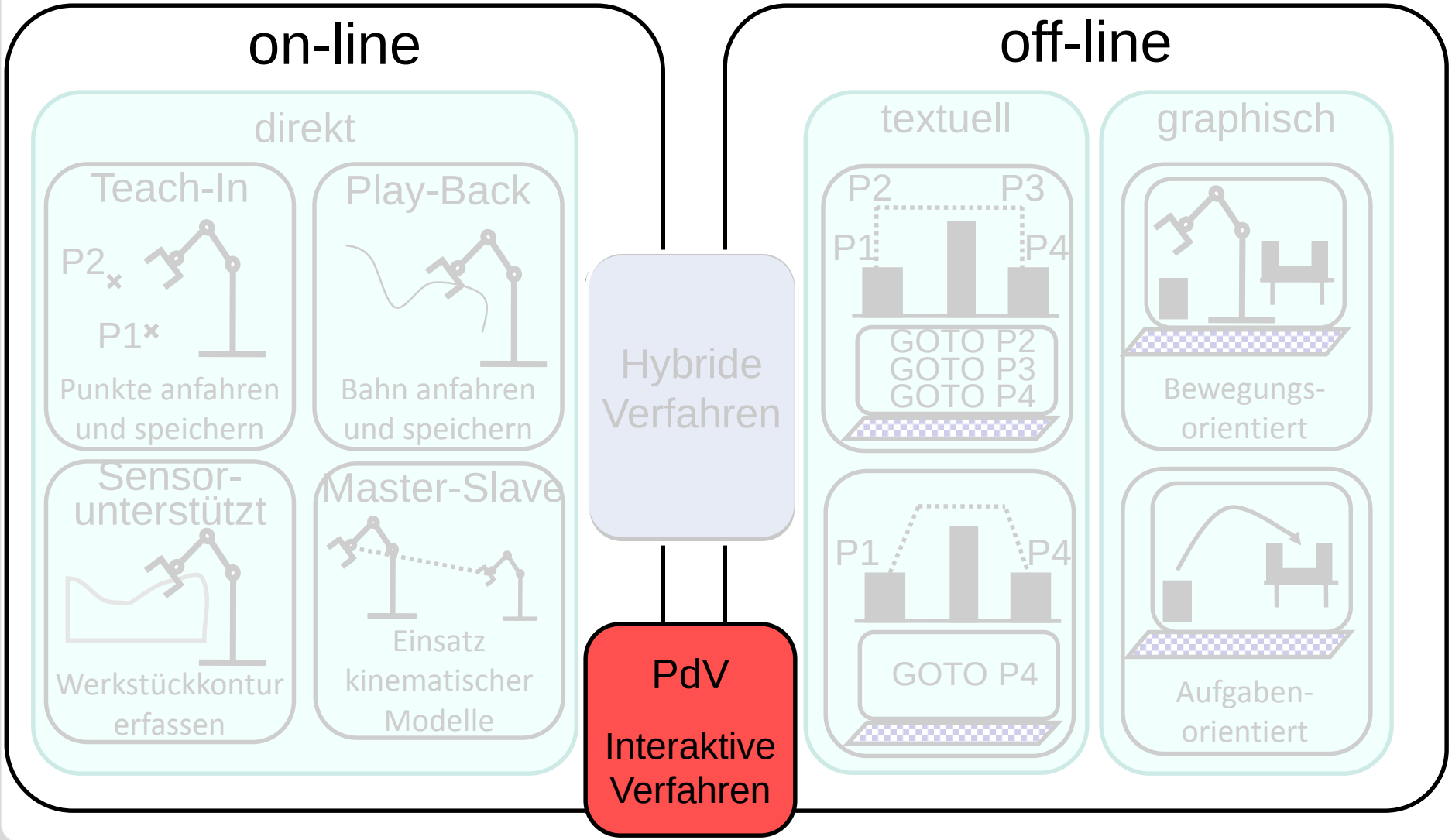
Karlsruhe Institute of Technology

Robotik I: Einführung in die Robotik
Programmieren durch Vormachen

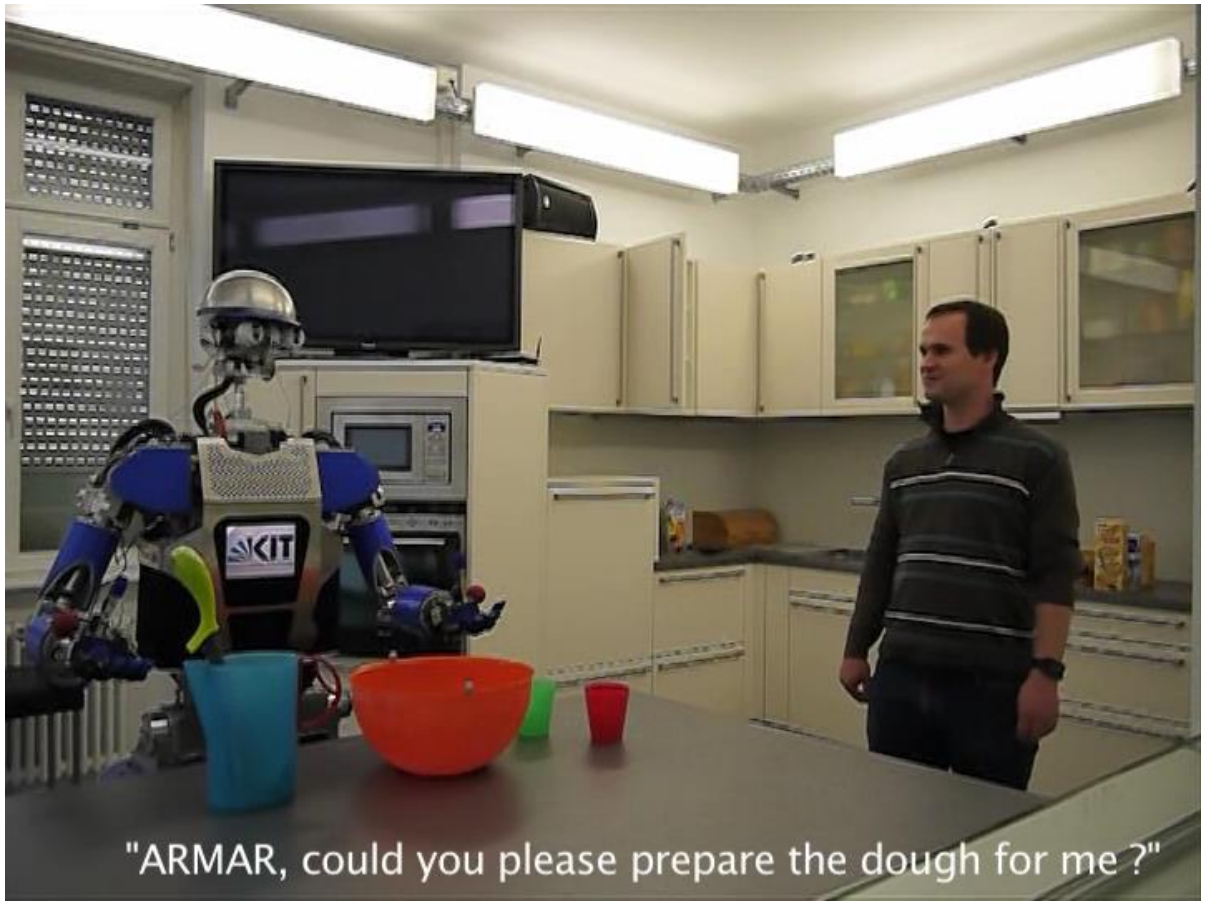
Tamim Asfour

KIT-Fakultät für Informatik, Institut für Anthropomatik und Robotik (IAR)
Hochperformante Humanoide Technologien (H^2T)

Interaktive Programmierverfahren

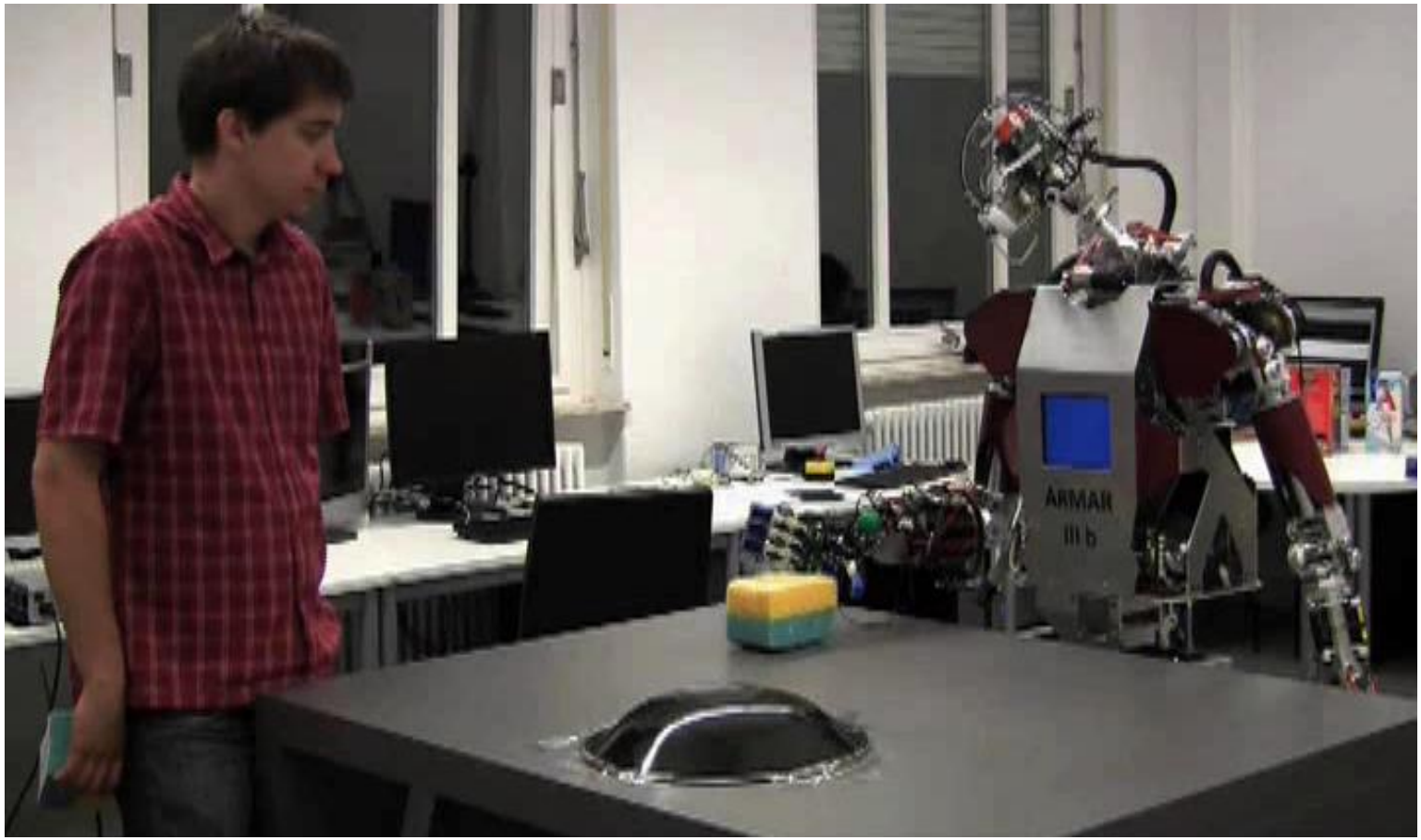


Lernen aus Beobachtung des Menschen



Wächter & Asfour (2015)

