

Julian Feller, Max Schleuniger

Occlusion-Aware Presenter

Interaction Engineering WS 13/14

05.02.2014

Inhaltsverzeichnis

Abstract	2
Funktionalität	3
Technologien	4
Processing	4
Microsoft Kinect 360	4
SimpleOpenNI	4
OpenCV	4
PBox2d	4
Probleme	5
Blickwinkel der Zuschauer	5
Position der Kinect zur Projektionsfläche	5
Related Work	6
Proxemic Interactions: The New Ubicomp?	6
Occlusion Aware Interfaces	6

Abstract

In unserem Projekt entwickeln wir eine neue Technik aus dem Bereich der Occlusion-Aware Interfaces.

Oftmals besteht während einer interaktiven Präsentation die Situation, dass sich die präsentierende Person vor wichtigen projezierten Inhalten befindet und diese gegenüber dem Publikum verdeckt.

Wir verwenden eine Microsoft Kinect 360 um zu erkennen, ob eine Person die Projektionsfläche verdeckt. Die Elemente, die sich im verdeckten Bereich befinden, werden automatisch auf die sichtbare Fläche verschoben. Dafür verwenden wir die Physik Engine Box2D, beziehungsweise deren Processing Wrapper PBox2D.

Zusätzlich interpretiert unser System, unter Berücksichtigung von Proxemics, Gesten des Präsentierenden und setzt diese direkt um.

Durch die Gesten soll beispielsweise die Fokussierung der Zuschauer auf einen bestimmten Bereich der aktuellen Folie gelenkt werden können.

Die Kalibrierung der Kinect auf die Projektionsfläche funktioniert voll automatisch, das erleichtert es dem Benutzer, das System an unterschiedlichen Orten schnell aufzubauen.

Unser System deckt dabei einen neuen Bereich, der Möglichkeiten der Occlusion Aware Interfaces, ab.

Funktionalität

Mithilfe des Wrappers "simpleopenni" werden die Daten (Webcam und Tiefenkamera) der Kinect empfangen und mit dem Framework "OpenCV" weiterverarbeitet. Die Kalibrierung der Kinect auf die Projektionsfläche des Beamers, erfolgt mit sogenannten Checkerboards. Diese werden dazu auf der Projektionsfläche in allen 4 Ecken angezeigt. Durch OpenCV lässt sich nun errechnen wo die Projektionsfläche anfängt beziehungsweise aufhört.

Je nach Position, an der sich der Präsentierende befindet, reagiert unser Programm unterschiedlich. Dafür haben wir zwei Bereiche definiert:

1. Unmittelbar vor der Bildfläche werden die verdeckten Bereiche sichtbar gemacht:

Mithilfe von OpenCV wird erkannt welche Bereiche der Bildfläche durch den Präsentierenden verdeckt werden, danach wird berechnet welche Objekte sich derzeit in diesen Bereichen befinden. Die Objekte können zum Beispiel Text, Bilder oder Aufzählungen sein.

Hierfür verwenden wir PBox2d, eine Physik Engine die es uns ermöglicht Kollisionen zwischen den Objekten zu erkennen.

Diese werden auf der Oberfläche an einer Achse aufgehängt. Wenn also eine Person Objekte verdeckt dann bewegen sie sich entlang der Achse um im sichtbaren Bereich zu bleiben.

2. Im Abstand von 1.5 Meter zur Bildfläche werden Gesten erkannt und interpretiert

Spotlight-Geste:

Zur Aktivierung der Geste muss der Präsentierende die linke Hand über seinen Kopf halten. Dadurch wird ein Spotlight auf der Präsentationsfläche eingeblendet, welches er mit seinem Arm steuern kann. Wir haben dafür den Vektor von Ellenbogen zur Hand genommen und diesen mit der Projektionsfläche geschnitten.

Wenn sich beide Hände unterhalb der Hüfte befinden wird die Geste wieder deaktiviert.

Technologien

Processing

Processing ist auf die Einsatzbereiche Grafik, Simulation und Animation ausgelegt und basiert auf der Programmiersprache Java. Es stellt eine eigene Entwicklungsumgebung (IDE) zur Verfügung.

<http://www.processing.org/>

Microsoft Kinect 360

Die Kinect besteht aus einer Kombination aus Tiefensensor, 3D-Mikrofon, Farbkamera und Software.

Über den Tiefensensor erkennt die Kinect das Skelett des Benutzers und man bekommt alle Koordinaten übermittelt für Hände, Ellenbogen, Kopf, Rumpf, Füße und Beine. Dadurch konnten wir unsere Gesten interpretieren.

