

# Einführung in die Robotik

**Jianwei Zhang**  
zhang@informatik.uni-hamburg.de

**T | A**  
**M | S**  
Universität Hamburg  
Fakultät für Mathematik, Informatik und Naturwissenschaften  
Department Informatik  
Technische Aspekte Multimodaler Systeme

12. Juli 2011

## Gliederung

- Allgemeine Informationen
- Einführung
- Koordinaten eines Manipulators
- Kinematik-Gleichungen
- Inverse Kinematik von Manipulatoren
- Differentielle Bewegungen mit homogenen Transformationen
- Jacobi-Matrix eines Manipulators
- Aufgabenbeschreibung
- Robotergrammierung auf drei Ebenen
- Trajektoriegenerierung
- Trajektoriegenerierung
- Einführung in RCCL
- Dynamik

## Gliederung (cont.)

- Roboterregelung
- Programmierung auf Aufgabenebene und Bahnplanung
- Programmierung auf Aufgabenebene und Bahnplanung
- Programmierung auf Aufgabenebene und Bahnplanung
- Architekturen sensorbasierter intelligenter Systeme
  - Architekturen sensorbasierter intelligenter Systeme
  - Das Perzeption-Aktion-Modell mit Gedächtnis
  - Das CMAC-Modell
  - Die Subsumtions-Architektur
  - Steuerungsarchitektur eines Fisches
  - Verhaltensfusion
  - Hierarchie
  - Eine Architektur für lernende Roboter

## Gliederung (cont.)

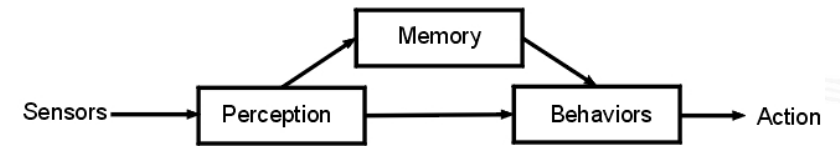
- Das AuRA-Modell - Arkin '86
- Aus- und Rückblick

## Architekturen sensorbasierter intelligenter Systeme

### Themenübersicht

- ▶ Basisverhalten
- ▶ Verhaltensfusion
- ▶ Subsumtion
- ▶ Hierarchische Entwürfe
- ▶ Interaktive Architekturen

## Das Perzeption-Aktion-Modell mit Gedächtnis



## Das CMAC-Modell - I

(Englisch)

CMAC: "Cerebellar Model Articulation Controller"

The input vector **S** is modelled as sensory firing cell patterns. The combination of the cell patterns produces an association vector **A**. This association cell vector multiplied by the weight matrix **W** produces a response vector **P**.

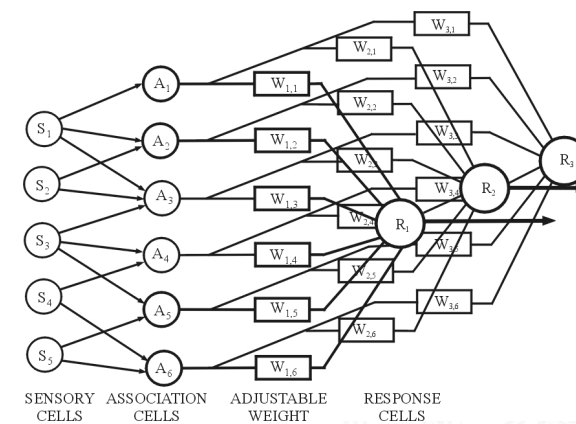
The CMAC model can be viewed as two mappings:

$$f : \mathbf{S} \rightarrow \mathbf{A}$$

$$g : \mathbf{A} \rightarrow \mathbf{P}$$

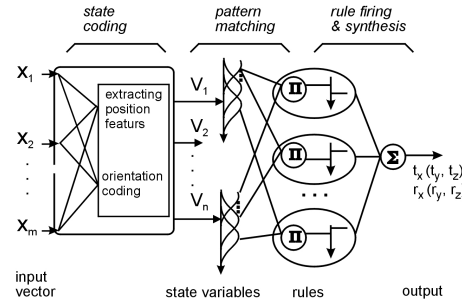
**S** = {sensory input vectors}  
**A** = { association cell vectors}  
**P** = { response output vectors }

