

Full Body Gesture
Recognition & Tracking



Quelle: www.omekinteractive.com

Ganzkörper- Gestensteuerung

Mit 3-D Sensor (z.B. MS Kinect, ASUS Xtion)

Zwischenpräsentation

Gliederung

- Motivation – Wozu braucht man eine Gestensteuerung?
- Aktuelle Anwendungen
- Technologien – Stand der Technik
- Probleme bei der Erkennung von Gesten
- Methoden zur Klassifizierung von Gesten
- MS Kinect
- Software Entwicklung mit Kinect
- Further Work

Motivation

- **Natural User Interaction (NUI)**, weg von Eingabegeräten wie Maus und Tastatur
- Gesten sind schon immer eine Form zwischenmenschlicher Kommunikation
- Spracherkennung unterstützen bzw. ergänzen, z.B. Kommunikation trotz hohem Geräuschpegel im Raum, räumliche Distanz überbrücken
- Kommunikation mit taub-stummen Menschen unterstützen - Gebärdensprache
- Steuerung von Robotern, TV-Geräte, usw. anhand von erkannten Gesten

Aktuelle Anwendungen

- **Unterhaltungsbranche**
Nintendo Wii, MS Kinect
- **Medizin**
steriles browsen von Patientendaten
während OP
- **Augmented Reality - AR**
computergestützte Erweiterung der
Realitätswahrnehmung, meist visuell
z.B. Computer-Spiele
- **Industrie**
Forschungsprojekte zur Steuerung
von Robotern durch Gesten



Bildquelle: [8]



Bildquelle: [11]

Technologien – Stand der Technik

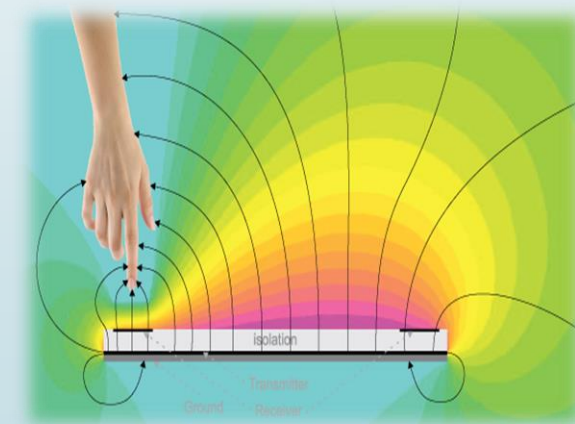
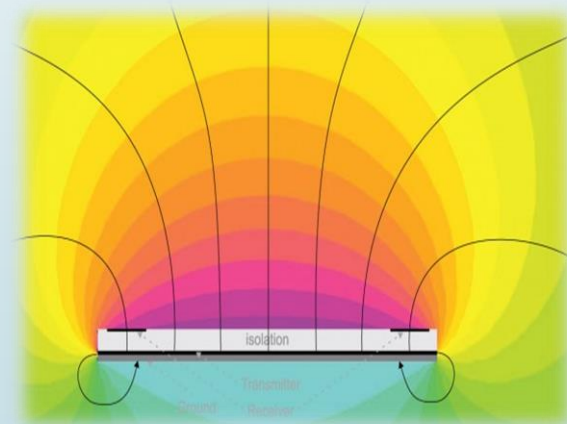
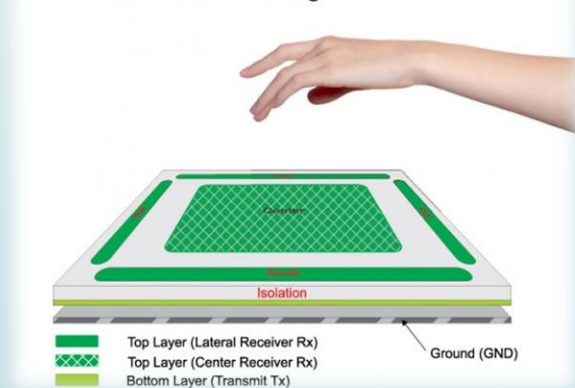
Gerätebasierte Gestenerkennung

- Am Körper getragene/ mit der Hand geführte Sensorik (Beschleunigungs-, Positionssensor)
Bsp: Wii-Sensor, Datenhandschuh
- E-Feld Sensortechnologie für 3D Gestenerkennung.^[2]



