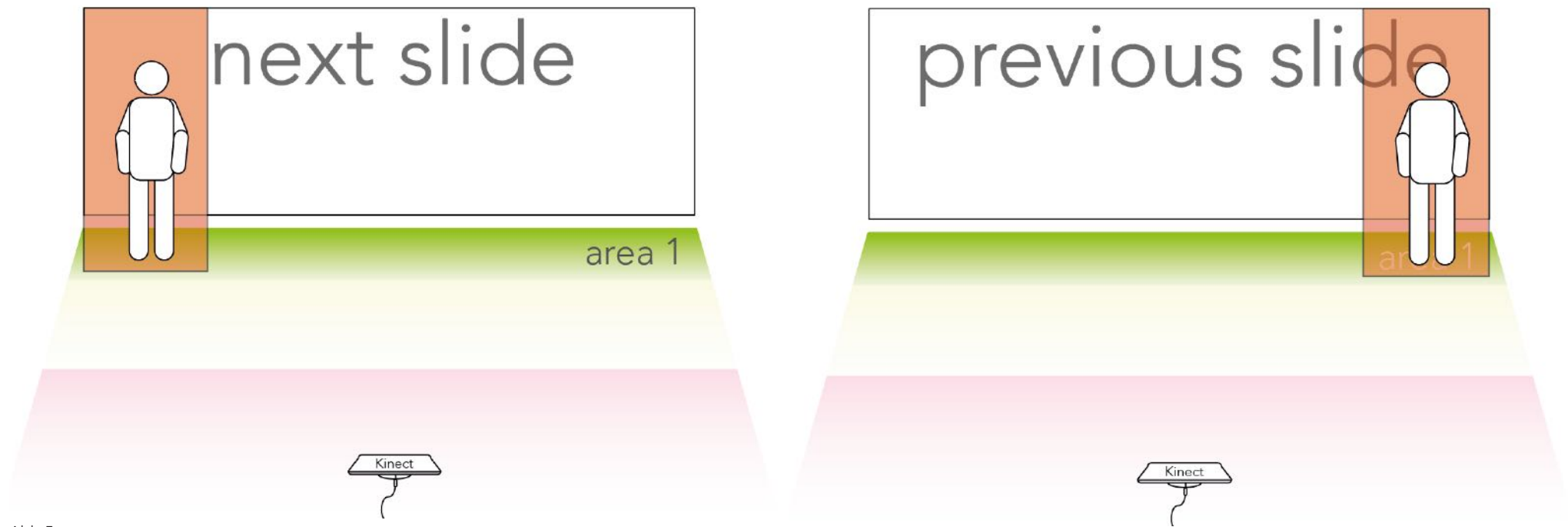


Abb.5

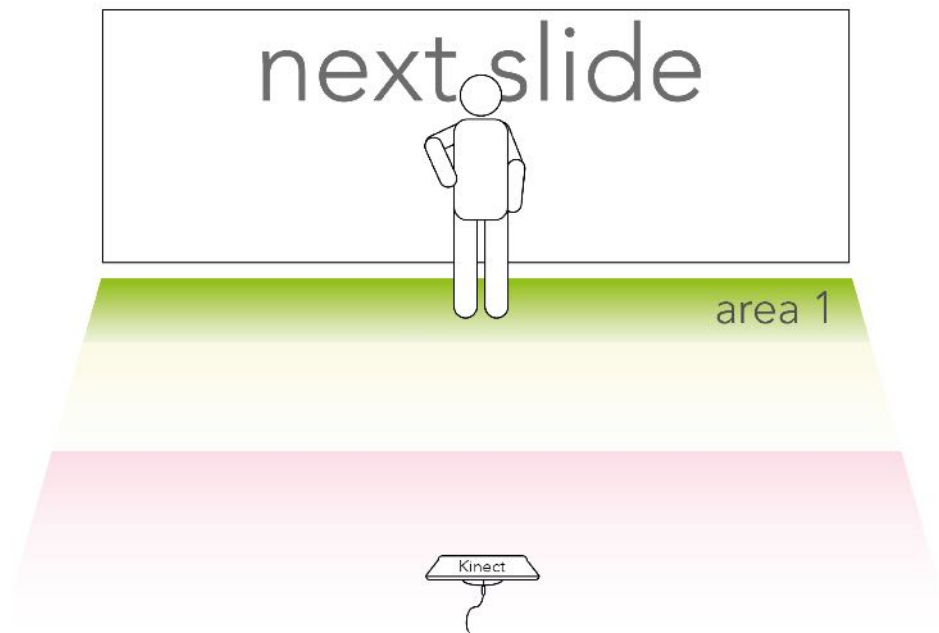


Abb. 6
Rechte Hand in die Hüfte stemmen löst einen Tastendruck der rechten Pfeiltaste aus