Lernen aus Beobachtung des Menschen



Inhalt

- Übersicht
- Kernfragen von Programmieren durch Vormachen
- **Perzeption:** Arten der Demonstration
  - Kinästhetisches Lernen, Teleoperation, Shadowing, Beobachtung
- **Kognition:** Lernen und Repräsentation von Fähigkeiten und Aufgaben
  - Segmentierung von Demonstrationen
  - Repräsentationsformen
  - Generalisierung
  - Verfeinerung
- **Aktion:** Ausführung von gelernten Fähigkeiten und Aufgaben
  - Ausführung von Aktionen mit online Adaption
  - Ausführung von komplexen Tasks in dynamischen Umgebungen

Programmieren durch Vormachen (PdV)

- Hauptziel von Programmieren durch Vormachen
  - **Intuitive** Roboterprogrammierung **ohne Expertenwissen**
  - Lernen von Aufgaben durch Generalisierung auf Basis mehrerer Observationen

- PdV impliziert Lernen und Generalisierung und ist deshalb kein Playback Verfahren.

- Das Thema ist auch bekannt als
  - Programming by Demonstration (PbD)
  - Learning from/by demonstration
  - Learning from human demonstration (LfD)
  - Apprenticeship learning
  - Imitation learning

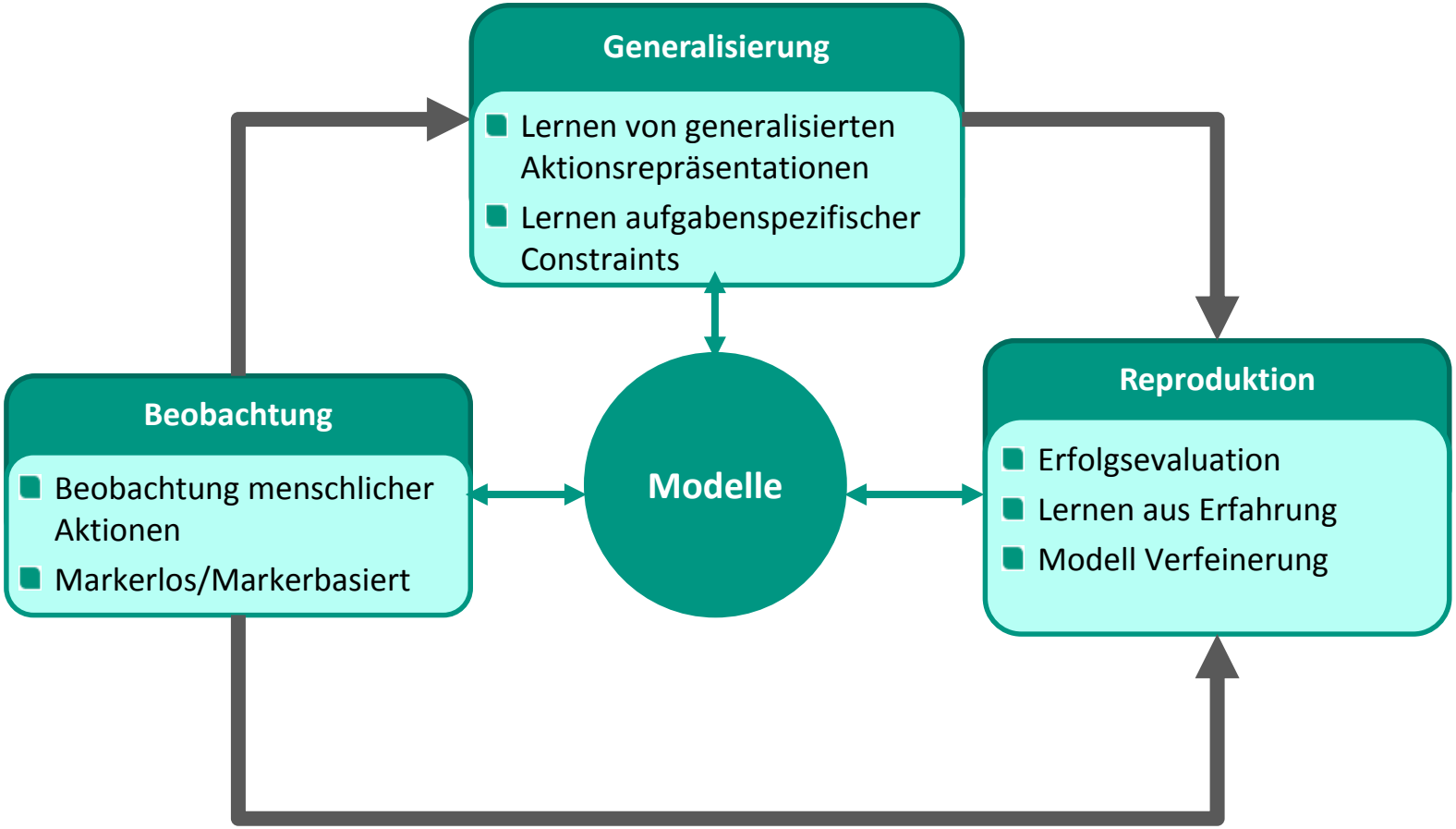
Literatur

A. Billard, S. Calinon, R. Dillmann and S. Schaal, Robot Programming by Demonstration, In *Handbook of Robotics*, Springer, pp. 1371-1394, 2008.

B. Argall, S. Chernova, M. Veloso and B. Browning, *A survey of robot learning from demonstration*, Robotics and Autonomous Systems, 2009.

J. Romero, *From human to robot grasping*, PhD Thesis, 2011

Lernen durch Beobachtung des Menschen



Module von Programmieren durch Vormachen

- Drei-Phasen-Modell
  - Zerlegung in Teilprobleme
  - Lösung in getrennten Modulen
  - Austausch von Modulen möglich

Drei Gründe für PdV

- Mächtiger Mechanismus zur **Komplexitätsreduktion** des Suchraums während des Lernens (im Gegensatz zum Ausprobieren aller Möglichkeiten).
  - Gute Beobachtungen können als Startpunkt gewählt werden
  - Schlechte Beobachtungen können direkt aus dem Suchraum eliminiert werden

- Impliziter Mechanismus zum **Trainieren von Robotern**, der das explizite und mühsame händische Programmieren durch Menschen reduziert oder sogar ganz eliminiert.

- Hilft beim Verständnis der Kopplung zwischen Perzeption und Aktion und dem Lernen relevanter Beziehungen zwischen diesen

Hauptherausforderungen in PdV

- **Wer soll imitiert werden?**
  - Lehrerauswahl: wähle einen Lehrer von dessen Verhalten der Imitierende am meisten profitiert.
  - Problem wird bei den meisten Methoden durch Auswahl eines dedizierten Lehrers vermieden.

Hauptherausforderungen in PdV

- Wer soll imitiert werden?
- **Wann soll imitiert werden?**
  - Der Imitierende muss die Bewegung **segmentieren** und Start und Ende des gezeigten Verhaltens ermitteln.
  - Der Imitierende muss entscheiden, ob es angemessen ist das beobachtete Verhalten im **aktuellen Kontext** auszuführen und wie oft es wiederholt werden muss.
  - Problem wird bei den meisten Methoden durch explizites Kennzeichnen von Start und Ende der Demonstration vermieden.

Hauptherausforderungen in PdV

- Wer soll imitiert werden?
- Wann soll imitiert werden?
- **Was soll imitiert werden?**
  - Wie findet man heraus **welche Aspekte der Demonstration von Interesse sind**?
    Bei Aktionen sind es die Bewegungen (Trajektorien) des Vorführenden.
    In anderen Situationen sind das Ergebnis und die Effekte von Aktionen wichtig.
  - Manche beobachtbaren **Eigenschaften sind irrelevant** und können **ignoriert** werden.
    Ist es zum Beispiel wichtig, dass der Vorführende immer von Norden her zum Tisch läuft?
  - Bei kontinuierlichen Regelungsaufgaben entspricht diese Aufgabe der Bestimmung des **Merkmalraums** für das Lernen, sowie der **Constraints** und der **Kostenfunktion.**
  - Bei diskreten Regelungsaufgaben, die z.B. mit Reinforcement Learning und symbolischem Reasoning behandelt werden, entspricht diese Aufgabe der **Definition** des **Zustands-** und **Aktionsraums** und wie dem automatische Lernen von **Vor- und Nachbedingungen (Constraints)**.

Hauptherausforderungen in PdV

- Wer soll imitiert werden?
- Wann soll imitiert werden?
- Was soll imitiert werden?
- **Wie soll imitiert werden?**
  - Bestimme, wie der Roboter das gelernte **Verhalten ausführen** wird, um die Metrik zu maximieren, die beim „*Was soll imitiert werden?*“ Problem gefunden wurde.
    - Auf Grund **unterschiedlicher Kinematik** kann z.B. ein Roboter menschliche Bewegungen nicht identisch reproduzieren.
    - Darf z.B. ein Roboter mit Rädern ein Objekt anfahren, wenn der Lehrer es mit dem Fuß angestoßen hat, oder muss er stattdessen einen Arm einsetzen?
  - Der Roboter muss eine **Abbildung** von Perzeption zu einer Sequenz eigener Bewegungen lernen. Deshalb bestimmen das **Embodiment** und die Körpereinschränkungen wie beobachtete Aktionen imitiert werden können (**Korrespondenzproblem**).

Hauptherausforderungen in PdV

- Wer soll imitiert werden?
- Wann soll imitiert werden?

Größtenteils unerforscht!

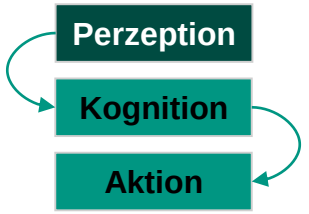
- Was soll imitiert werden?
- Wie soll imitiert werden?