## CMAC - II

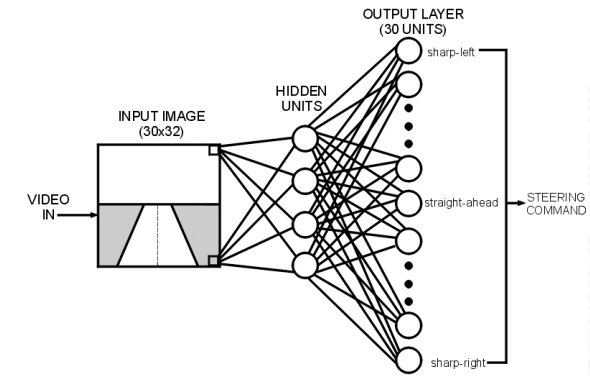


## B-Spline-Modell

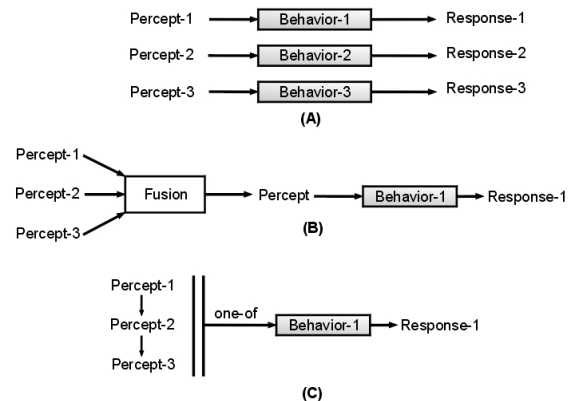
The B-Spline model is an ideal implementation of the CMAC-Model. The CMAC model provides an neurophysiological interpretation of the B-Spline model.



## Alvinn - Visuelle Navigation CMU



## Handlungsorientierte Perzeption



## Die Subsumtions-Architektur

(Englisch, R. Brooks)

A robot program employing the modelling/planning paradigm is composed of a sequence of steps. These functional units transform a snapshot of sensory data into a series of actions intended to achieve a specified goal.

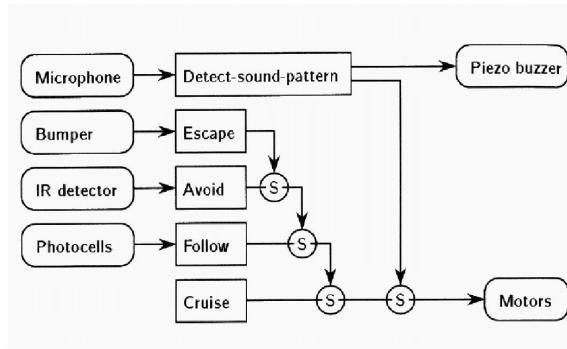
A subsumption architecture for a mobile robot "Rug Warrior" begins with a behaviour called "Cruise" (moving forward).

"Follow" is triggered by photo-cells to move toward right.

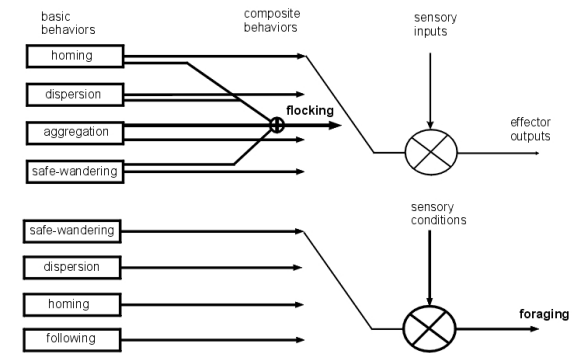
"Avoid" suppresses "Follow" and "Cruise" when the near-infrared sensors detect an imminent collision and "Escape" also helps to avoid obstacles.

The highest-level behaviour, "Detect-Sound-Pattern" caused the robot to play a tune.