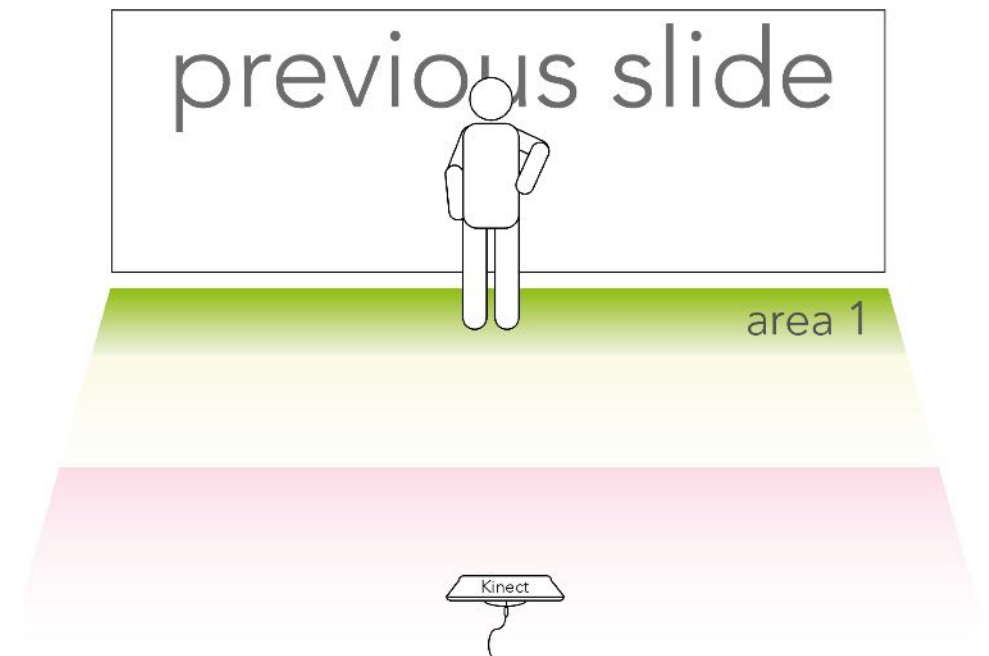
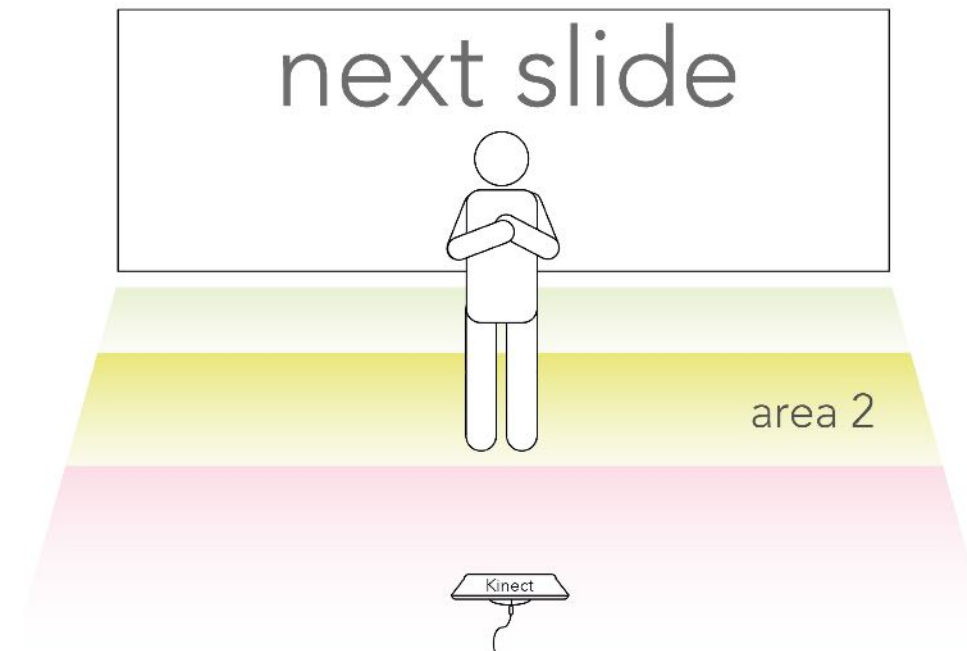


Abb. 6a
Linke Hand in die Hüfte stemmen löst einen Tastendruck der linken Pfeiltaste aus



9

Abb. 7

Beide Hände vor dem Brustkorb gekriegt löst einen Tastendruck der rechten Pfeiltaste aus