Lernen einer Fähigkeit oder einer Aufgabe!

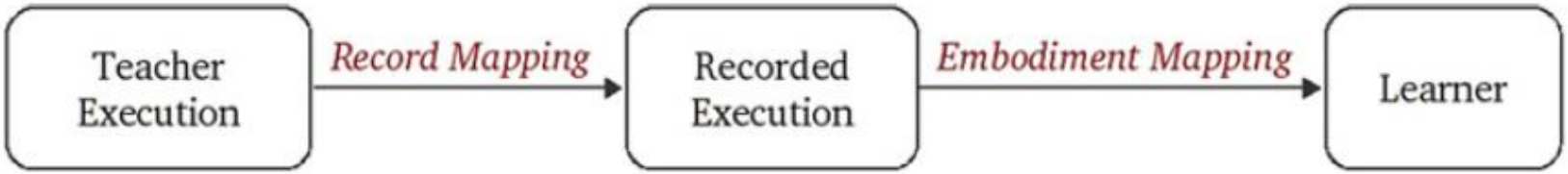
Inhalt

- Übersicht
- Kernfragen von Programmieren durch Vormachen
- **Perzeption:** Arten der Demonstration
  - Kinästhetisches Lernen, Teleoperation, Shadowing, Beobachtung
- **Kognition:** Lernen und Repräsentation von Fähigkeiten und Aufgaben
- **Aktion:** Ausführung von gelernten Fähigkeiten und Aufgaben

Aufnahme der Demonstration



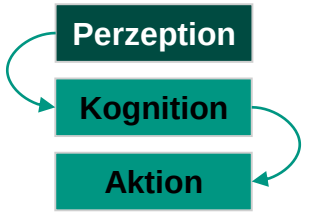
- Beim Lernen: Übertragen der aufgenommenen Aktion vom Lehrer auf den Roboter
- Zwei Schritte:
  - Aktionen des Lehrers aufnehmen (Record Mapping)
  - Aktion auf den Roboter abbilden (Embodiment Mapping)
    - Direkte Abbildung oder Abbildungsvorschrift notwendig



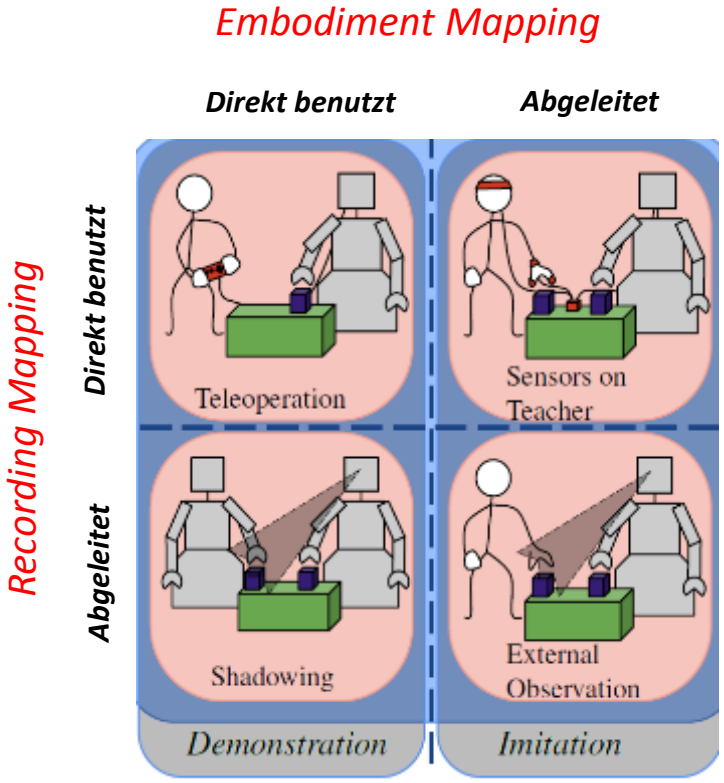
[Argall et al. 2009]

Abbildung der Ausführung des Lehrers auf den Roboter

Aufnahme der Demonstration

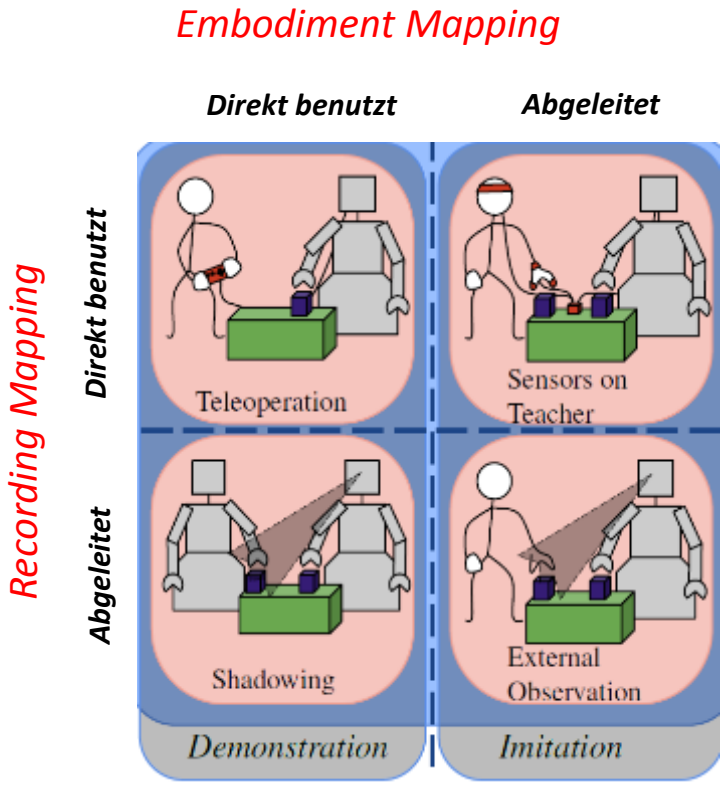


- Beim Lernen: Übertragen der aufgenommenen Aktion vom Lehrer auf den Roboter
- **Zwei Schritte:**
  - Aktionen des Lehrers aufnehmen
  - Aktion auf den Roboter abbilden
    - Direkte Abbildung oder Abbildungsvorschrift notwendig
- Beide Schritte können direkt oder indirekt durchgeführt werden
- Es gibt also vier Typen von Lernen durch Vormachen:
  - Teleoperation
  - Shadowing
  - Sensoren am Lehrer
  - Externe Beobachtung



[Argall et al. 2009]
[Romero 2011]

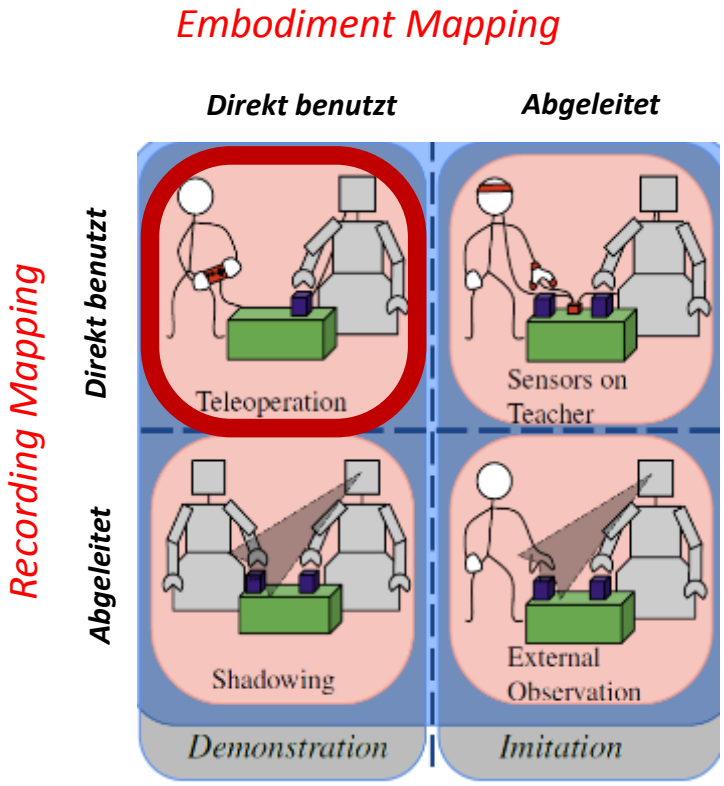
Schnittstellen zum Vormachen



[Argall et al. 2009]
[Romero 2011]

- *Erste Zeile*: Aufgenommene Daten können direkt vom Roboter genutzt werden
- *Zweite Zeile*: Benötigte Daten sind nur indirekt verfügbar und müssen aus den aufgenommen Daten abgeleitet werden
- *Linke Spalte*: Vorgemachte Bewegungen entsprechen dem Körper des Roboters und können direkt benutzt werden
- *Rechte Spalte*: Vorgemachte Bewegungen entsprechen nicht dem Körper des Roboters, weshalb eine Abbildung auf den Roboter notwendig ist.

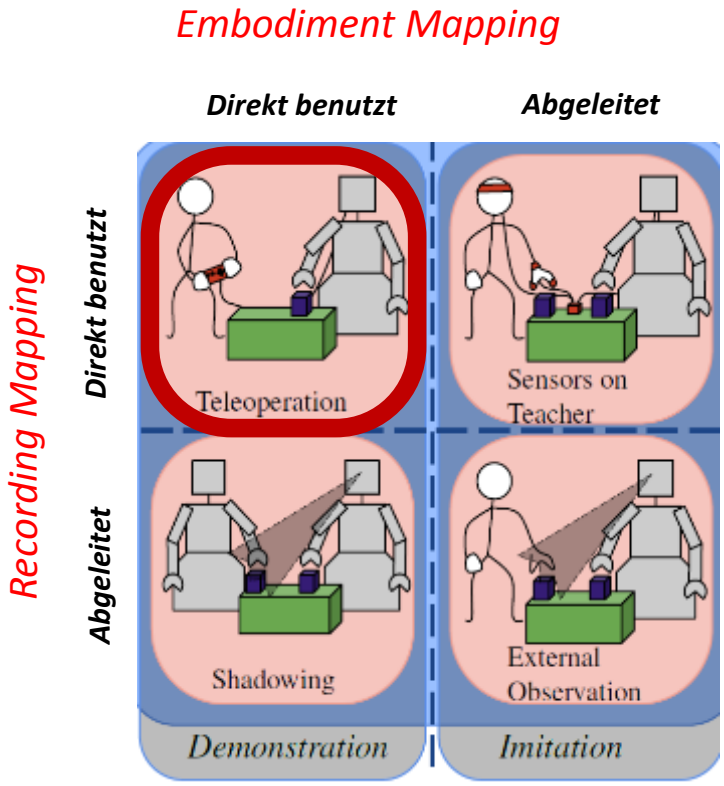
Schnittstellen zum Vormachen: Teleoperation