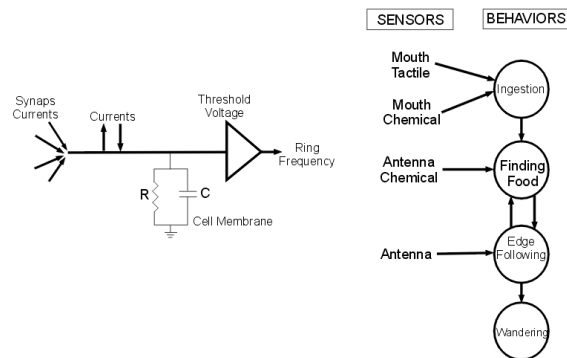
## Die Subsumtions-Architektur



## Foraging and Flocking



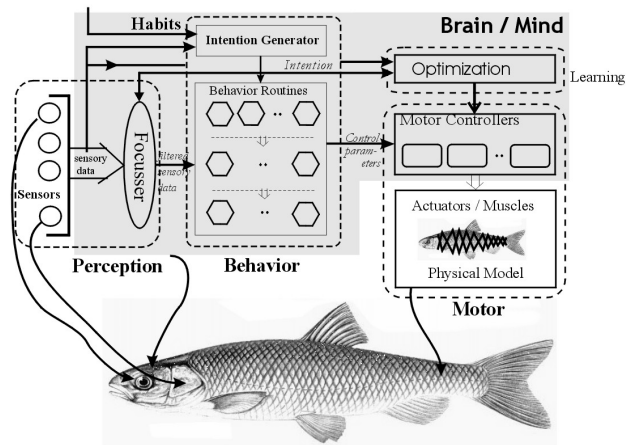
## Cockroach Neuron / Behaviors



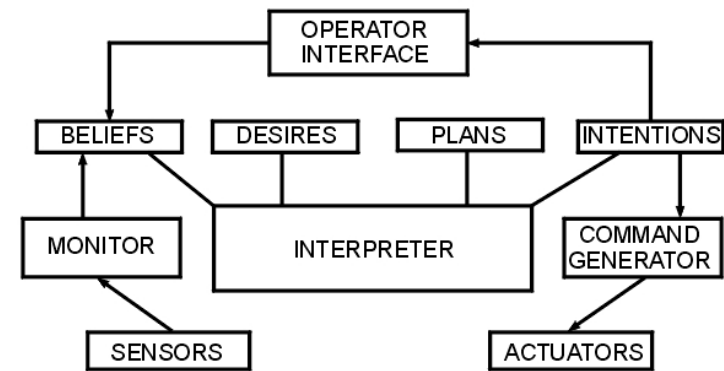
## Steuerungsarchitektur eines Fisches

- (E.)  
 Control and information flow in artificial fish:  
 Perception: sensors, focusser, filter  
 Behaviours: behaviour routines  
 Brain/mind: habits, intention generator  
 Learning: optimization  
 Motor: motor controllers, actuators/muscles

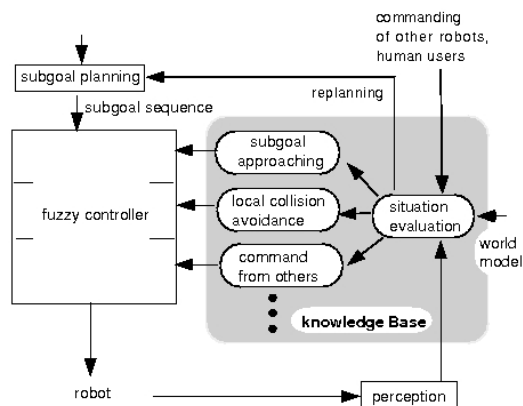
## Steuerungsarchitektur eines Fisches



## Procedural Reasoning System - SRI 1987



## Hierarchische Fuzzy-Regelung eines Roboters



## Hierarchische Fuzzy-Regelung eines Roboters

Für die Situationsbewertung werden Fuzzy-Regeln benutzt.

Durch die **Situationsbewertung** werden 3 Fuzzy-Parameter bestimmt:

- ▶ die **Priorität  $K$**
- ▶ den **Replanning-Selector**
- ▶ und **NextSubgoal**, ob ein Unterziel vorbei ist.