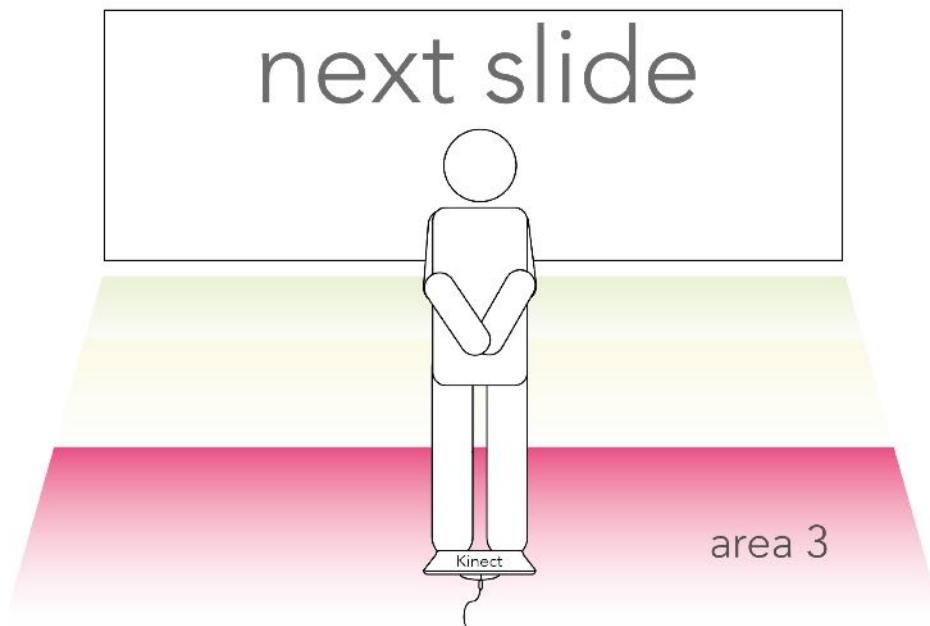


Abb. 8
Hände vor dem Schoß gekreuzt löst einen Tastendruck der rechten Pfeiltaste aus

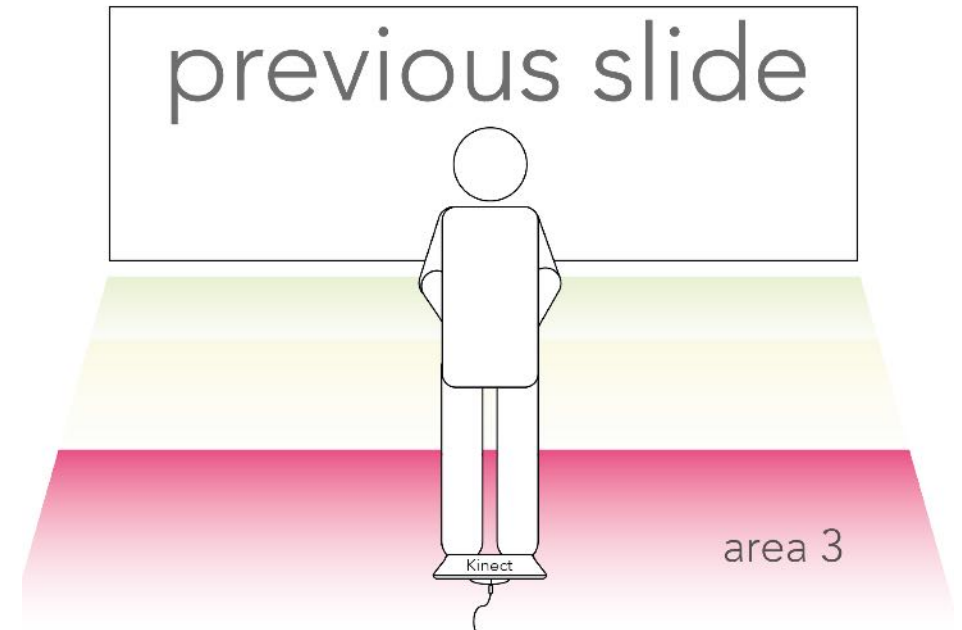


Abb. 8a
Hände hinter dem Rücken gekreuzt löst einen Tastendruck der linken Pfeiltaste aus

4. IMPLEMENTIERUNG

4.1. PLANUNG

Bevor mit der eigentlichen Implementierungsphase begonnen wurde, wurde zunächst ein kleiner Iterationsplan erstellt. In ihm wurden die einzelnen Iterationsschritte und deren Reihenfolge definiert. Innerhalb einer Iteration wurde jeweils eine bestimmte Funktionalität implementiert, die dann am Ende der entsprechenden Iteration besprochen und getestet wurde.

Für die Entwicklung der Software wurde die Entwicklungsumgebung Visual Studio 2015 verwendet. Das Projekt wurde mit der Programmiersprache „WPF“ (Windows Presentation Foundation) realisiert, wobei die Präsentation der Oberfläche mit der Auszeichnungssprache „XAML“ (Extensible Application Markup Language) und die Logik mit C# programmiert wurde.

Um die Arbeitsschritte zu Planen und übersichtlich zu halten wurde das Online-Iterationstool „Trello“ verwendet.