[Argall et al. 2009]
[Romero 2011]

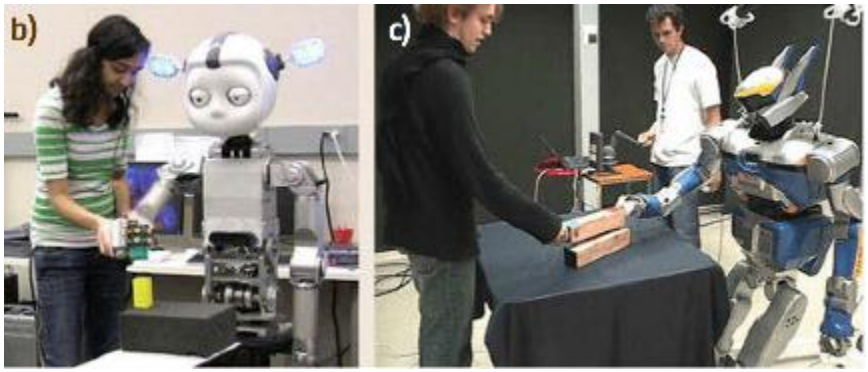
- Lehrer steuert den Roboter **direkt** mittels **Joysticks** oder anderen Remote-Steuerungsgeräten
- Die **Sensoren des Roboters** (z.B. Position, Kraft) nehmen die Demonstration auf
- Embodiments des Lehrers und Schülers sind die selben (der Roboter selbst)
  → Keine Nachbearbeitung nötig (**Kein Korrespondenzproblem**)
- Aber: Benutzung des Steuerungsgeräts benötigt oft Training, da oft 6+ Freiheitsgrade gesteuert werden

Teleoperation vs Kinästhetisches Lernen



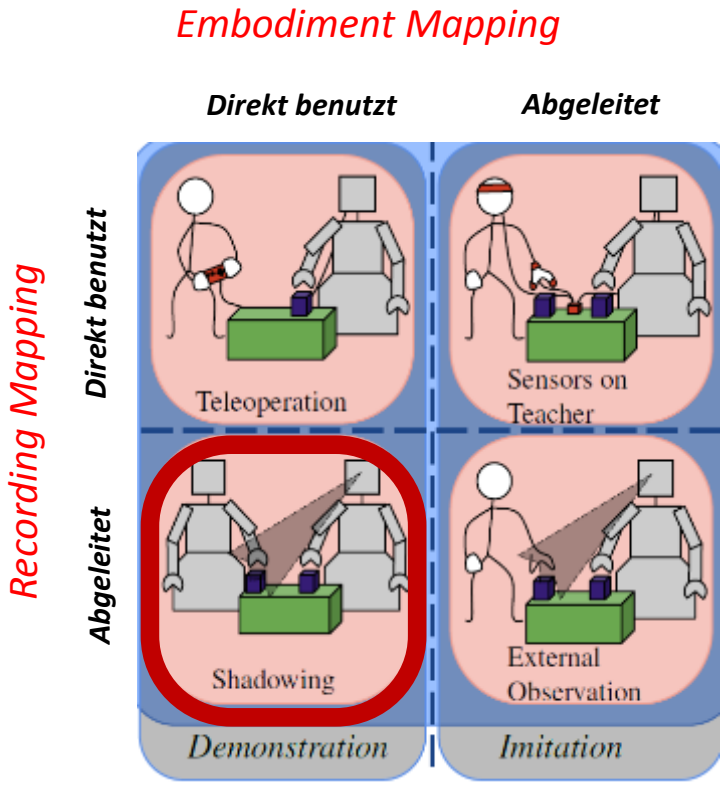
[Argall et al. 2009]
[Romero 2011]

- Bei kinästhetischem Lernen wird der Roboter direkt durch physikalische Interaktion bewegt
  - Setzt Compliance (aktiv/passiv) beim Roboter voraus
- Aufnahme der Demonstration wie bei Teleoperation
- Mensch braucht für gewöhnlich mehr DoF als der Roboter, um eine Aktion zu demonstrieren
  - Zweihändige Aktionen nur schwer möglich
- Mensch muss sich im Arbeitsbereich des Roboters befinden
  - Sicherheitsrisiko
  - Nicht von Entfernung möglich (z.B. Robonaut auf Weltraumstation)



Kinesthetic Teleoperation [Billard et al. 2008]

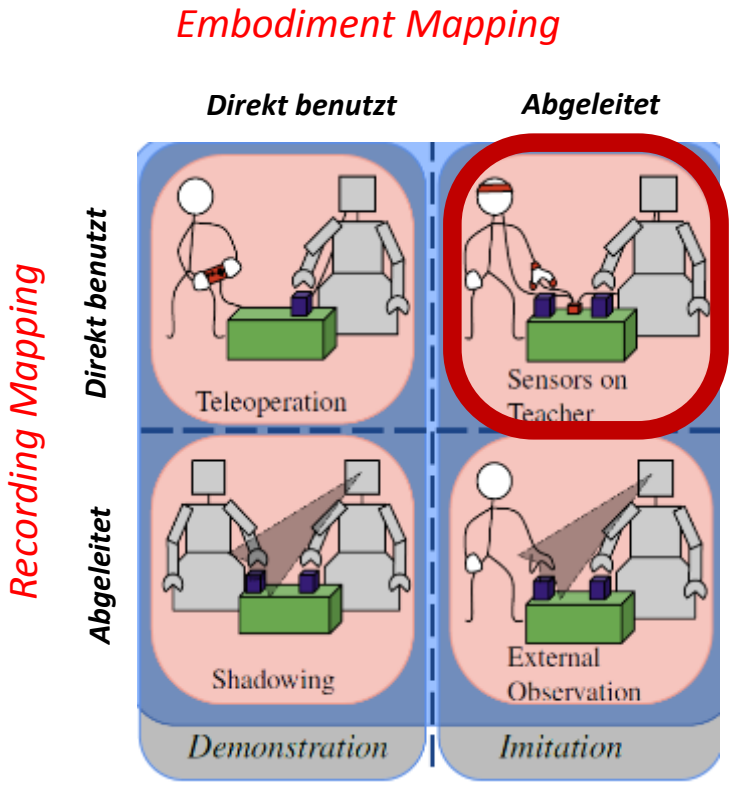
Schnittstellen zum Vormachen: Shadowing



[Argall et al. 2009]
[Romero 2011]

- Lehrer und Schüler haben gleiches Embodiment
- Kein direkter Zugriff auf die Daten des Lehrers
- Nachahmung der Aktionen des Lehrers (durch Beobachtung) und Aufnahme des Aktion mit eigenen Sensoren
- Daher ist die Aufnahme indirekt
- Aufnahme direkt im Embodiment des Schülers

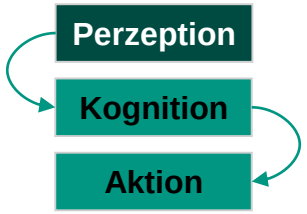
Schnittstellen zum Vormachen: Sensoren auf dem Lehrer



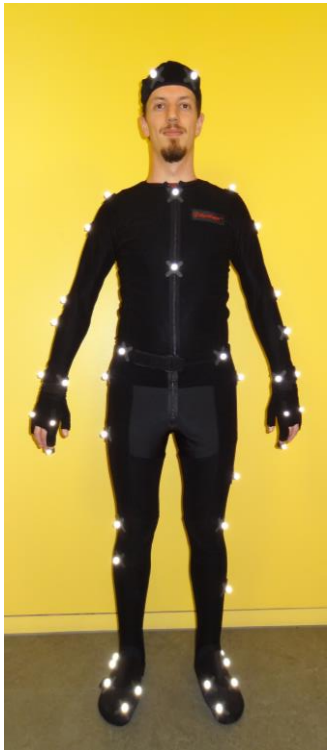
[Argall et al. 2009]
[Romero 2011]

- Bewegungen des Lehrers werden direkt aufgenommen und der Roboter hat direkten Zugriff → Explizite und direkte Aufnahme
- Hohe Genauigkeit der Aufnahmen
- Mensch kann sich frei bewegen
- Abbildung vom Menschen auf Roboter nötig (Korrespondenzproblem)

Erfassung der menschlichen Bewegung



- Die Erfassung der menschlichen Bewegung ist Ausgangspunkt für die Programmierung durch Vormachen
- **Marker-basierte Verfahren**:
  - Marker werden an vorher festgelegten anatomischen Landmarken des menschlichen Körpers angebracht
  - Marker sind entweder aktiv (z.B.: Codierung mit LED) oder passiv (z.B.: reflektierend)
  - Mensch muss zur Erfassung vorbereitet („vermarkert“) werden
- **Marker-lose Verfahren**:
  - Direkte Rekonstruktion der menschlichen Pose aus Kamera-Daten (RGB-/Tiefendaten)
  - Keine Vermarkerung des Menschen erforderlich

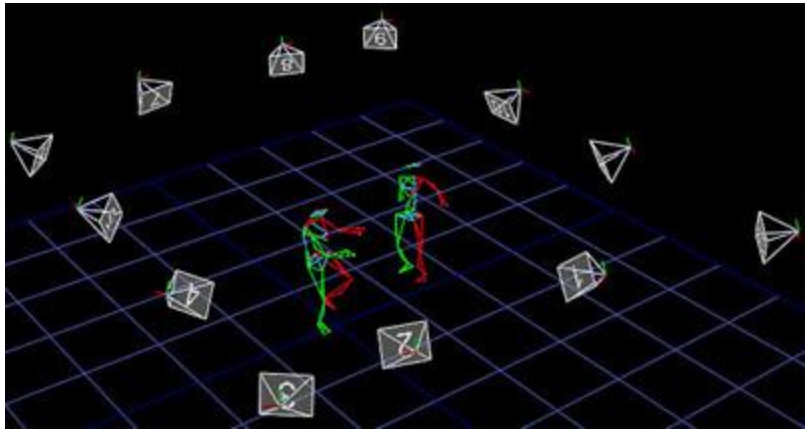


Fields et al. (2011): *Human Motion Capture Sensors and Analysis in Robotics*
Moeslund et al. (2006): *A Survey of Computer Vision-Based Human Motion Capture*

Marker-basierte optisch-passive Bewegungserfassung

- Lokalisierung von infrarot-reflektierenden passiven Markern mit Hilfe eines Mehr-Kamera-Systems
- Weit verbreitete Lösung, verschiedene kommerzielle Anbieter (z.B. VICON und OptiTrack)