Eine typische Regel: IF ( $SL85$  IS HIGH) AND ( $SL45$  IS VL) AND ( $SLR0$  IS VL) AND ( $SR45$  IS VL)

AND ( $SR85$  IS VL) THEN  $K$  IS HIGH AND *Replan* IS LOW

**Koordinierung von mehreren Regelbasen:**

$$Speed = Speed_{LCA} \cdot K + Speed_{SA} \cdot (1 - K)$$

$$Steer = Steer_{LCA} \cdot K + Steer_{SA} \cdot (1 - K)$$

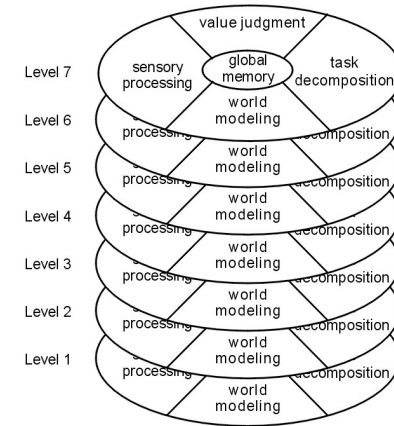
# Hierarchie

(Englisch, J. Albus)

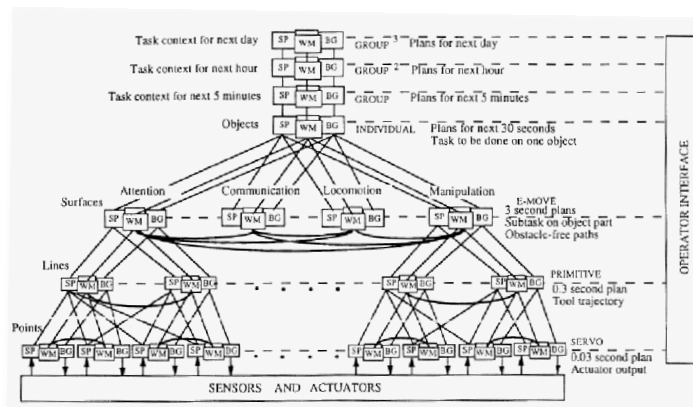
A RCS (real-time Control System) reference model architecture for intelligent system.

Processing modes are organised such that the BG (Behaviour Generation) modules form a command tree.

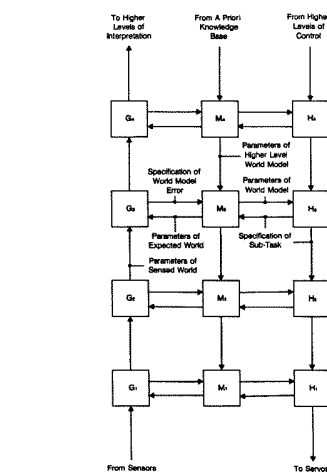
Information in the knowledge database is shared between WM (World Model) modules in nodes within the same subtree. On the right, are examples of functional characteristics of the BG modules and stored by the WM in the knowledge database at each level.