4. IMPLEMENTIERUNG

4.2. IMPLEMENTIERUNG DER WERKZEUGE

Nachdem die benötigten Werkzeuge eingebunden waren, konnten die jeweiligen Kameras der Kinect angesprochen werden. Um die Kameras testweise verwenden zu können, wurde eine Fensteroberfläche angelegt, mit der maximal möglichen Auflösung von 640x480 Pixel. Die Aktivierung der Kameras erfolgt unmittelbar beim Lade-Event des Fensters. Die entsprechenden Events für die für die Weiterverwendung der Kameras werden ebenfalls geschaffen (siehe Abb. 8).

Um nun das Videobild der Kamera live in dem Fenster sehen zu können, wurde beim Fenster ein Image-Element hinzugefügt. Dieses wird beim Event der Farbkamera, dem „sensor_ColorFrameReady“-Event, mit einem konstanten Pixelsatz mit einer bestimmten Auflösung, dem sogenannten „Bitmap Stream“, befüllt.

```
sensor.Start();  
//Color camera enable  
sensor.ColorStream.Enable(ColorImageFormat.RgbResolution640x480Fps30);  
sensor.ColorFrameReady += new EventHandler<ColorImageFrameReadyEventArgs>(sensor_ColorFrameReady);  
  
//Depth camera enable  
sensor.DepthStream.Enable(DepthImageFormat.Resolution320x240Fps30);  
  
//Skeleton camera enable  
sensor.SkeletonStream.Enable();  
sensor.SkeletonFrameReady += new EventHandler<SkeletonFrameReadyEventArgs>(sensor_SkeletonFrameReady);
```

13

Abb.9

4. IMPLEMENTIERUNG

4.3. IMPLEMENTIERUNG DER SKELETT-ERKENNUNG

Nachdem die Farbkamera ein funktionierendes Bild lieferte, wurde als Nächstes die Skeletterkennung bzw. die Erkennung der Gelenkpunkte („JoinTypes“) realisiert. Das im Punkt 4.2 Beschriebene Event für die Skelettkamera wurde bereits angelegt und konnte nun genutzt werden.

Die Kinect-Kamera besitzt die von Haus aus integrierte Skeletterkennungssoftware und muss im Event in einer von der „Microsoft.Kinect.Skeleton“-Bibliothek gelieferten Variable „Skeleton“ deklariert werden. Anschließend konnten die einzelnen JoinTypes des Skeletts separat angesprochen werden.

14

```
//Track the position
Skeleton closestSkeleton = skeletons.Where(s => s.TrackingState == SkeletonTrackingState.Tracked)
    .OrderBy(s => s.Position.Z * Math.Abs(s.Position.X))
    .FirstOrDefault();

if (closestSkeleton == null)
    return;

//If the camera finally recognizes you

//Define the JointTypes of your body
var head = closestSkeleton.Joints[JointType.Head];
```

Abb. 10

15

4. IMPLEMENTIERUNG

4.3. IMPLEMENTIERUNG DER GESTENSTEUERUNG

Nachdem die Skelettkamera die einzelnen JointTypes korrekt verfolgte, konnten die Positionen dieser JointTypes für Gestensteuerungen während der Laufzeit verwendet werden. Hierfür wurden drei Zonen, die sich durch den Abstand zur Kamera unterscheiden, definiert. In diesen Zonen wurden die Gesten (siehe Punkt 2 „Konzept“) angelegt. Die Erkennung einer Geste basiert auf den Positionen von ausgewählten JointTypes. Dabei wird zwischen der X-Achse, Y-Achse und Z-Achse unterschieden. Sollten sich alle erforderliche Positionen aller JointTypes im definierten Bereich befinden, löst das die dementsprechende Geste aus. Dabei wird ein fiktiver Tastatur-Knopfdruck erzeugt und an die gerade geöffnete und fokussierte Applikation gesendet.

Nach jeder ausgelösten Geste wird der jeweilige Thread, der für das Kamera-Erkenn-Event zuständig ist, fünf Sekunden lang pausiert. Dies verhindert das mehrmalige Auslösen von Gesten. Eine Pausier- Funktion bietet die „Microsoft.Kinect“ - Bibliothek leider nicht.

Während man sich in einer Area oder außerhalb davon befindet wird eine Information in der Symbolleiste sichtbar, welche ständig darüber informiert, in welcher Zone man sich gerade befindet. Ebenso wird beim Aktivieren einer Geste eine Symbolleistennachricht angezeigt. Dafür wurde die „System.Windows.Forms.NotifyIcon“ – Bibliothek verwendet. Diese ist in Windows Betriebssystemen für die Darstellung von Symbolen und Symbolbenachrichtigungen zuständig.

Abschließend wurde eine globale Variable definiert um das fehlerhafte Auslösen von Gesten beim Start auszuschließen. Bei dem Start der Kamera befinden sich die JoinTypes in einer zufälligen Position und können so unerwartete Gesten auslösen.