Bildquelle : www.ffh-wildau.de

Standard Electrode Design



Bilderquelle: Microchip Technology

Technologien – Stand der Technik

Kamerabasierte Gestenerkennung 2D

- ▶ Kameragestützte Systeme, um Bilder der Benutzer zu erstellen (eine oder mehrere Kameras möglich)
- ▶ Anwendung von 2D Bildanalyse Techniken
Bsp: Finger Tip Erkennung (2D) zur Steuerung von Mediengeräten

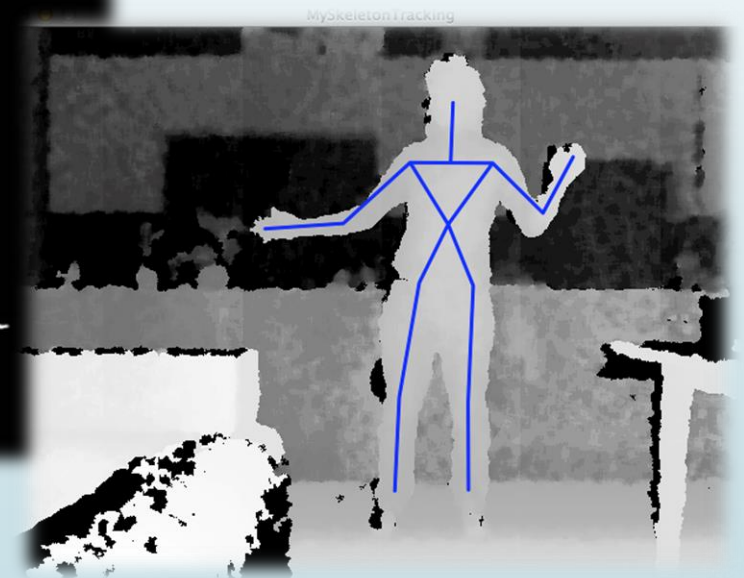


Bildquelle: Eyesight, www.cnet.com

Technologien – Stand der Technik

Kamerabasierte Gestenerkennung 3D

- Anwendung von 3D Bildanalyse Techniken -> Tiefensensor
Bsp: MS Kinect, Asus Xtion, Leap Motion



Bilderquellen: learning.codasign.com, golancourses.net

Technologien – Stand der Technik

Funk basierte Gestenerkennung

- Wisee Technik (Whole-Home Gesture Recognition Using Wireless Signals)[3]



Bildquelle: wisee.cs.washington.edu

- Funktioniert auf der Basis von Doppler-Effekt.

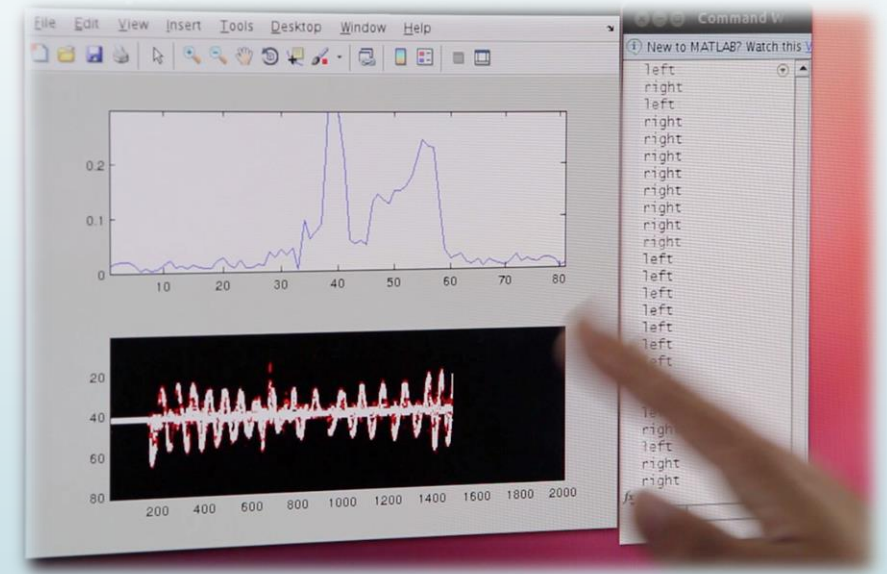
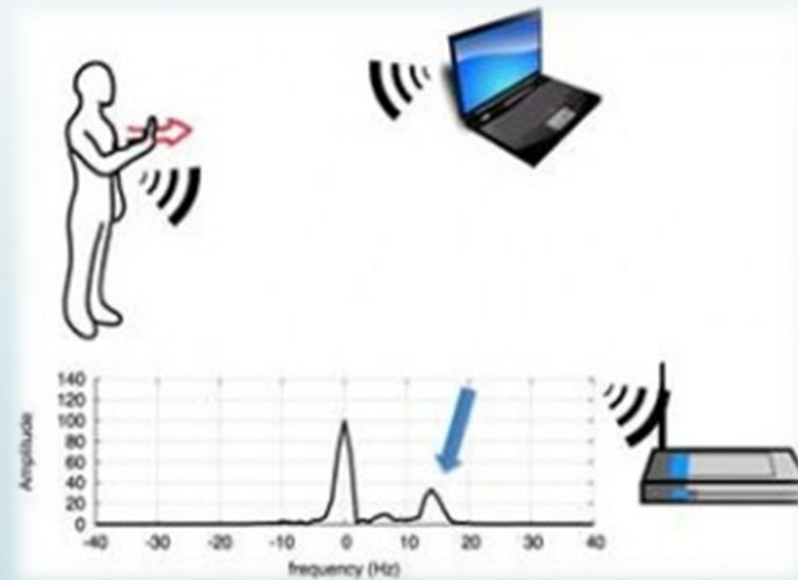


