



# AW 1 - Vortrag

Hochschule für Angewandte Wissenschaften Hamburg  
Hamburg University of Applied Sciences

## Simulationsmodell für visuell geführte Roboter

von Bernd Pohlmann

Betreuender: Prof. Dr. Andreas Meisel

# Inhalt

1. Motivation
2. Ziel
3. Einführung Robotik
4. Kinematik
5. Denavit-Hartenberg
6. Kameramodell
7. Simulation
8. Ausblick

# Motivation

## Wozu ein Simulationsmodell?

- ◆ **Sicheres** und schnelles Testen von Algorithmen
- ◆ Vereinfachung auf das Wesentliche
- ◆ Ausschliessen von Störfaktoren wie physikalischen Gesetzen

## Warum visuell geführt?

- ◆ Keine Sensorik (ausser der passiven Kamera)
- ◆ Genauere Kinematik



[linuxfordevices.com]

# Ziel

Eine vollständige Simulationsumgebung für Roboter aller Bauarten (in Matlab)

- ◆ Roboterplattform
- ◆ 3D Umgebung
- ◆ 3D Objekte
- ◆ Kinematik
- ◆ (Physik)



+



= ?

## Zielkontext

- ◆ Assistenzroboter in der iFlat [i(Flat)Robot]
- ◆ Einfache Schnittstelle zu anderen Arbeiten (SIFT, inverse Kinematik, Bahnplanung)

# Einführung Robotik(1/3)

- ◆ Klassifizierung
  - ◆ stationäre Roboter
    - ◆ Industrieroboter (KUKA), Cenadarm2, *Katana*
    - ◆ Assistenzroboter in der med. Anwendung
  - ◆ mobile Roboter
    - ◆ Logistik- & Überwachungssysteme
    - ◆ autonome Fahrzeuge
    - ◆ *Assistenzroboter* für Haushalt, Pflege, Militär



[Justin - DLR]



# Einführung Robotik(2/3)

- ◆ Die drei Gesetze der Robotik von Asimov (iRobot)
  - Ein Roboter darf keinen Menschen verletzen oder durch Untätigkeit zu Schaden kommen lassen.
  - Ein Roboter muss den Befehlen eines Menschen gehorchen, es sei denn, solche Befehle stehen im Widerspruch zum ersten Gesetz.
  - Ein Roboter muss seine eigene Existenz schützen, solange dieser Schutz nicht dem ersten oder zweiten Gesetz widerspricht.
  
- ◆ Spezifikation des Katana Roboters
  - Knickarmroboter mit Standfuss und Kamera an seiner Hand
  - Nutzlast: 500g
  - Steuerung durch PC



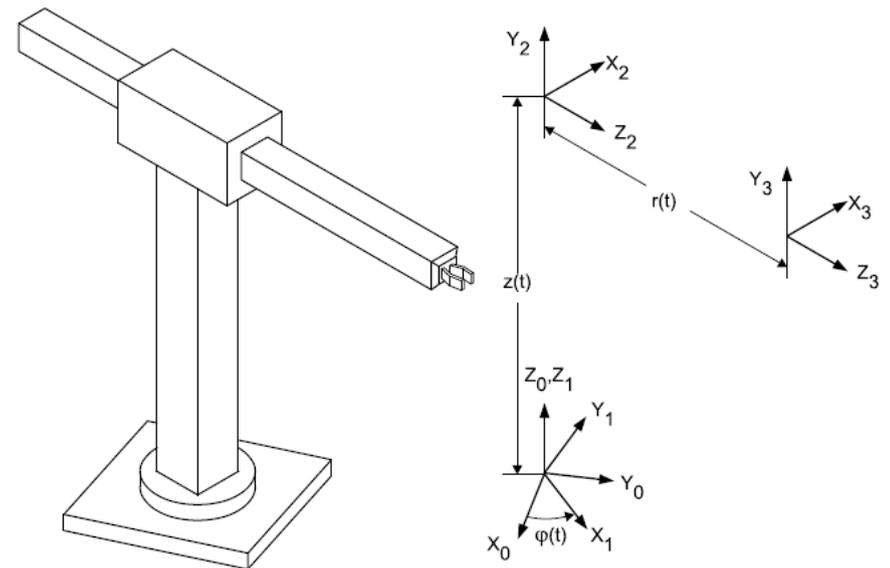
# Einführung Robotik(3/3)

## ◆ Freiheitsgrade

- ◆ Ein freier starrer Körper hat 6 Freiheitsgrade (DOF).
- ◆ Eine Pose ergibt sich aus Position + Orientierung.
- ◆ Eine kinematische Kette mit mehr als 6 DOF ist redundant.