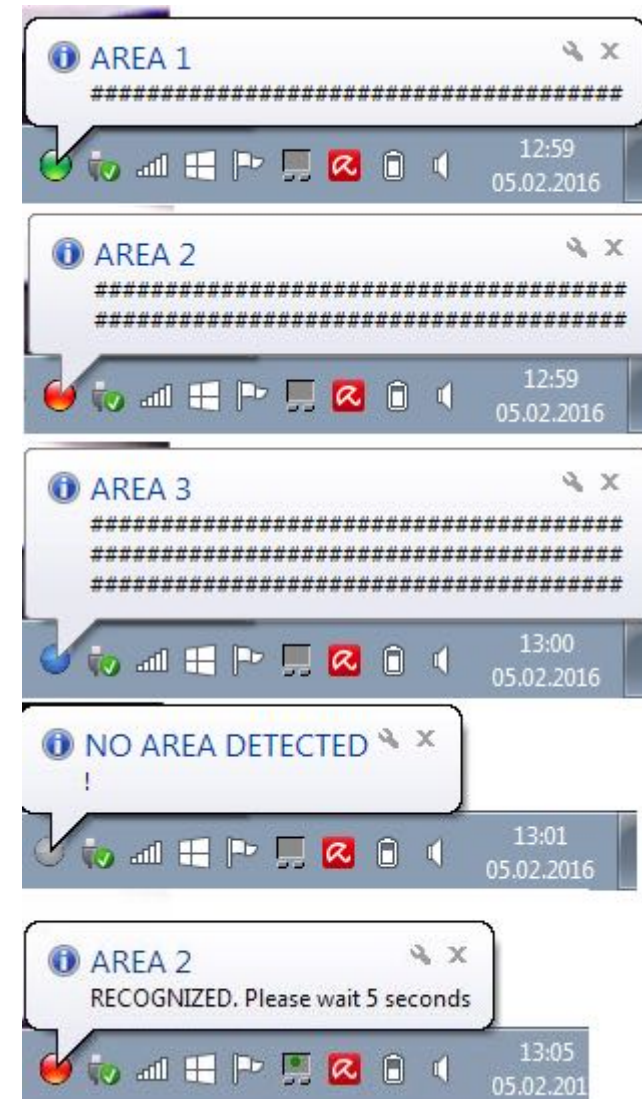
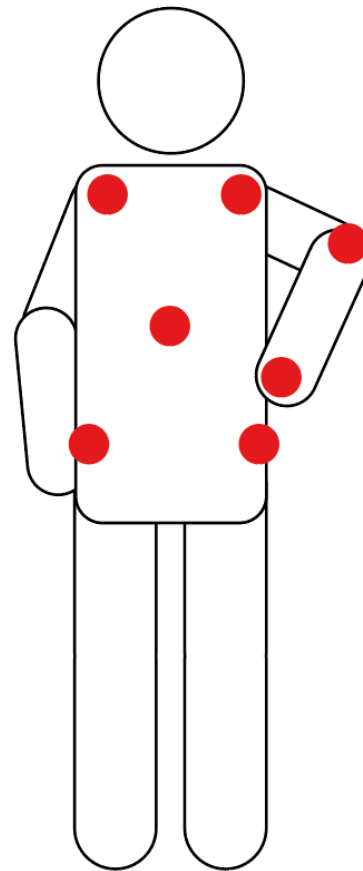


Abb. 11



area 1

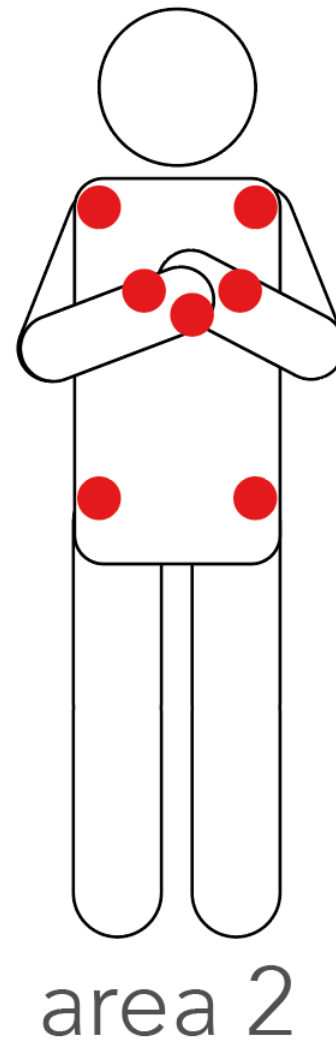
Rechtes Handgelenk.X \leq Rechte Schulter.X + 13cm &&
 Rechtes Handgelenk.X \geq Rechte Schulter.X - 13cm

Rechtes Handgelenk.Z \geq Rechte Hüfte.Z - 8cm &&
 Rechtes Handgelenk.Z \leq Rechte Hüfte.Z + 8cm

Rechtes Handgelenk.Z \geq Rechte Schulter.Z - 10 cm &&
 Rechtes Handgelenk.Z \leq Rechte Schulter.Z + 10cm

Rechtes Handgelenk muss über der Rechten Hüfte sein
 und unter der Spinebase
 Rechter Ellenbogen muss über der Spinebase sein und
 unter der Spinebase Y + 20cm sein.