Bildquelle: wikipedia/Doppler-Effekt

Technologien – Stand der Technik

Funkbasierte Gestenerkennung

► Prinzip der Wisee-Technik



Bilderquelle: washington.edu

Gliederung

- Motivation – Wozu braucht man eine Gestensteuerung?
- Aktuelle Anwendungen
- Technologien – Stand der Technik
- Probleme bei der Erkennung von Gesten
- Methoden zur Klassifizierung von Gesten
- MS Kinect
- Software Entwicklung mit Kinect
- Further Work

Problematik bei der Erkennung von Gesten

- Gleiche Geste -> unterschiedliche Ausführung von verschiedenen Personen
- Individuelle Interpretationsspielräume im Bezug auf Gesten
- Physische, persönliche Merkmale wie Größe und Armlänge der Nutzer
- Abstand im 3D Raum zwischen Nutzer und System

Folge: Erkennungssystem muss anhand eines begrenzten Trainingsgestensatzes in der Lage sein, auch Personenübergreifend Gesten zu erkennen.

Verfahren zur Gestenklassifizierung

- ▶ **Hidden Markov Modell (HMM)**
tatsächliche Systemzustände sind verborgen (hidden), jedem inneren Zustand sind beobachtbare Ausgabesymbole (Emissionen) zugeordnet, die je nach Zustand mit gewissen Wahrscheinlichkeiten auftreten. Aufgabe: aus beobachteten Sequenzen der Emissionen zu Aussagen über die verborgenen Zustände zu kommen (Bsp. Gefangener im Verließ) [5]
- ▶ **Slow Feature Analysis (SFA)**
aus zeitabhängigen Daten wichtige Komponenten herausfiltern und redundante Daten vernachlässigen (wenig Trainingsdaten, Personenunabhängig) [7]
- ▶ **Dynamic Time Wrapping**
Algorithmus der die Gemeinsamkeit von zwei Signalen unterschiedlicher Länge herausfiltert [6]
z.B. Spracherkennung -> unterschiedlich lange Betonung von Vokalen
z.B. Gestenerkennung -> unterschiedliche Geschwindigkeit bei Gesten
- ▶ **Feature Vektoren**
3D Gestendaten aufzeichnen, z.B. MS-Kinect, zusätzliche Features berechnen -> Feature Vektor -> Training verschiedener Klassifizierer [7]

Microsoft Kinect Sensor

► RGB-Kamera

8 Bit VGA Sensor,
30 fps mit 640x480 Pixel,
15 fps (USB Limit) mit 1280x1024 Pixel
mit speziellen Treibern

► IR-Kamera

monochromatisch lichtempfindlicher CMOS Sensor,
30 fps mit 640 x 480 Pixel,
Sichtfeld: 57° horizontal, 43° vertikal,
Reichweite laut MS: 0,8 – 3,5 m
andere Treiber: 0,6 – 10,0 m
Genauigkeit nimmt proportional zum Quadrat der Entfernung ab
Auflösung bei 2 m Abstand = 1 cm

