

Entwicklung einer Gestiksteuerung mittels Kinect für den humanoiden Roboter Nao

STUDIENARBEIT

für die Prüfung zum
Bachelor of Engineering
des Studienganges Informationstechnik
an der
Dualen Hochschule Baden-Württemberg Karlsruhe

von
Lukas Essig
und
Michael Stahlberger

Abgabedatum 12. Mai 2014

Bearbeitungszeitraum	12 Wochen
Matrikelnummer	8898018, 1367912
Kurs	TINF11B3
Ausbildungsfirma	Fiducia IT AG Karlsruhe
Betreuer	Herr Prof. Hans-Jörg Haubner

Erklärung

Gemäß §5 (3) der „Studien- und Prüfungsordnung DHBW Technik“ vom 22. September 2011.

Ich habe die vorliegende Studienarbeit selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel verwendet.

Ort Datum

Unterschrift

Sperrvermerk

Die nachfolgende Arbeit enthält vertrauliche Daten und Informationen der Fiducia IT AG.

Veröffentlichungen oder Vervielfältigungen -auch nur auszugsweise- sind ohne ausdrückliche schriftliche Genehmigung des Unternehmens nicht gestattet.

Die Arbeit ist nur den Korrektoren sowie den Mitgliedern des Prüfungsausschusses zugänglich zu machen.

Inhaltsverzeichnis

1	Einleitung	1
	Liste der ToDo's	1
2	Kinect	2
2.1	Kinect Hardware	2
2.2	Kinect Software	2
2.3	Kinect SDK	3
3	Realisierung	4
3.1	Architektur	4
3.2	Prototyp	4
	Literaturverzeichnis	5
	Liste der ToDo's	6

Abbildungsverzeichnis

Tabellenverzeichnis

Abkürzungsverzeichnis

CFS	Completely Fair Scheduler
EDD	Earliest Due Date
FCFS	First come - First served
IS25TO	Transaktionen und Operation-Tools
ITS	IT Services
LIFO	Last In - First Out
SJF	Shortest Job First

1 Einleitung

Liste der ToDo's -Ziele d. Studienarbeit, -

2 Kinect

Im folgenden Kapitel wird der Aufbau und die Funktion des Kinect Sensors erläutert.

2.1 Kinect Hardware

Microsoft entwickelte 19nn ein Produkt namens XBox Kinect. Dieses Produkt war ursprünglich als Zusatzhardware für die XBox 360 gedacht.

[2]

To do (2) To do (3) To do (4)

2.2 Kinect Software

Das Produkt Xbox Kinect kam bereits vor einigen Jahren auf den Markt. Somit hatten die Entwickler Zeit, um ein SDK zu entwickeln, was alle wichtigen Programmfunktionen bereits enthält. Dies macht es einem Entwickler relativ leicht eine Anwendung zu erstellen, die bestimmte Kinect-Funktionen bereitstellt. Die Hardware liefert folgende Werte: -Audiosignal von n Microphonen -RGB Bildsignal der Kamera -Tiefensignal (wobei von Treiber errechnet) -¿ IR Tiefenscan... näher erläutern

Für dieses Projekt sind die Audiodaten jedoch irrelevant. Wichtig sind primär die Bilddaten inklusive Tiefenwerte. Das Kinect SDK bietet bereits standardmäßig Zugriff auf die grundlegenden Funktionen: Dabei können bestimmte Proxyobjekte registriert und abgerufen werden. Die wichtigste Implementierung eines dieser Proxyobjekte ist für dieses Projekt der Skeletonstream.

Dieser Stream berechnet anhand der Tiefendaten und der RGB-Daten werte für ein Menschliches Skelett:

-Armwinkel -Positionen -Kinectraum...

DELME [1]

To do (5) To do (6) To do (7) To do (8)

2.3 Kinect SDK

3 Realisierung

In diesem Kapitel wird näher auf das Zusammenspiel von Sensorik (Kinect) und Aktorik (Nao) eingegangen. Zunächst wird die Idee der Softwarearchitektur vorgestellt. Im Anschluss daran, wird auf die entworfenen Algorithmen näher eingegangen.

To do (9) To do (10) Wahl Bibliothek, warum?... To do (11) To do (12) To do (13)

3.1 Architektur

Die Architektur der Anwendung kann grob in zwei Teilbereiche gegliedert werden, den Sensorteil und den Aktorteil. Der Sensorteil ist dafür zuständig, die entsprechenden Gesten des Benutzers zu ermitteln und zu verarbeiten. Der Aktorteil der Anwendung ist dafür zuständig die Werte entsprechend auf die Wertebereiche des Nao-Koordinatensystems zu transformieren und diese dem Roboter zu übermitteln.

To do (14) To do (15) To do (16)

3.2 Prototyp

Anhand der obigen Überlegungen wurde der erdachte Algorithmus zunächst anhand eines Prototyps implementiert. Dieser sollte zunächst dazu dienen, das Microsoft Kinect SDK näher kennen zu lernen^{To do (17)}. Der erste Prototyp erfüllte folgende Funktionen:

-Kinect SDK Library -(Connect-Disconnect von Kinect) -Winkelerkennung des Benutzers mit Anzeige in Grad -Anzeige des Kamerabildes mit Gelenkkennzeichnung von Kopf und Armen

To do (18)

Literaturverzeichnis

- [1] Joachim Hertzberg. *Mobile Roboter : Eine Einführung aus Sicht der Informatik*. Springer, Berlin, 2009.
- [2] Jarrett Webb. *Beginning Kinect programming with the Microsoft Kinect SDK*. Apress, New York, 2012.

Liste der ToDo's

- ☒ 1 (p. 2): Kapitel schreiben
- ☐ 2 (p. 2): -Erstmals preiswerter 3D Sensor -Zunächst für Spielkonsole -Später für PC erhältlich -Aufbau: -Microphonarray zweck=richtung d. Geräusch -IR Sensor = Deep Stream -RGB Kamera = Bild -Rauminformationen für Software -Vorteil: Günstig, leicht zu entwickeln, da SDK vorhanden
- ☐ 3 (p. 2): Hardware Bilder
- ☐ 4 (p. 2): erklärungen zu Bilder
- ☐ 5 (p. 2): SDK Funktionen erklären: skelton, mehrere user, deep stream, color stream
- ☐ 6 (p. 2): Screenshot von Deep Stream (Kinect Studio)
- ☐ 7 (p. 2): Unterschied Microsoft SDK und Freie Implementierung OpenNI
- ☐ 8 (p. 2): Bild von Kinect-Koordinatensystem -¿ Bezug auf Nao
- ☐ 9 (p. 4): Wahl Programmiersprache (warum? -¿Kinect und Nao gleich, Visual Studio)
- ☐ 10 (p. 4): (
- ☐ 11 (p. 4): Erklärung Webots, Choreograph, Nao Verbindung, Libraries...
- ☐ 12 (p. 4): Danach -¿ echter Nao im Laber -¿ Unterschied zur Simulation
- ☐ 13 (p. 4): Laborbedingungen Steuerungen, Vor- und Nachteile der Bedienung

- ☐ 14 (p. 4): Architekturidee: Input, Verarbeitung(evtl. Filter-; Probleme Ruckeln?), Output,
- ☐ 15 (p. 4): Klassendiagramm
- ☐ 16 (p. 4): Sequenzdiagramm eines bestimmten Anwendungsfalls
- ☐ 17 (p. 4): getrennt zusammen?
- ☐ 18 (p. 4): Screenshot von Prototypprogramm