Abb.12



Linkes Handgelenk.X \geq Linke Schulter.X + 8cm

Linkes Handgelenk.Y \geq Linke Hüfte

Linkes Handgelenk.Y \geq Spinebase.Y - 10 cm &&

Linkes Handgelenk.Y \leq Spinebase.x + 10cm

Linkes Handgelenk.Z \geq Linke Hüfte.Z - 50cm &&

Linkes Handgelenk.Z \leq Linke Hüfte.Z + 10cm

Rechtes Handgelenk.X \leq Rechte Schulter.X - 8cm

Rechtes Handgelenk.Y \geq Rechte Hüfte.Y

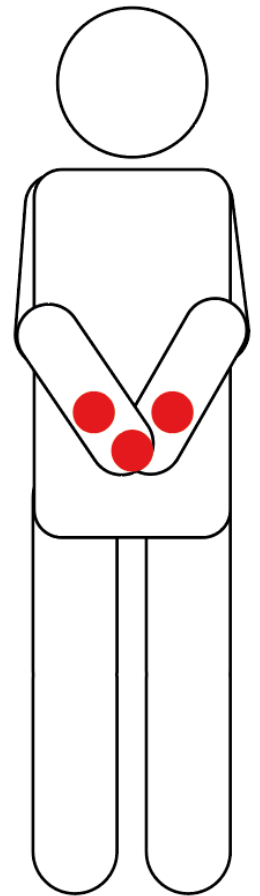
Rechtes Handgelenk.Y \geq Spinebase.Y - 10 cm &&