## ◆ Koordinatensysteme

- ◆ Basiskoordinatensystem
- ◆ Weltkoordinatensystem
- ◆ (End-)Effektorkoordinatensystem (TCP)
- ◆ Werkstückkoordinatensystem



# Kinematik(1/2)

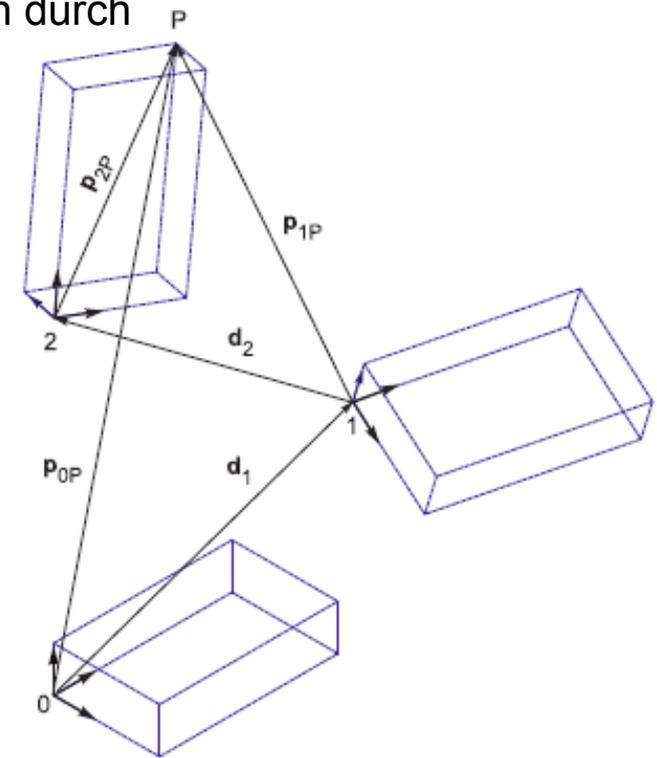
## ◆ Lokalisation und Orientierung

◆ Die Lage eines Körpers im Raum wird beschrieben durch

- ◆ einen Positionsvektor  $d$ , der die translatorische Verschiebung bezüglich eines Koordinatensystems angibt und
- ◆ einer Orientierungsmatrix  $R$ , die die Verdrehung bezüglich des Koordinatensystems angibt.