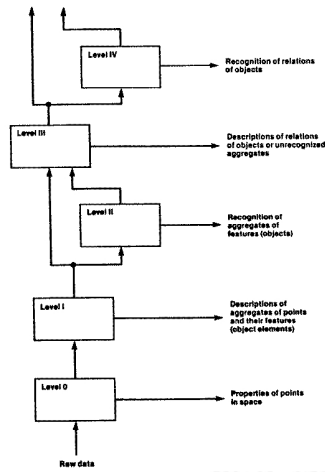
# Hierarchie



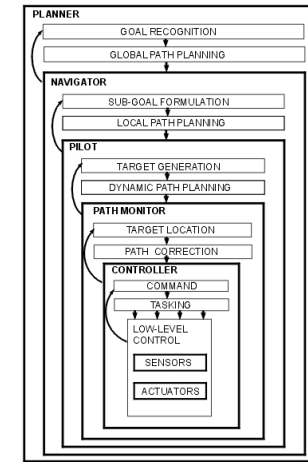
# Hierarchie



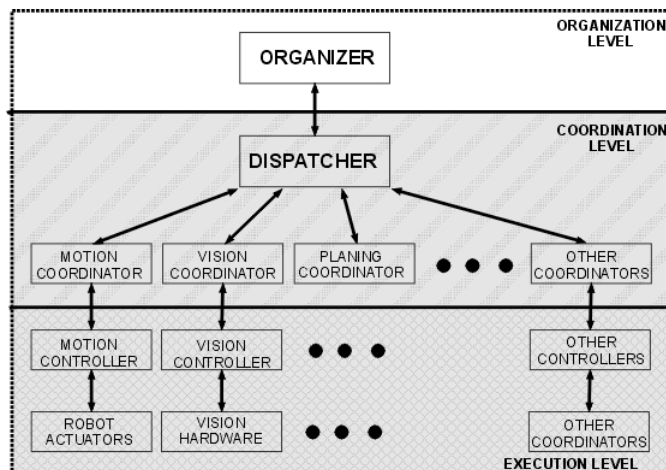
## Sensor-Hierarchie



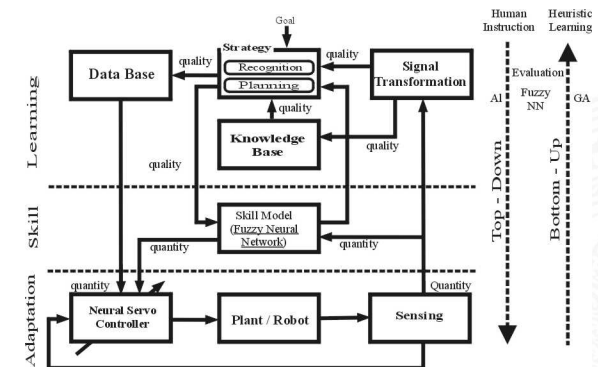
## Ein anderes Beispiel - Meystel



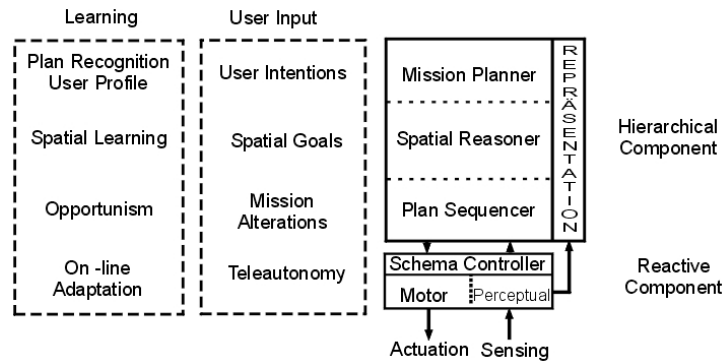
## Ein anderes Beispiel - Saridis



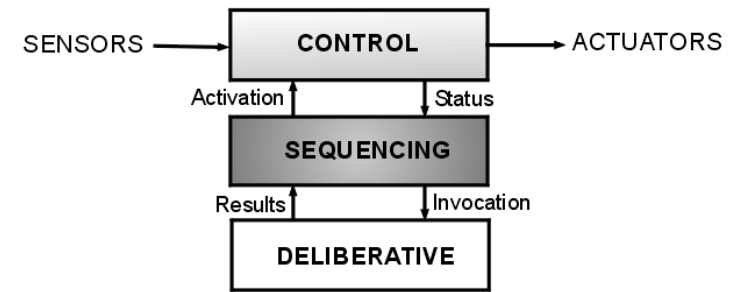
## Eine Architektur für lernende Roboter



## Das AuRA-Modell - Arkin '86



## Atlantis – Nasa Rovers 1991



## Gliederung

- Allgemeine Informationen
- Einführung
- Koordinaten eines Manipulators
- Kinematik-Gleichungen
- Inverse Kinematik von Manipulatoren
- Differentielle Bewegungen mit homogenen Transformationen
- Jacobi-Matrix eines Manipulators
- Aufgabenbeschreibung
- Robotergrammierung auf drei Ebenen
- Trajektoriegenerierung
- Trajektoriegenerierung
- Einführung in RCCL
- Dynamik

## Gliederung (cont.)

- Roboterregelung
- Programmierung auf Aufgabenebene und Bahnplanung
- Programmierung auf Aufgabenebene und Bahnplanung
- Programmierung auf Aufgabenebene und Bahnplanung
- Architekturen sensorbasierter intelligenter Systeme
- Aus- und Rückblick



## Thematische Zusammenfassung – Regelung

- ▶ Industrieroboter:
  - ▶ Positionregler mit PID-Reglern;
  - ▶ mit Schwerkraftkompensation
- ▶ Forschung:
  - ▶ modellbasierte Regelung
  - ▶ hybride Position-Kraft-Regelung
  - ▶ “under-actuated control”
  - ▶ “backwards controllable” (“direct drive”, “artificial muscle”) structure
  - ▶ Externesensorbasierte Regelung

## Thematische Zusammenfassung – Regelung

Adept-Roboter: direct drive



## Thematische Zusammenfassung – Mechanische Strukturen von Roboter

behandelt:

- ▶ Offene Kette mit rotatorischen Gelenken
- ▶ Hybride von rotatorischen und translatorischen Gelenken (SCARA, Stanford)

weitere:

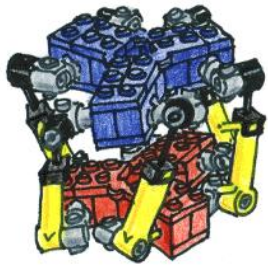
- ▶ mobile Roboter, Laufmaschine
- ▶ Geschlossene Kette u.a. Stewart Mechanismus (Craig 279)
- ▶ Antrieb ohne Motoren

## Die Stewart-Plattform

eine Plattform, die auf dem Boden steht, und eine, die mit insgesamt 6 variablen Beinen (in der Regel Hydrauliken) mit der unteren verbunden ist.

Die Befestigungen der Beine an beiden Plattformen geschieht mittels Kardangelenken. Mathematisch entspricht das einem sechs-dimensionalen Konfigurationsraum ohne Singularitäten. Der Vorteil einer parallelen Mechanik ist die große Nutzlast, die diese Systeme tragen können. Ein sequentieller Greifarm kann die auftretenden Drehmomente nicht bewältigen.

## Die Stewart-Plattform



## Thematische Zusammenfassung – Algorithmen

- ▶ Transformation
- ▶ Trajektoriengenerierung (z.B. Gerade Kartesische Bahn)
- ▶ Approximierte Darstellung von Roboterglidern und Objekten
- ▶ Graph-Bildung (V-Graph, T-Graph, ...)
- ▶ Suche-Algorithmen
- ▶ Andere Bahnplanungsalgorithmen
- ▶ Sensorfusion
- ▶ Vision: Erkennung (statisch, dynamisch), Rekonstruktion von Position und Lage
- ▶ Aktionsplanung
- ▶ Sensorgeführte Bewegung

## Gesamt-Rückblick - I

- ▶ **Einleitung:**  
Definition, Klassifikation, Grundkomponenten, DOF
- ▶ **Koordinaten-Transformation:**  
Manipulator-Koordinaten, homogene Transformationen, Drehmatrizen, ihre Inverse und ihre Verknüpfung, Transformations-Gleichungen (s. Paul81, Craig89, FGL87)
- ▶ **Kinematik:**  
Problematik der Kinematik und der inversen Kinematik, DH-Konvention und ihre Anwendungen, die algebraische und geometrische Lösung der inversen Kinematik, differentielle homogene Transformationen, Jacobi-Matrizen, Singuläritäten (s. Paul81, Craig89, FGL87)

## Gesamt-Rückblick - I

- ▶ **Trajektoriengenerierung:**  
Aufgabe und Randbedingungen, Polynom-Lösungen zwischen zwei Punkten sowie vier Punkten, Faktoren der optimalen Bewegung, Gerade Bewegungen im Kartesischen Raum: Realisierung und Probleme, Konzept der B-Spline-Interpolation, B-Spline Basisfunktionen. (s. Craig89, FGL87, Literatur über B-Splines)
- ▶ **Programmierung:**  
Aufgabenbeschreibung: Schritte von der Frame-Definition zur Erstellung von Bewegungsprogrammen, Vorteile und Konzepte von RCCL (s. Paul81, RCCL-Guide), Typen der Roboterprogrammierung, Off-line Programmierung (IRIP-Unterlagen, Craig89)
- ▶ **Regelung:**  
Regelungssysteme des PUMA-Roboters, lineare und Modell-basierte Regelung, PID-Regler, Regelungskonzepte im Kartesischen Raum (s. Craig89, FGL87).

## Gesamt-Rückblick - I

- ▶ **Sensoren:**  
Klassifikation, Interne Sensoren: Prinzip und Anwendung in der Regelung (s. Craig87, FGL87)
- ▶ **Dynamik:**  
Probleme, Newton-Euler'sche Gleichungen und Lagrange'sche Gleichungen (Lösung für Arme mit 1-2 Gelenken) Struktur einer dynamischen Gleichung (s. Paul81, FGL87)
- ▶ **Kollisionsvermeidung:**  
Begriffe um Konfigurationsraum, Konzepte der Transformation zum Konfigurationsraum, Objekt-Repräsentationen, Potentialfeld-Methoden, Probabilistische Ansätze
- ▶ **Steuerungsarchitektur:**  
Subsumtion, CMAC, Hierarchische,

## Ergänzte Literatur

- SiBo96** H.-J. Siegert und S. Bocionek. *Robotik: Programmierung intelligenter Roboter*, Springer-Verlag, 1996.
- Schilling90** R. J. Schilling. *Fundamentals of Robotics: Analysis and Control*, Printice Hall, 1990.
- Yoshikawa90** T. Yoshikawa. *Foundations of Robotics*, The MIT Press, 1990.
- SpVi89** M. W. Spong and M. Vidyasagar. *Robot Dynamics and Control*, Wiley, 1989.