Rechtes Handgelenk.Y \leq Spinebase.x + 10cm

Rechtes Handgelenk.Z \geq Rechte Hüfte.Z - 50cm &&

Rechtes Handgelenk.Y \leq Rechte Hüfte.Z + 10cm

Abb.13



Rechtes Handgelenk.X \leq rechte Hüfte
Linkes Handgelenk.X \geq linke Hüfte

Rechte Hand < rechte Hüfte.Y+10cm
Linke Hand < linke Hüfte.Y+10cm

area 3

Abb.14

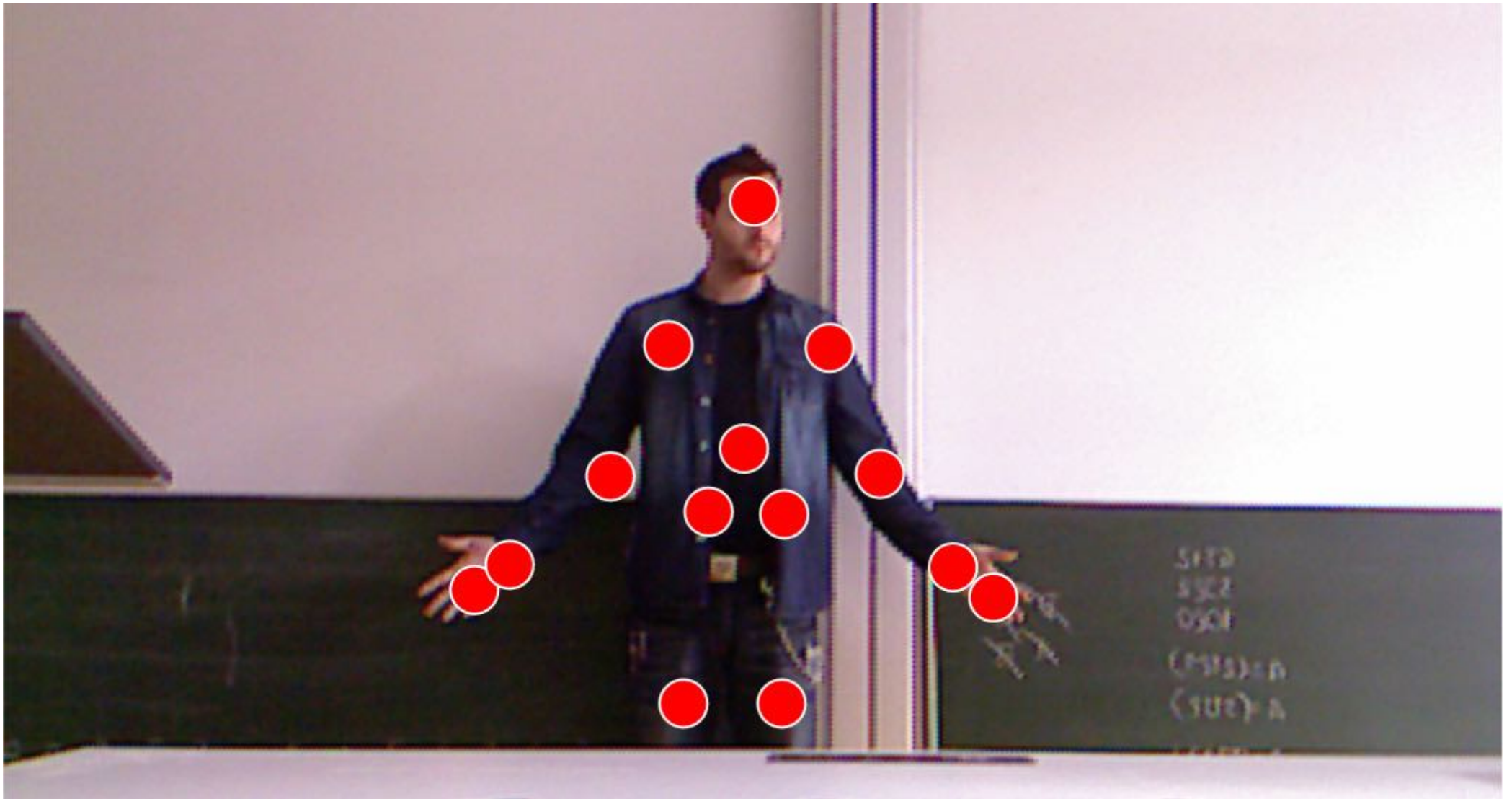


Abb. 15: JointTypes

5. EVALUATION

Im Rahmen des Projektes wurde eine Evaluation durchgeführt. Das Ziel der Evaluation war es, Feedback zu erhalten, um rauszufinden welche Aspekte unseres Designs verbessert werden müssen und welche Aspekte von den Benutzern anerkannt werden.

TEILNEHMER

Es haben 6 Teilnehmer (4 männliche und 2 weibliche), in einem Alter von 23 und 29, von der Fachhochschule und dem Freundeskreis teilgenommen. Alle Teilnehmer sind regelmäßige Computernutzer (4+ Stunden pro Tag). 3 Teilnehmer hatten etwas bis gar keine Erfahrung mit interaktiven Geräten (z.B. XBOX-Kamera oder MYO-Armband). Das Bildungsniveau lag bei allen im Hochschulbereich.

TESTUMGEBUNG

Als Testumgebung diente ein Vorlesungsraum der Fachhochschule. Ein Beamer war für die Darstellung einer Testpräsentation mit einer Auflösung von 640x480 Pixel im Raum vorhanden. Die XBOX 360-Kamera wurde mittig und 3 Meter vor der Präsentationsfläche platziert.

AUFGABENSTELLUNG

Die Aufgabenstellung beinhaltete 10 Fragen. Davon befassten sich 8 Fragen mit den einzelnen Gesten, jeweils eine Frage zur Erkennung des visuellen Feedbacks und zur allgemeinen Meinung zum System, verglichen zu gängigen Alternativen. Die Teilnehmer wurden vorher kurz über die einzelnen Gesten und Areas unterrichtet.

PROZEDUR

Die Testpersonen führten die Aufgaben alleine aus. Jede Person konnte den Fragebogen selbst durcharbeiten und die Fragen auch gleich beantworten bzw. Vorschläge und Anmerkungen vermerken. Die Reihenfolge musste dabei nach der Fragestellung eingehalten werden. Die Hauptfragen bezogen sich auf folgende Punkte:

Wie gut reagierte das System?

Diese Frage konnte mit 5 Möglichkeiten, „sehr gut“ bis „sehr schlecht“, beantwortet werden.

Wie intuitiv finden Sie die Gesten?

Diese Frage konnte mit 5 Möglichkeiten, „sehr intuitiv“ bis „überhaupt nicht intuitiv“, beantwortet werden.

Wie gut haben Sie anhand des visuellen Feedbacks (die Farbe und Text in der Symbolleiste) die Areas erkannt?

Diese Frage konnte mit 5 Möglichkeiten, „sehr gut“ bis „sehr schlecht“, beantwortet werden.

Wie finden Sie das System, wenn Sie es mit anderen gängigen Präsentationsmöglichkeiten vergleichen (Pointer, Tastatur etc.)

Diese Frage konnte mit 5 Möglichkeiten, „viel besser“ bis „viel schlechter“, beantwortet werden.

ERGEBNISSE

Im Großen und Ganzen wurde das System positiv bewertet. 4 der Teilnehmer fanden das System besser als alternative Präsentationsmethoden, jedoch unter der Voraussetzung, dass die Erkennung verbessert und die Gestensteuerung optimiert wird. Des Weiteren fand ein Teilnehmer die Präsentationsmethode gleich gut und einer fand es schlechter als die gängigen Alternativen.