◆ Eulersche Winkel  $(\alpha, \beta, \gamma)$

◆ Einführung homogener Koordinaten

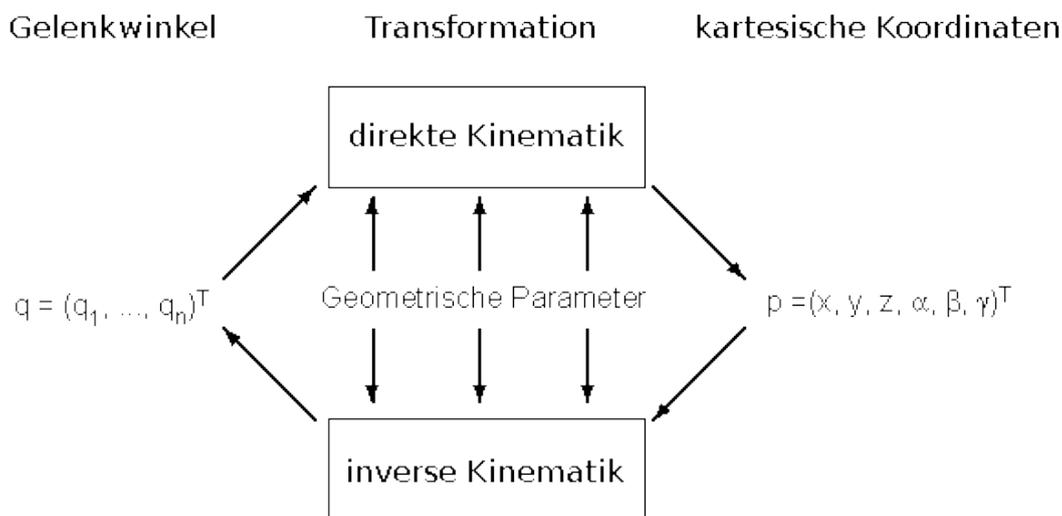


$$\begin{aligned}
 {}^0p_{0P} &= {}^0d_1 + {}^0p_{1P} \\
 &= {}^0d_1 + {}^0R_1^1 p_{1P} \\
 &= {}^0d_1 + {}^0R_1({}^1d_2 + {}^1R_2^2 p_{2P}) \\
 &= {}^0d_1 + {}^0R_1 d_2 + {}^0R_1 R_2^2 p_{2P}
 \end{aligned}$$

$${}^0r_{0P} = {}^0T_1^1 T_2^2 r_{2P}$$

# Kinematik(2/2)

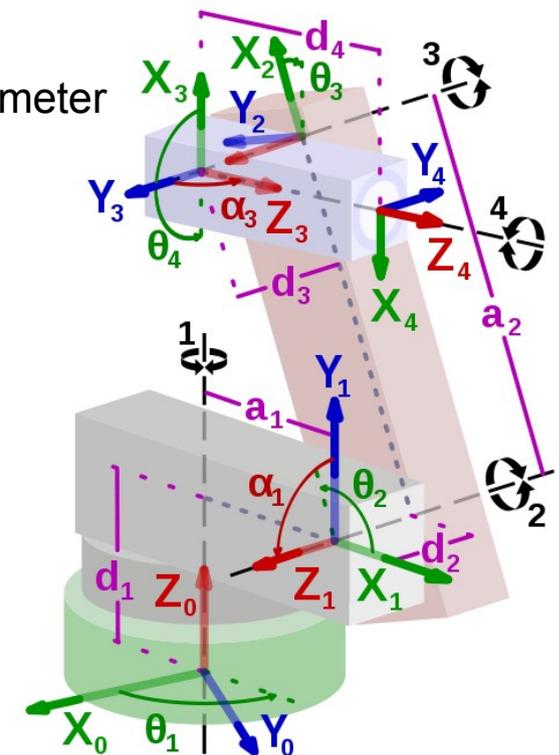
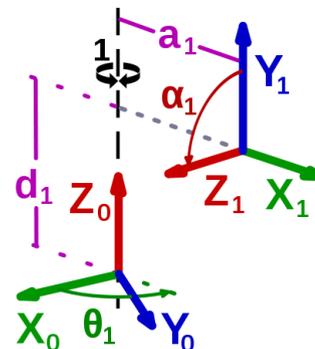
- ◆ direkt
  - ◆ Das Einstellen bestimmter Winkel für die einzelnen Gelenke führt zu einer Pose.  
(*einfach*)
- ◆ indirekt
  - ◆ Eine Pose ist vorgegeben und die Gelenkwinkel werden (iterativ) ermittelt  
(*komplex*) [Christoph Schmiedecke]



# Denavit-Hartenberg(1/2)

## Das Standardverfahren zur Berechnung der direkten Kinematik

- ◆ Jeder Teilkörper eines Roboters bekommt ein festes Koordinatensystem ( $K_0$  bis  $K_n$ )
- ◆ Ziel: Beschreibung der Pose eines Werkzeugs in Basiskoordinaten
- ◆ Beschreibung der *offenen kinematischen Kette* durch 4 x DOF Parameter
  - ◆  $\Theta$  – Gelenkwinkel um die  $z_{n-1}$  Achse
  - ◆  $d$  – Gelenkabstand entlang der  $z_{n-1}$  Achse
  - ◆  $a$  – Armelementlänge entlang der  $x_{n-1}$  Achse
  - ◆  $\alpha$  – Verwindung um  $x_n$