- Vorteile:
  - Hohe räumliche Genauigkeit (Submillimeter-Bereich)
  - Hohe zeitliche Auflösung (1-2 kHz möglich)
- Nachteile:
  - Teuer und hoher technischer Aufwand
  - Aufwand für Nachbearbeitung durch Verdeckungen und Labeling der Marker
  - Anforderungen an Aufnahmebereich (z.B. nur indoor möglich)



Marker-basierte optisch-aktive Bewegungserfassung

- Marker übertragen eine codierte Kennung, z.B. mit Infrarot-LED

- Vor-/Nachteile ähnlich zu optisch-passiven Systemen, aber
  - vereinfache Handhabung des Marker Labelings
  - dafür: Batterien/Kabel zur Stromversorgung jedes Markers notwendig

© Qualisys

IMU-basierte Bewegungserfassung

- Anbringung von inertialen Messeinheiten (IMUs) an anatomisch definierten Punkten des menschlichen Probanden (z.B. 17 beim Xsens MVN)

- Vorteile:
  - Geringe Anforderungen an Aufnahme-Umgebung (z.B. outdoor)
  - Genaue Bestimmung der menschlichen Pose
- Nachteile:
  - Exakte Platzierung der IMUs erforderlich
  - Keine exakte Positionsbestimmung des Menschen im globalen Koordinatensystem möglich (IMU-Drift)

© Xsens

Mechanische Bewegungserfassung

- Direkte Messung von Gelenkwinkeln (z.B. Potentiometer)

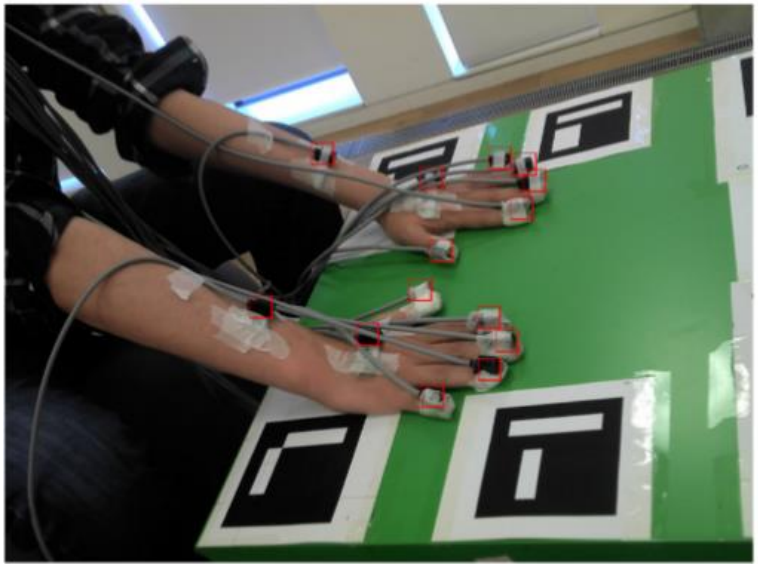
- Vorteile:
  - Geringe Anforderungen an Aufnahme-Umgebung (z.B. outdoor)
- Nachteile:
  - Einschränkung der menschlichen Bewegung durch Exoskelett → Siehe Vorlesung „Anziehbare Robotertechnologien“ im SS
  - Hoher technischer Aufwand
  - Keine Positionsbestimmung des Menschen im globalen Koordinatensystem möglich

© Sarcos

Magnetische/akustische Bewegungserfassung

- Lokalisierung von Markern mit elektromagnetischen oder akustischen (Ultraschall) Prinzipien

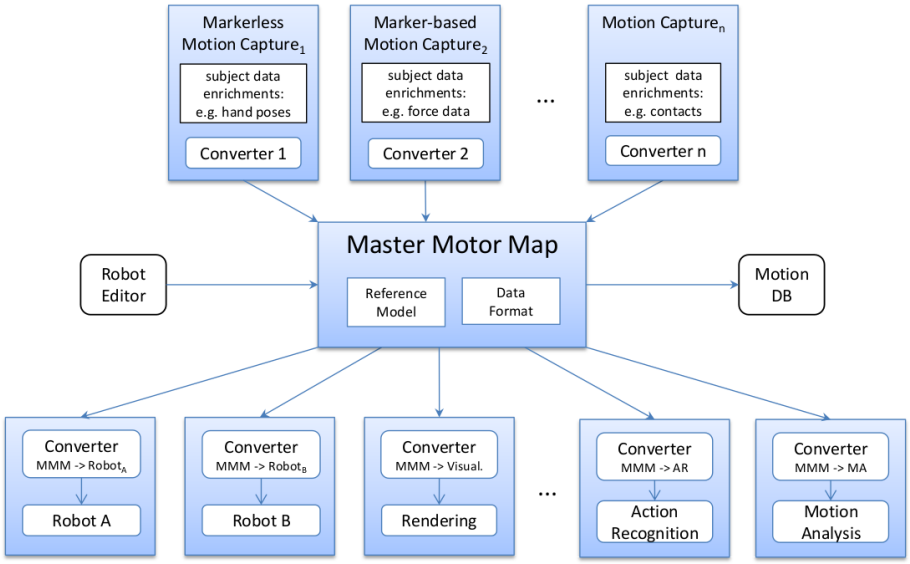
- Vorteile:
  - Günstiger als marker-basierte optischen Lösungen
  - Geringe Probleme mit Verdeckungen
- Nachteile:
  - Stark begrenzte Reichweite
  - Empfindlichkeit gegenüber Störungen (metallische Gegenstände bzw. andere Ultraschallquellen)
  - Teilweise hohe Ungenauigkeiten

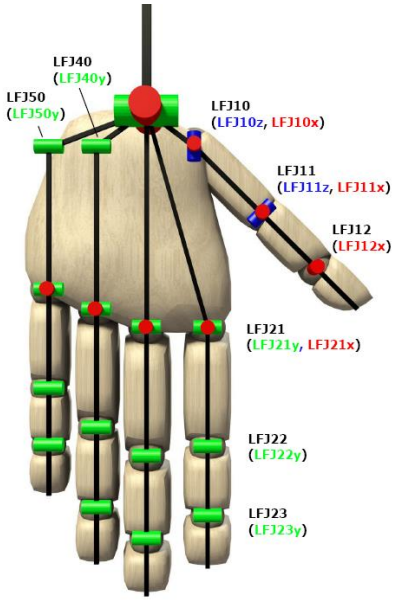


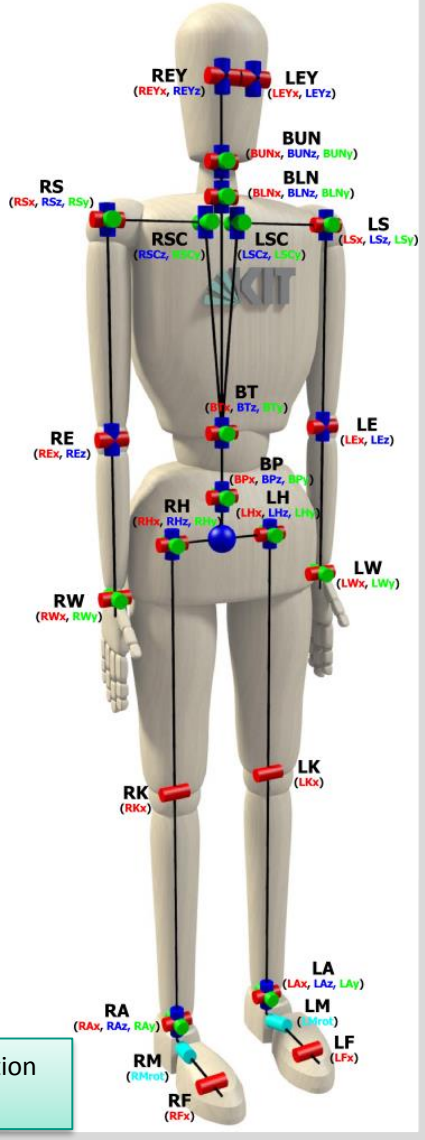
Sandilands et al. (2013)

Master Motor Map (MMM) Framework

- Framework zur vereinheitlichenden Darstellung menschlicher Ganzkörperbewegung
- Erfasste Bewegungen unterschiedlicher Menschen und Erfassungstechniken werden auf Referenzmodell des menschlichen Körpers mit 104 Bewegungsfreiheitsgraden abgebildet
- https://mmm.humanoids.kit.edu

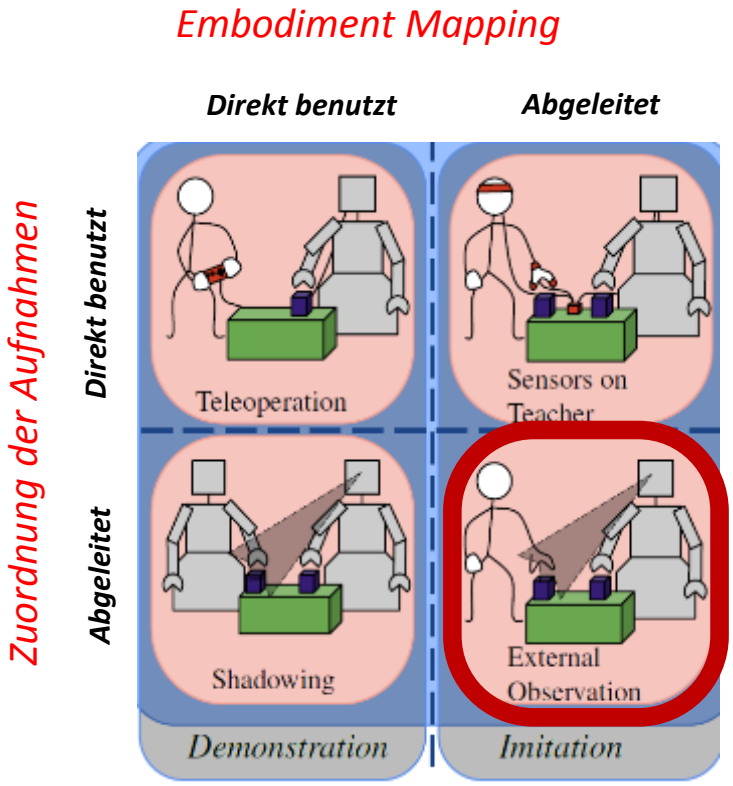






C. Mandery, Ö. Terlemez, M. Do, N. Vahrenkamp and T. Asfour, Unifying Representations and Large-Scale Whole-Body Motion Databases for Studying Human Motion, IEEE Transactions on Robotics, Vol. 32, No. 4, pp. 796 - 809, August, 2016