Die Teilnehmer hatten bereits nach 5 Minuten die Bedienung verinnerlicht und hatten nur Probleme bei einem flüssigen Ablauf der Präsentation. Eine Anmerkung war, dass diese Methode viel Übung erfordert, da man sich doch die Positionen und Bewegungsabläufe stark verinnerlichen muss.

Die Gestenerkennung funktionierte größtenteils und wurde von den Testern anerkannt. Allerdings führte die gelegentliche Fehlinterpretation einer Geste zu Verunsicherungen bei den Testpersonen. Als mögliche Erweiterung wurde vorgeschlagen, individuelle Gesten zu ermöglichen, die dem natürlichen Bewegungsablauf des Nutzers entsprechen. Area 3 und die Seitenbereiche der Area 1 waren aufgrund ihrer Simplizität am beliebtesten. Area 2 war mitunter am anfälligsten für Fehlinterpretationen. Grund dafür war das unbewusste Gestikulieren beim Sprechen selber. Area 1 wies aufgrund unterschiedlicher Physiognomien und darausfolgenden unterschiedlichen Bewegungsabläufen die größten Probleme auf. Dies könnte leicht durch eine Kalibrierung oder Personalisierung gelöst werden. Alle benutzten Gesten wurden als intuitiv eingestuft. Als Wunsch wurden weitere Gesten genannt wie z.B. Hände vor der Brust kreuzen, Beinbenutzung oder Hände im Kopfbereich.

6. LESSONS LEARNED

Im Zuge des Projektes konnten die Autoren wertvolle Erfahrungen bzgl. der Projektplanung und der Erforschung neuer bzw. unbekannter Technologien sammeln. Dabei wurde besonders deutlich, von welcher großer Bedeutung stetige Kommunikation untereinander für eine erfolgreiche Projektumsetzung ist.

Außerdem konnten neue und weitreichende Erkenntnisse in Bezug auf interaktive Medien und deren Einsatzmöglichkeiten gesammelt werden. Durch das Erforschen der jeweiligen für die Autoren neuen Technologien, vor allem deren Grenzen, eigneten sich die Autoren weitreichende und zukunftsrelevante Kenntnisse in diesem Gebiet an.

Abschließend kann man sagen, dass die Realisierung des Projektes nicht nur einen Mehrwert für die Fachbereiche bietet, sondern auch für die Autoren eine große Bereicherung war.

7. AUSBLICK

Obwohl alle definierten Anforderungen realisiert werden konnten, können in Zukunft dennoch neue Anforderungen definiert bzw. Erweiterungsvorschläge entwickelt werden. Vom Studienkurs wurde beispielsweise bereits angefragt, ob es möglich wäre, ein anderes bzw. noch besseres Feedback zu entwickeln. Eine Koppelung mit dem MYO-Armband, das mittels Vibrationen ein Feedback vermittelt und dem Nutzer anzeigt, dass eine Geste erkannt und nach einem gewissen Zeitraum (zB. 2 Sekunden) eine Aktion ausgelöst wird, könnte eine immense Aufwertung und Verbesserung der Nutzbarkeit darstellen. Zudem sollte eine Abbruchfunktion, mit deren Hilfe fälschlich erkannte Gesten außer Kraft gesetzt werden, eingeführt werden.

Eine bessere Erkennung der Gelenkpunkte könnte durch fortschrittlichere Hardware verbessert werden, wie z.B. durch die XBOX ONE Kamera mit verbesserter Tiefen- und Farbkamera.

ABBILDUNGSVERZEICHNIS

- Abb. 1 Bsp. aus Präsentation mit NINGU (Area 2)
- Abb. 2 Steve Ballmer. Screenshot: <https://www.youtube.com/watch?v=l14b-C67EXY>
- Abb. 3 Bsp. Präsentation mit Myo. Screenshot: <https://www.youtube.com/watch?v=Zr5OWclA4qs>
- Abb. 4 Bsp. Präsentation mit Kinect https://www.youtube.com/watch?v=e9_ehn3Wk5s
- Abb. 5 Area 1 mit Entry-Zones
- Abb. 6&6a Area 1 Gesten
- Abb. 7 Area 2 Geste
- Abb. 8&8a Area 3 Gesten
- Abb. 9 Codeausschnitt Kamera-Aktivierung
- Abb. 10 Codeausschnitt Skelett-Kamera und JointTypes
- Abb. 11 Screenshot der Symbolleiste
- Abb. 12 Area 1: JointTypes & Pseudocode
- Abb. 13 Area 2: JointTypes & Pseudocode
- Abb. 14 Area 3: JointTypes & Pseudocode
- Abb. 15 Kamerabild & erkannte JointTypes