<http://www.xbox.com/de-DE/Kinect>

SimpleOpenNI

SimpleOpenNI ist ein Wrapper für OpenNI und NITE und speziell für Processing entwickelt. Es sind zwar nicht alle Funktionen der beiden Bibliotheken enthalten, dafür bietet SimpleOpenNI aber einen leichteren Zugang zu den Funktionen.

<http://code.google.com/p/simple-openni/>

OpenCV

OpenCV ist ein Bibliothek die Algorithmen zur Bildverarbeitung bietet. Sie ermöglicht unter anderen vielen Dingen auch eine Kalibrierung einer Kamera.

<http://www.opencv.org/>

PBox2d

PBox2D kapselt einige Funktionen der auf Java portierten Version von Box2D. Einer Physikengine die in C++ entwickelt wurde.

<https://github.com/shiffman/PBox2D>

Probleme

Blickwinkel der Zuschauer

Oft ist es bei einer Präsentation der Fall, dass die Zuschauer verteilt im Raum sitzen, also nicht jeder den selben Blickwinkel hat. Dadurch entsteht das Problem, dass Objekte auf der Projektionsfläche für manche sichtbar sind, die für andere jedoch schon im verdeckten Bereich liegen.

Es ist daher unmöglich für jeden Zuschauer eine optimale Darstellung der Inhalte zu erreichen. Am besten ist es wenn das Publikum relativ zentral hinter der Kinect sitzt.

Position der Kinect zur Projektionsfläche

Wenn der Tiefensensor der Kinect schief zur Projektionsfläche steht kommt es zu Problemen, da sich schon Teile der Projektionsfläche selbst im Occlusion Bereich befinden. Dadurch verschiebt sich der komplette Bereich der Proxemics und es kommt zu Fehldarstellungen.

Um eine optimale Darstellung zu erhalten ist es daher nötig die Kinect möglichst senkrecht zur Projektionsfläche zu positionieren.

Related Work

Proxemic Interactions: The New Ubicomp?

Saul Greenberg, Nicolai Marquardt, Till Ballendat, Rob Diaz-Marino, Miaosen Wang

<http://grouplab.cpsc.ucalgary.ca/grouplab/uploads/Publications/Publications/2011-ProxemicInteraction.Interactions.pdf>

Das Paper von Saul Greenberg et. al. beschreibt ein proxemisches Interface. In einem Wohnraum ist ein Fernseher an der Wand aufgehängt, der auf Personen reagiert. Je nach Position der Person im Raum werden unterschiedliche Informationen auf dem Display angezeigt

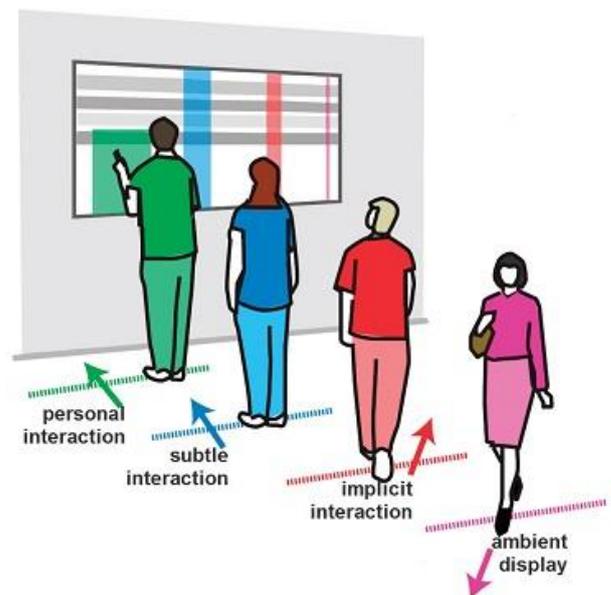
Es wurden vier Zonen definiert auf die das System reagiert. Diese entsprechen im wesentlichen den Zonen die auch zwischen Menschen gelten.

Intimdistanz: bis 45 cm

Persönliche Distanz: 45 bis 120 cm

Soziale Distanz: 120 bis 360 cm

Öffentliche Distanz: ab 360 cm



Occlusion Aware Interfaces

Daniel Vogel, Ravin Balakrishnan

<http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.165.4868&rep=rep1&type=pdf>

Dieses Paper beschreibt ein Interface das verdeckte Bereiche bei der Bedienung eines Tablets erkennt und Objekte dahinter in einen Bereich der sichtbar ist verschiebt.

Jeder Benutzer muss für die Benutzung des Systems seinen digitalen Arm einmalig kalibrieren, damit die Umfänge korrekt dargestellt werden.