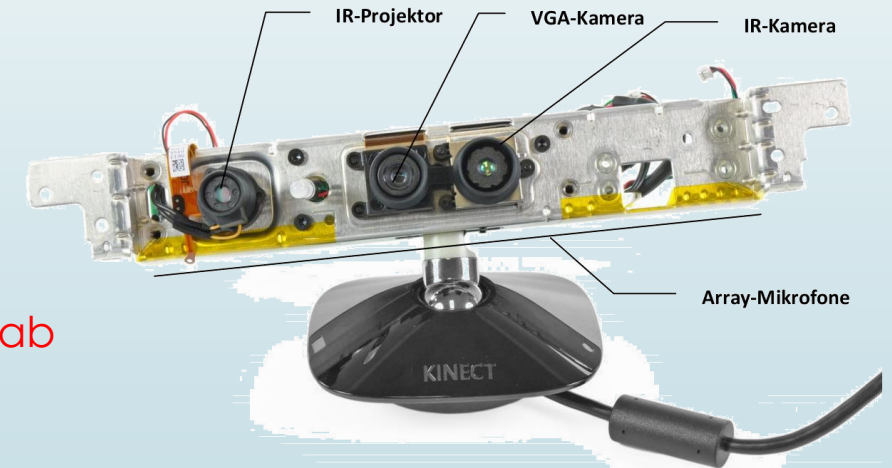
Quelle für Kinect Daten: [10]

► Infrarot-Projektor

erzeugt Lichtpunktswolke,
 $\lambda = 830 \text{ nm}$,
Laserklasse 1,
Leistung 60mW

► Array Mikrofone

4 Stück,
je 16 bit
Sample-Rate 16kHz



Bildquelle: [8]

Software Entwicklung mit Kinect

1) **OpenNi + NITE**

OpenNi Framework bietet open Source APIs. Unterstützt Spracherkennung, Handgesten und Ganzkörper Bewegungs Tracking
PrimeSense hat eigene open source Treiber veröffentlicht + Motion Tracking
Middleware NITE
-> Win, Linux, Mac

2) **OpenKinect**

Ist eine offene Community die an Treibern für die Kinect arbeitet, aktuell
Libfreenect
-> Win, Linux, Mac

3) **Microsoft Kinect for Windows**

SDK und APIs für Entwicklung unter Windows

4) **Code Laboratories**

Die CL NUI bietet Treiber und SDK für die Kinect unter Windows

Quelle: [12]

Further Work

- ▶ Entwicklungsumgebung unter Linux / VMware installieren
- ▶ Test der HW und SW anhand von Beispielprogrammen
- ▶ Entwurf / Test der Kommunikation zwischen Applikation und SCITOS
- ▶ Entwicklung einer Gestenerkennung
 - ▶ Konzept (SW Architektur, Klassifizierung von Gesten...)
 - ▶ Erstellung eines Gesten-Portfolios (Auswahl von 3-5 Gesten)
 - ▶ Kalibrier-Funktion notwendig? Wenn ja wie?
 - ▶ Ansteuerung der SCITOS Funktionen (z.B. fahren, stopp, ...) nach erkannter Geste
- ▶ Implementierung einer vorführbaren Applikation
- ▶ Erkennung von Personen anhand von Gesten
- ▶ User-Interface (Visualisierung) zum einlernen neuer Gesten
- ▶ Dokumentation

Quellen

- [1] Jean, Jared St.: Kinect Hacks. 1. Aufl.. Sebastopol: "O'Reilly Media, Inc.", 2012.
- [2] <http://www.microchip.com/pagehandler/en-us/technology/gestic/technology/home.html>
- [3] <http://wisee.cs.washington.edu/>
- [4] <http://www.heise.de/netze/meldung/Gestensteuerung-per-WLAN-1885044.html>
- [5] http://de.wikipedia.org/wiki/Hidden_Markov_Model
- [6] <http://de.wikipedia.org/wiki/Dynamic-Time-Warping>
- [7] Renee Schulz, Entwicklung und Vergleich von Verfahren zur Verbesserung der Gestenerkennung für den Einsatz in Natural User Interfaces, Master-Thesis, Fachhochschule Köln, August 2013
- [8] http://wiki.zimt.uni-siegen.de/fertigungsautomatisierung/index.php/Einsatzm%C3%B6glichkeiten_einer_3D-Kamera_in_der_Produktionstechnik_am_Beiispiel_der_Kinect-Kamera#Aufbau_und_Funktionsweise
- [9] http://openkinect.org/wiki/Imaging_Information
- [10] https://www.iosb.fraunhofer.de/servlet/is/6635/MarkusTirpitz_Studienarbeit.pdf
- [11] <http://structure.io/>
- [12] <http://joantziouvara.wordpress.com/category/code-laboratories-cl-nui-kinect/>

Vielen Dank für Eure Aufmerksamkeit!