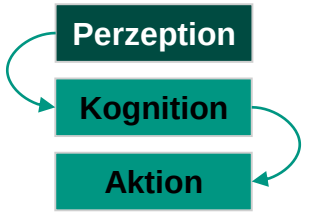
Schnittstellen zum Vormachen: Externe Beobachtung



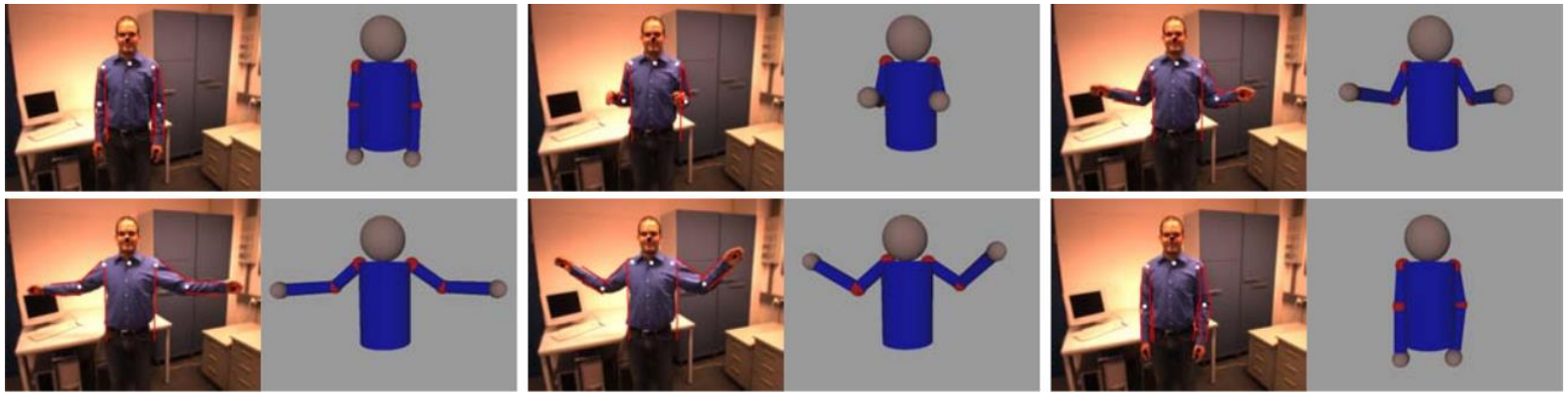
[Argall et al. 2009]
[Romero 2011]

- Beobachtung durch Sensoren des Roboters (Kameras)
- Komplexe Algorithmen nötig zur Bildverarbeitung
- Niedrige Präzision
- Haptik des Lehrers kann nicht aufgenommen werden
- Abbildung vom Menschen auf Roboter nötig (Korrespondenzproblem)

Marker-lose optische Bewegungserfassung



- Rekonstruktion der menschlichen Pose aus RGB- und/oder Tiefen-Daten
- Vorteile:
  - Sehr günstig, geringer Hardware-Aufwand
  - Geringe Anforderungen an Aufnahme-Umgebung
- Nachteile:
  - Komplexe Algorithmen und hohe Fehlerrate (aktives Forschungsgebiet)
  - Niedrige zeitliche Auflösung, vor allem bei Online-Einsatz



Azad et al. (2008)

Skeleton Tracking

- Tiefenkameras liefern Tiefenbilder und Skelettdaten mit 30 fps
- Skelettdaten werden aus jedem Einzelbild berechnet
- Genauigkeit zwischen RGB-Tracking und Marker-basiert

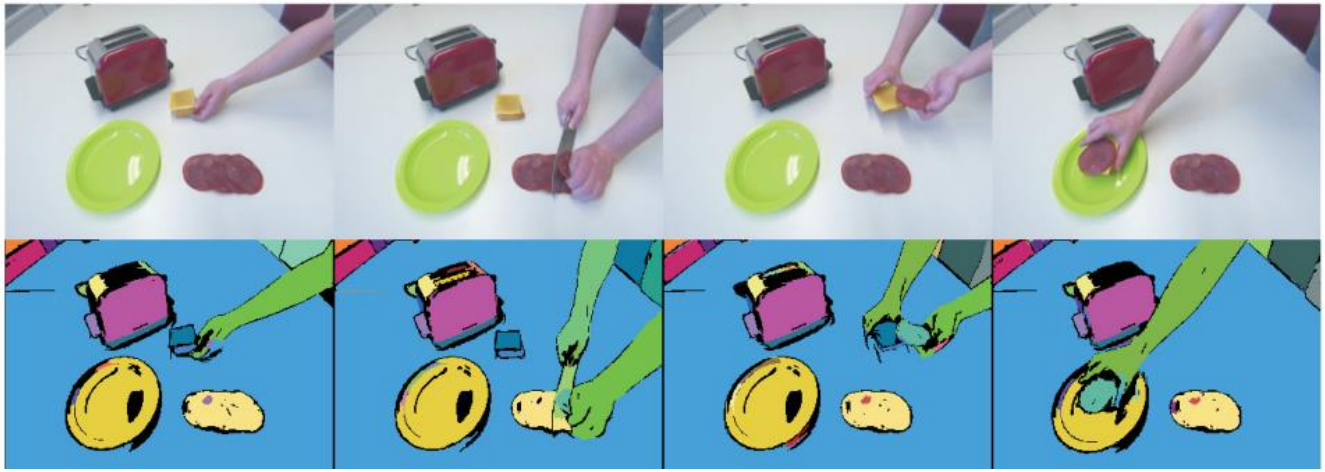
J. Shotton et al., *Real-Time Human Pose Recognition in Parts from a Single Depth Image, CVPR, 2011*

Segmentation & Tracking

- Tracking von Lehrer und Objekten durch Farb- und Tiefenregionen
- Optischer Fluss (Bewegungsmuster) wird benutzt, um Bewegungen zwischen 2 Frames zu tracken
- Segmentmittelpunkte pro Frame stellen die Trajektorie dar

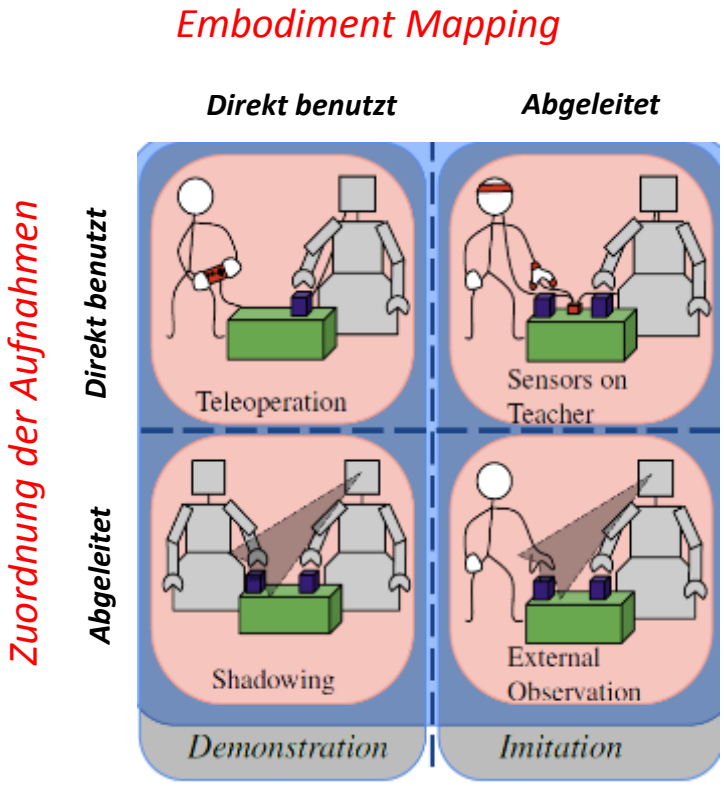
Kamera Bilder

Getrackte Bildregionen



A. Abramov, E.E. Aksoy, J. Dörr, K. Pauwels, F. Wörgötter, *Real-Time Human Pose Recognition in Parts from a Single Depth Image, 3DPVT, 2010*

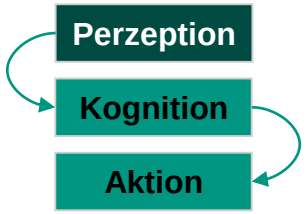
Interfaces for Demonstration



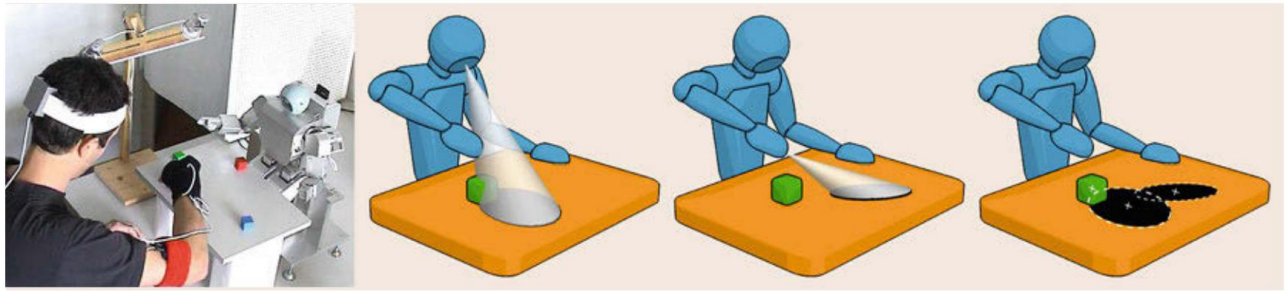
[Argall et al. 2009]
[Romero 2011]

- Der Begriff *Demonstration* wird benutzt für:
  - Teleoperation
  - Shadowing
- *Imitation* wird benutzt für
  - Sensoren auf dem Lehrer
  - Externe Beobachtung

Mensch-Roboter Interaktion für PdV



- Weitere Aufnahmekanäle für PdV: **Multimodale Interaktionen**
- Interaktion während des Lernvorgangs mit dem Menschen
  - Zusätzliche Modalitäten zur Verbesserung des Lernprozesses
    - Blickrichtung des Lehrers
    - Zeige-Gesten des Lehrers
    - Sprachliche Informationen: Ergänzende Informationen
      - Vokale Deixis: *Hier, Dort, Jetzt, …*
      - Hervorhebung von einzelnen Schritten
  - Roboter fragt bei Unklarheit
    - Repräsentation mit **Konfidenzmaß**
    - Roboter fragt gezielt nach Details
  - Mensch **weist** den Roboter gezielt auf schlecht ausgeführte **Aspekte** der Aktionen hin