## Intelligente Robotersysteme

Zugrundliegende Roboter-Technik (wie oben) plus:

- ▶ **Externe Wahrnehmung:**  
zuverlässige Umweltmessungen, Szenen-Verstehen
- ▶ **Wissensbasis:**  
über Umwelt, eigene Zustände, alltägliche Wissen wie von einem Mensch
- ▶ **Automatische Planung:**  
Aktion, Grobbewegung, Greifen und Sensordatenerfassung

## Intelligente Robotersysteme

- ▶ **Menschfreundliche Schnittstelle:**  
Verstehen der natürlchsprachlichen Anweisungen, Umsetzung zu Roboteraktionen, Disambiguität in Kontext-abhängigen Situationen
- ▶ **Adaptive Steuerung:**  
Evolution anstatt Programmierung, Lernfähigkeiten

Interessierte → Veranstaltungen im kommenden Semester:  
Projekt – Service Roboter  
Vorlesung – Angewandte Sensorik  
Vorlesung – Machinelles Lernen

## Automatische Planungssysteme

- ▶ **Aktionsplanung:**  
Aufgaben-Spezifikation, Zustandsdarstellung,  
Aufgaben-Zerlegung, Aktions-Reihenfolge-Generierung
- ▶ **Bewegungsplanung:**  
Roboter- und Umwelt-Repräsentation,  
Konfigurationsraum-Berechnung und -Darstellung,  
Suche-Verfahren
- ▶ **Sensorplanung:**  
Welche Sensoren, welche Zeitmomente, wo zu messen; interne  
sowie externe Parameter des Sensors

## Sensorgeführte Bewegung

- ▶ **Ziel:** Intelligente Regelung mit Fähigkeiten, an variierende  
Situationen anzupassen und auf Unsicherheiten zu reagieren
- ▶ **Steuerungsarchitektur:** Integration von Wahrnehmung,  
Planung und Aktion
- ▶ **Aufgaben der Sensordatenverarbeitung:**  
Lagebestimmung, Größe-Bestätigung, Annäherung-Detektion,  
Rutsche-Entdeckung, Erfolgs-Bestätigung, Fehler-Detektion,  
Inspektion

## Sensorgeführte Bewegung

- ▶ **Eingesetzte Sensoren:**  
Taktile Bilder, Sichtsysteme, Kraft-Moment-Meßsysteme,  
Distanz-Sensoren
- ▶ **Strategien:** *kalibrierte:* anhand von absoluten Informationen;  
*unkalibriert:* anhand von relativen Informationen
- ▶ **Arten der Wahrnehmung:** *passiv:* basiert auf bestimmter  
Sensor-Aktor-Konfigurierung;  
*aktiv:* abhängig von der Sensorplanung

## Kommerzielle Roboter in der Zukunft

werden:

- ▶ geschickter,
- ▶ kleiner,
- ▶ schneller,
- ▶ leichter,
- ▶ kräftiger,
- ▶ intelligenter,
- ▶ leichter zu bedienen,
- ▶ und billiger

sein.



## Große Herausforderungen an Robotik

### Verfahren:

- ▶ Symbolisches Verstehen der Umwelt
- ▶ Integrierte sensomotorische Koppelung
- ▶ Selbst-Lernen

### Systeme:

- ▶ Synergetische Multisensorik
- ▶ Agile Mobilität
- ▶ Geschickte Handhabungsfähigkeit



## Große Herausforderungen an Robotik

### Technik:

- ▶ Sensorik ähnlicher Komplexität wie die menschliche
- ▶ Neue Antriebsarten
- ▶ Nano-Roboter
- ▶ Multifinger-Hand
- ▶ Anthropomorphischer Roboter
- ▶ Fliegender Roboter