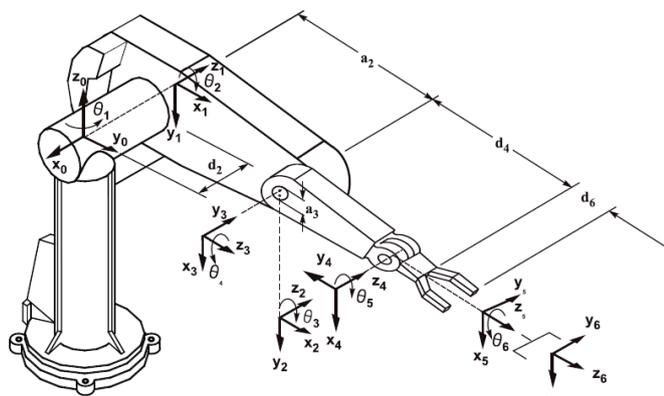


# Denavit-Hartenberg(2/2)

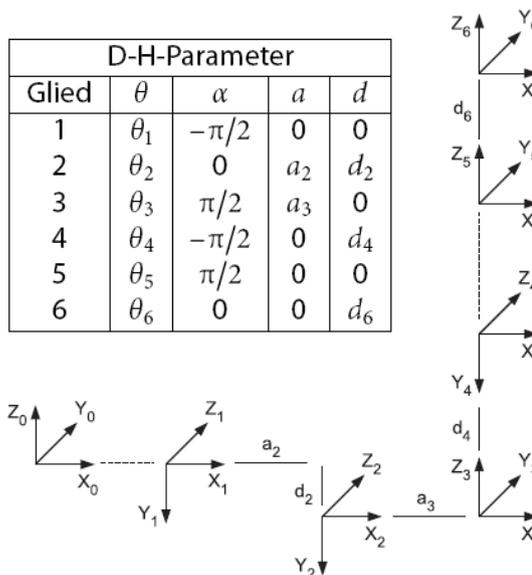
Vereinfacht bilden die beiden Translationen und Rotationen folgende Matrix:

$${}^{i-1}T_i = \begin{pmatrix} \cos \theta_i & -\sin \theta_i \cos \alpha_i & \sin \theta_i \sin \alpha_i & a_i \cos \theta_i \\ \sin \theta_i & \cos \theta_i \cos \alpha_i & -\cos \theta_i \sin \alpha_i & a_i \sin \theta_i \\ 0 & \sin \alpha_i & \cos \alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Am Beispiel:



D-H-Parameter				
Glied	$\theta$	$\alpha$	$a$	$d$
1	$\theta_1$	$-\pi/2$	0	0
2	$\theta_2$	0	$a_2$	$d_2$
3	$\theta_3$	$\pi/2$	$a_3$	0
4	$\theta_4$	$-\pi/2$	0	$d_4$
5	$\theta_5$	$\pi/2$	0	0
6	$\theta_6$	0	0	$d_6$



*Somit kann die Position des Endeffektors (TCP) in Basiskoordinaten ausgedrückt werden.*

# Kameramodell(1/2)

## Problemstellung:

- ◆ Ein Objekt hat eine Position und Orientierung im Raum
  - Objekt-/Werkstückkoordinaten
- ◆ Die Kamera ist auf dem Roboterarm befestigt und somit mobil
  - Kamerakoordinaten
- ◆ Ein mobiler Roboter hat ebenfalls eine Pose
  - Basiskoordinaten

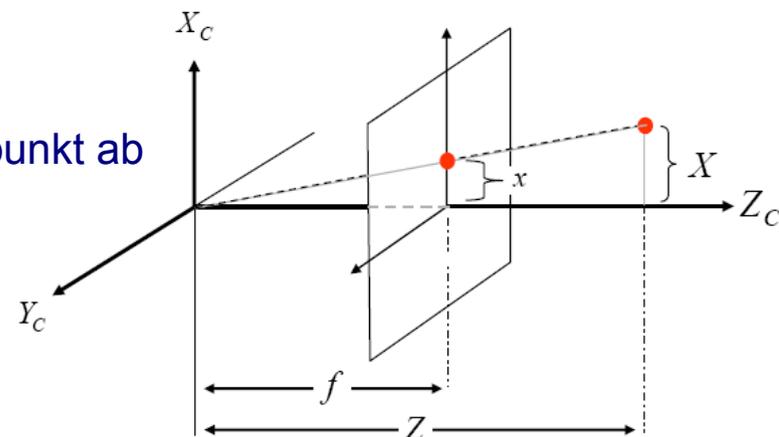
## Ziele:

- Die Bestimmung der Objektkoordinaten( $x, y, z, \alpha, \beta, \gamma$ ) über die Kamera relativ zum Weltkoordinatensystem (Raum)
- Hand-Auge-Koordination des Roboters

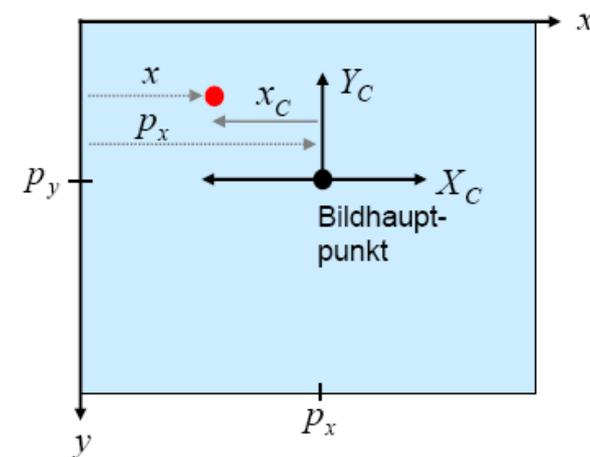
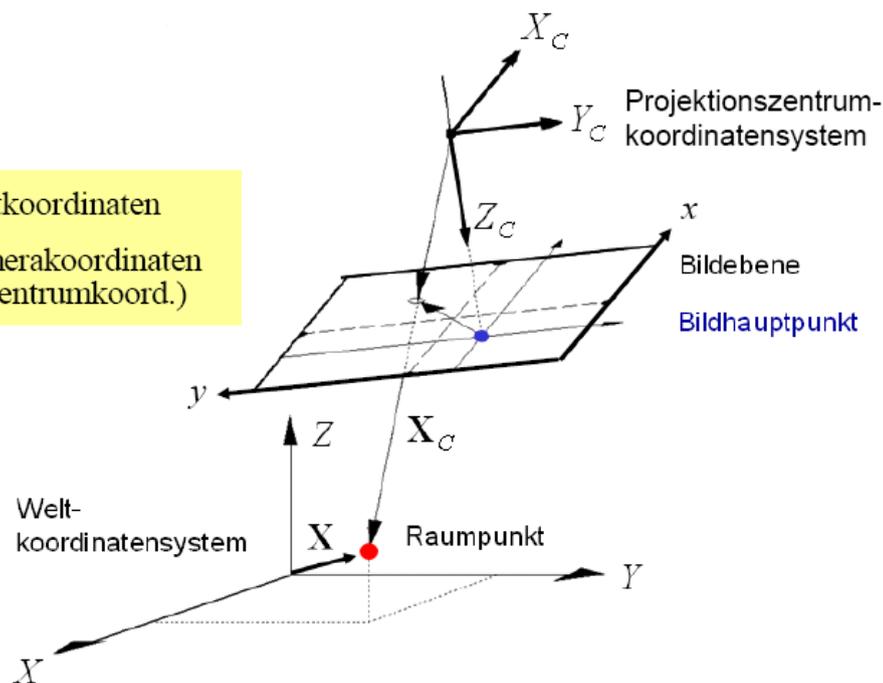
# Kameramodell(2/2)

## Voraussetzungen:

- Die Kamera bildet einen Raumpunkt auf einen Bildpunkt ab
- Der Ursprung des Bildkoordinatensystems ist der Durchstoßpunkt der optischen Achse

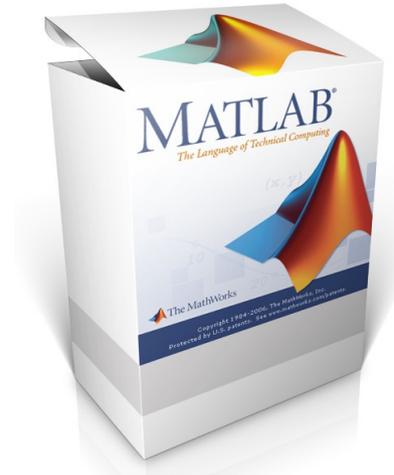


**X** : Punkt in Weltkoordinaten  
**X<sub>C</sub>** : Punkt in Kamerakoordinaten (Projektionszentrumkoord.)



# Simulation(1/3)

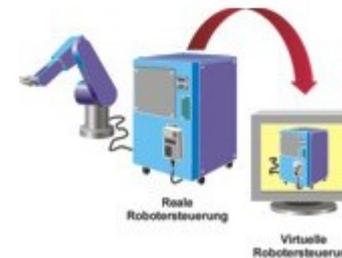
Mögliche Tools:



Microsoft®  
Robotics  
Developer  
Studio 2008



Verwandte Projekte:



# Simulation(2/3)

## Simulation des Katana (abstrakt)

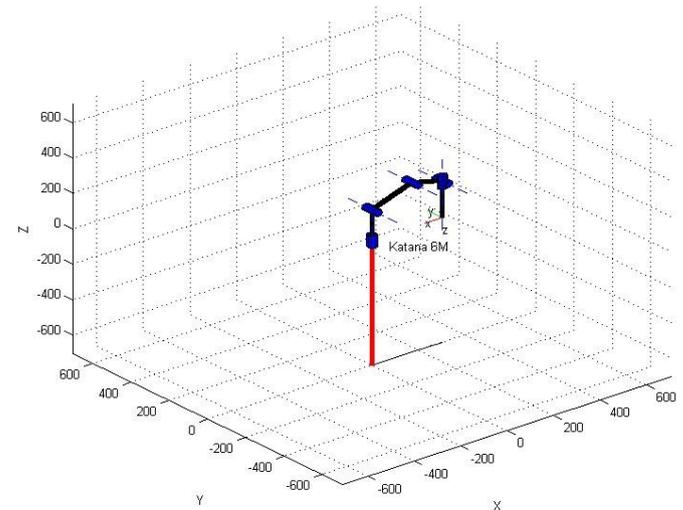
- ◆ Matlab Robotics Toolbox > Umgebung
- ◆ Abmessungen > Denavit-Hartenberg Parameter
- ◆ Gelenkbegrenzungen > die Grösse des Raumes
- ◆ ??? > Kinematik



```

l1=171;l2=190;l3=138;l4=206.1;
+ L1=link([+pi/2  0  0  l1  ]); =>
  L2=link([ 0    12  0  0  ]);
  L3=link([ 0    13  0  0  ]);
  L4=link([-pi/2  0  0  0  ]);
  L5=link([0     0  0  l4  ]);

```



## Simulation des Katana (konkret) >>> live Simulation

# Simulation(3/3)

## Simulationsziel:

Ein Roboter mit

- ◆ *austauschbarer Plattform* bewegt sich in
- ◆ einem *modellierten Raum* und kann
- ◆ mit seiner *Kamera*
- ◆ *Objekte identifizieren* und
- ◆ diese mit seinem Endeffektor *greifen*.

Um dies zu erreichen, sollen u.a. die Ergebnisse der Arbeiten zum Thema

- ◆ Inverse Kinematik (Schmiedecke),
- ◆ Bahnplanung (Teske),
- ◆ 3D BV / S.I.F.T. (Fries) und
- ◆ 3D Objekterkennung (Wagner) implementiert werden.

# Ausblick

## Zukünftige Aufgaben umfassen:

- ◆ Modellierung des Raumes (zunächst ohne Physik)
- ◆ Modellierung von Objekten (RayTracing)
- ◆ virtuelles Greifen von Objekten

## Vision:

- ◆ parallele Entwicklung aller Komponenten (n-Simulationen)
- ◆ Stark verringerte Entwicklungs-/Testzeit von neuen Algorithmen bei erhöhter Genauigkeit der Ergebnisse.

# Literatur

- ◆ Springer – *Handbook of Robotics (2008)*
- ◆ Prof. Dr. Röhrig (FH Dortmund) – *Robotik (Vorlesungsfolien)*
- ◆ Prof. Dr. Meisel (HAW Hamburg) – *3D Bildverarbeitung (Vorlesungsfolien)*
- ◆ Prof. Dr. Jianwei Zhang (UNI Hamburg) – *Einführung in die Robotik*
- ◆ <http://www.wikipedia.de/> (05.12.09) – *Denavit-Hartenberg  
Kinematik*

# The End

Vielen Dank  
für Ihre Aufmerksamkeit!