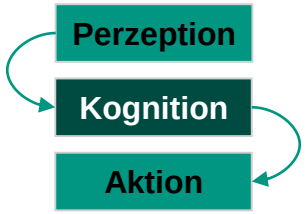


Calinon, 2006

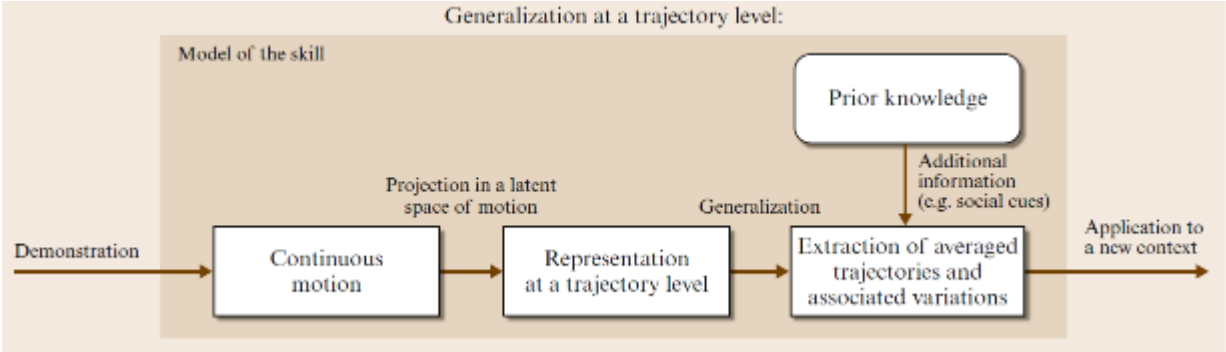
Inhalt

- Übersicht
- Kernfragen von Programmieren durch Vormachen
- **Perzeption:** Arten der Demonstration
- **Kognition:** Lernen und Repräsentation von Fähigkeiten und Aufgaben
  - Segmentierung von Demonstrationen
  - Repräsentationsformen
  - Generalisierung
  - Verfeinerung
- **Aktion:** Ausführung von gelernten Fähigkeiten und Aufgaben

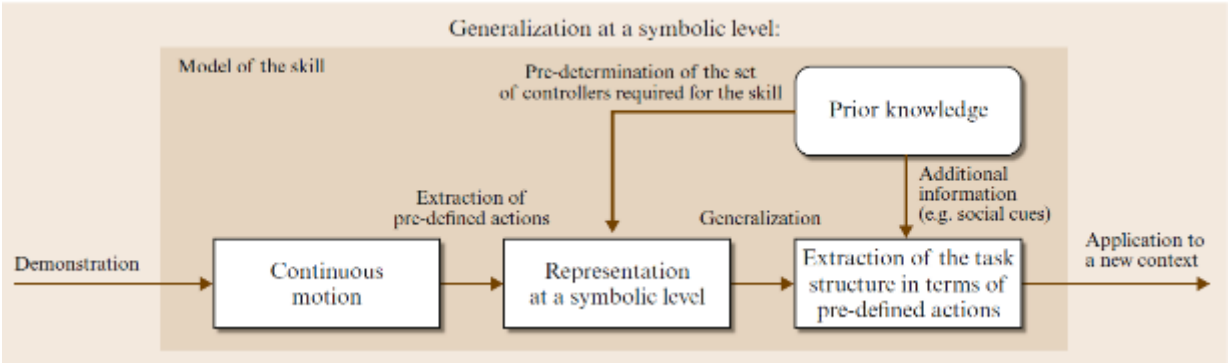
Lernen einer Fähigkeit (PdV)



- Eine Fähigkeit kann auf zwei Ebenen repräsentiert werden:
  - Trajektorie: Eine nicht lineare Zuordnung zwischen Sensor- und Motorinformation stellt eine Repräsentation auf niedriger Ebene dar



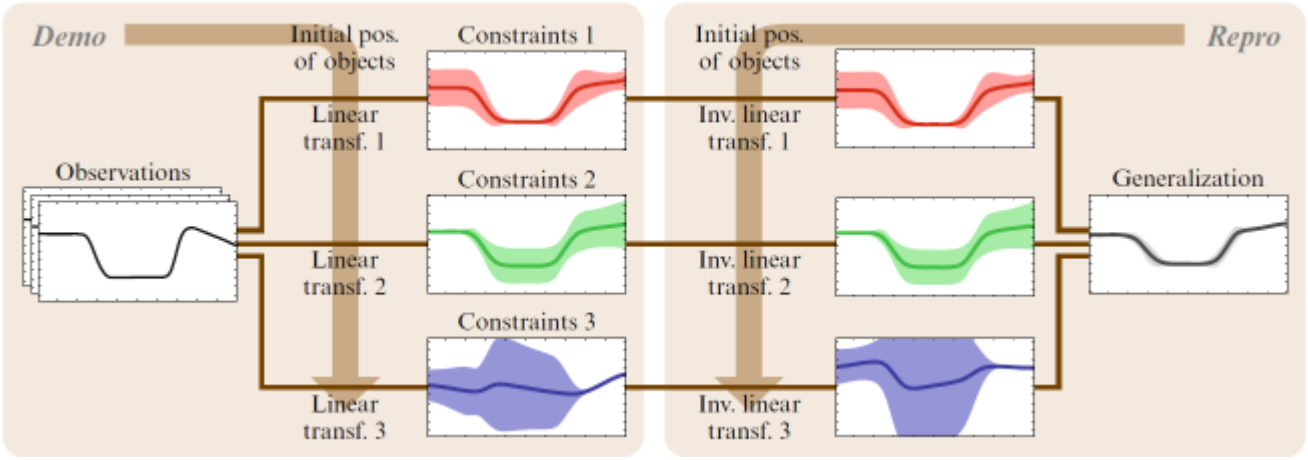
  - Symbolisch: Eine Zerlegung in eine Folge von Aktionen (*action perception units*) stellt eine Repräsentation auf hoher Ebene dar



Billard, A., Calinon, S. and Dillmann, R. (2016). Learning From Humans. Siciliano, B. and Khatib, O. (eds.). Handbook of Robotics, 2nd Edition, Chapter 74, pp. 1995-2014. Springer

Lernen einer Fähigkeit (PdV)

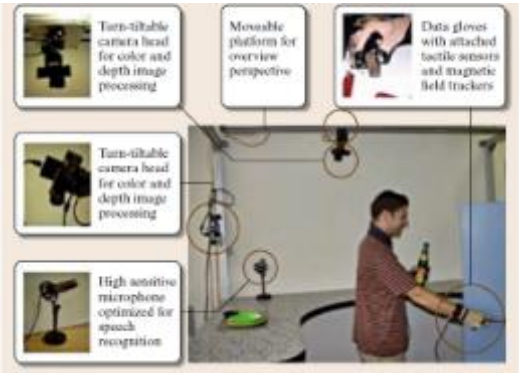
- Trajektorie: Verwendet Methoden zur **Dimensionsreduktion**, die das aufgenommene Signal in einen latenten Bewegungsraum mit niedrigerer Dimension projiziert
- Diese Methoden setzen lokale lineare Transformationen oder globale nicht-lineare Verfahren ein
- Die Kodierung kann spezifisch zu einer zyklischen (periodischen) Bewegung, zu einer diskreten Bewegung oder zu einer Kombination aus beiden sein

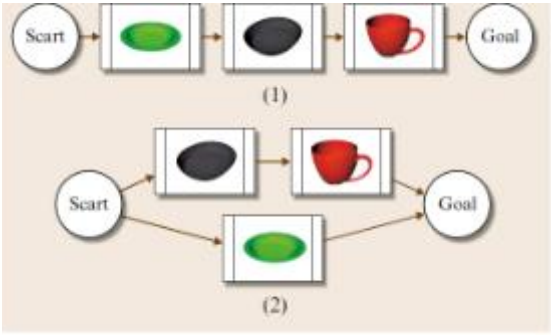


S. Calinon, A. Billard: What is the Teacher's Role in Robot Programming by Demonstration? – Toward Benchmarks for Improved Learning, Interact. Stud. 8(3), 441–464 (2007), Special Issue on Psychological Benchmarks in Human-Robot Interaction

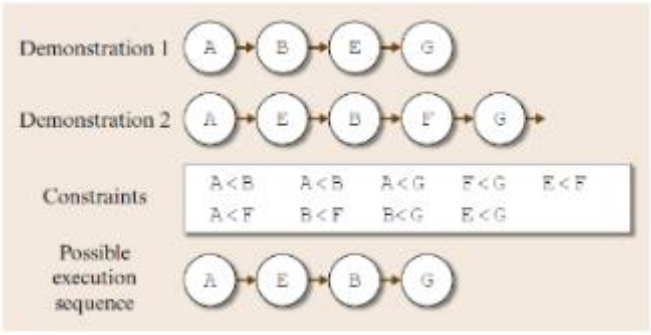
Lernen einer Fähigkeit (PdV)

- Symbolisch: Ein üblicher Ansatz segmentiert und kodiert die Aufgabe anhand von Folgen **vordefinierter** Aktionen, die symbolisch beschrieben sind.
- Das Kodieren und Reproduzieren der Aktionsfolgen kann mit klassischen Machine-Learning-Verfahren wie z. B. HMMs erfolgen.





M. Pardowitz, R. Zoellner, S. Knoop, R. Dillmann, Incremental Learning of Tasks from User Demonstrations, Past Experiences and Vocal Comments., IEEE Trans. Syst., Man Cybernet. 2007.



S. Ekvall, D. Kragic: Learning Task Models from Multiple Human Demonstrations, Proc. IEEE Int. Symposium on Robot and Human Interactive Communication (RO-MAN) 2006

Lernen einer Fähigkeit (PdV)

- Symbolisch: Ein üblicher Ansatz segmentiert und kodiert die Aufgabe anhand von Folgen **vordefinierter** Aktionen, die symbolisch beschrieben sind
- Das Kodieren und Reproduzieren der Aktionsfolgen kann mit klassischen Machine-Learning-Verfahren wie z. B. HMMs erfolgen

- Vorteil: Komplexe Fertigkeiten können effizient über einen interaktiven Prozess gelernt werden

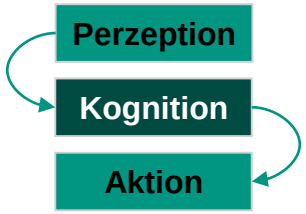
- Nachteil: Die Verfahren benötigen sehr viel Vorwissen, um die wichtigen Kennzeichen zur Segmentierung erkennen zu können

Lernen einer Fähigkeit (PdV)

- Die zwei Ebenen zur Beschreibung einer Fertigkeit:

Kodierung	Generalisierungsprozess	Vorteile	Nachteile
Trajektorie	Generalisierung von Bewegungen	Allgemeine Repräsentation von Bewegungen, die es erlaubt verschiedenste Arten von Signalen und Gesten zu kodieren	Kann komplexe Fertigkeiten nicht reproduzieren
Symbolisch	Sequentielle Organisation von vordefinierten Bewegungselementen	Ermöglichen es, eine Hierarchie zu lernen	Benötigt eine vordefinierte Menge von einfachen Controllern zur Reproduktion

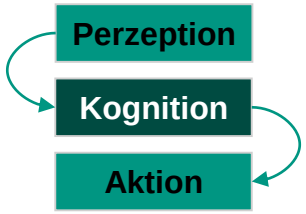
Roboterprogrammierung durch Vormachen (symbolisch)



Generelle Vorgehensweise:

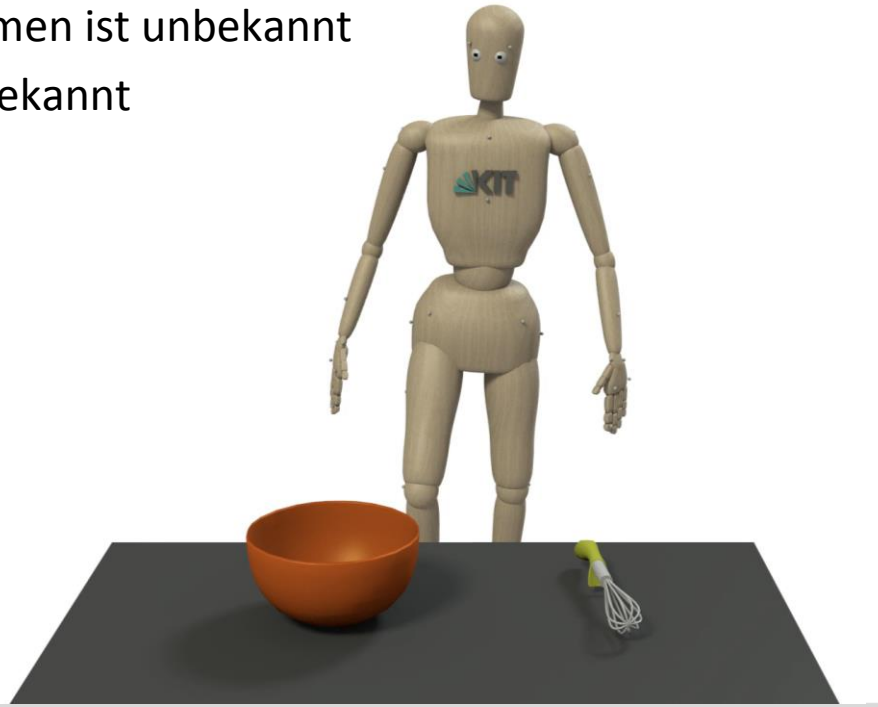
- Der Roboter muss die geeignetste Aktion auf Basis des **aktuellen Weltzustand**s bestimmen oder abschätzen
- Ziel: Eine **Zuordnung** zwischen Weltzustand und Aktion
  - Strategie zur Aktionswahl
- Um eine Strategie zu lernen zeigt ein Demonstrator dem Roboter Beispiele für mögliche Aktionen im aktuellen Weltzustand
  - Demonstrator: Lehrer
  - Roboter: Schüler
- Der Weltzustand kann nicht direkt bestimmt werden, sondern nur über die **Beobachtungen des Roboters** (Sensorik)
- Die gelernte Strategie erlaubt dem Roboter, Aktionen auf Basis seiner Beobachtungen auszuwählen

Vorverarbeitung: Segmentierung von Demonstrationen



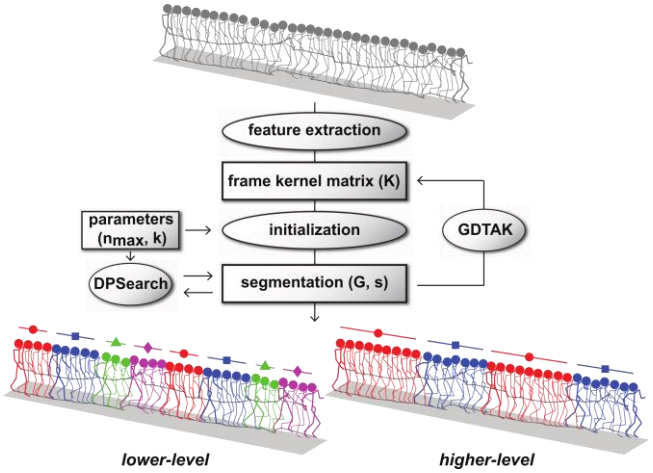
Motivation

- Demonstrationen bestehen meist aus Aktionssequenzen
- **Verständnis** von Demonstration **einfacher**, wenn diese in **kleinere Segmente** oder sogar **bekannte Aktionen** zerlegt sind
- Aber:
  - Welche/Wie viele Aktionen vorkommen ist unbekannt
  - Start und Ende von Aktionen ist unbekannt
  - Aktionen gehen nahtlos ineinander



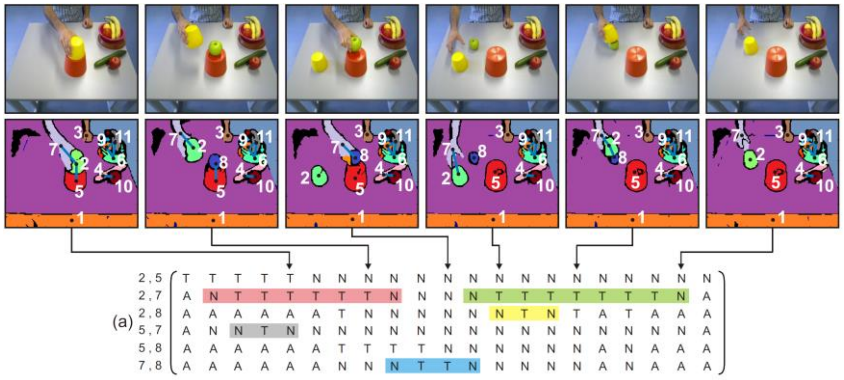
Segmentierung von Demonstrationen

- Hauptsächlich zwei Arten von Segmentierungen
  - Bewegungssegmentierung
    - Untersuchung der Bewegungstrajektorien nach
      - Mustern
      - Änderungen in der Bewegungsart
    - Algorithmen: HMM, PCA, Clustering, Template matching, Klassifikatoren



Zhou et al. 2013

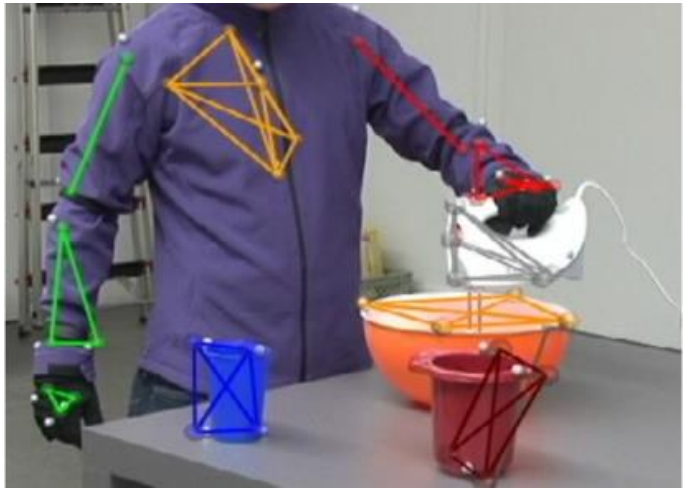
- Semantische Segmentierung (Aufgabensegmentierung)
  - Beobachtung des Lehrers und der Umgebung
  - Extraktion von semantischen Merkmalen
    - Zustände, Objekt-Relationen, Regeln

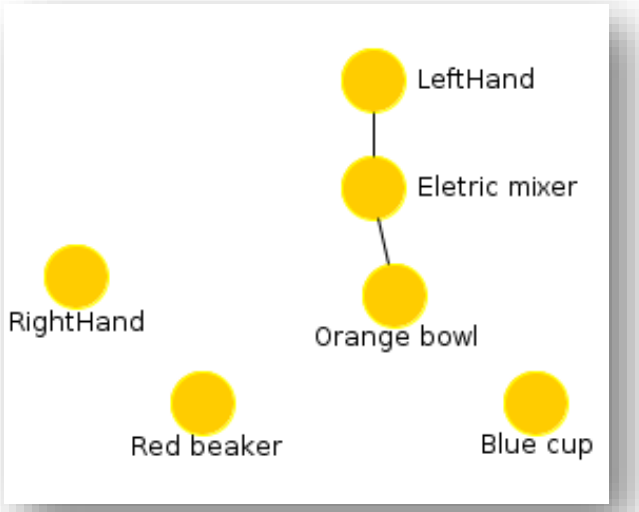


Aksoy et al. 2016

Aufgabensegmentierung: Idee

- **Objektkontaktrelationen** zur Repräsentation des Weltzustandes
- **Keyframes** werden bei **Änderungen** der Kontaktrelationen eingefügt
- Jedes Segment repräsentiert eine **Manipulationsaktion**,
  - Teig umrühren enthält z.B. greifen, rühren, abstellen, …
- Aber:
  - Segmente haben noch keine Label



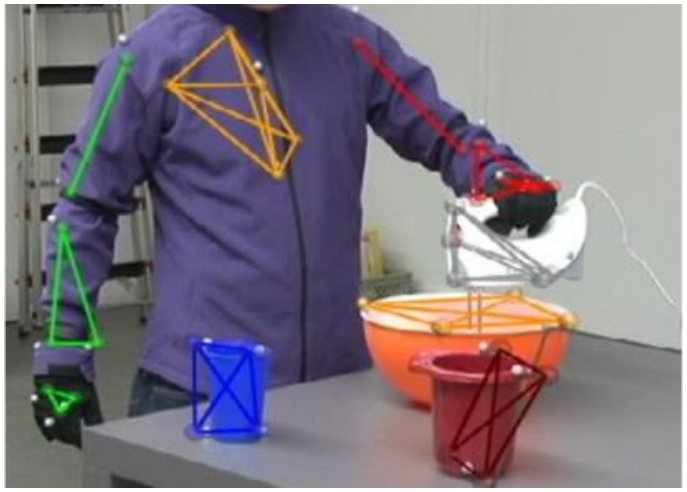


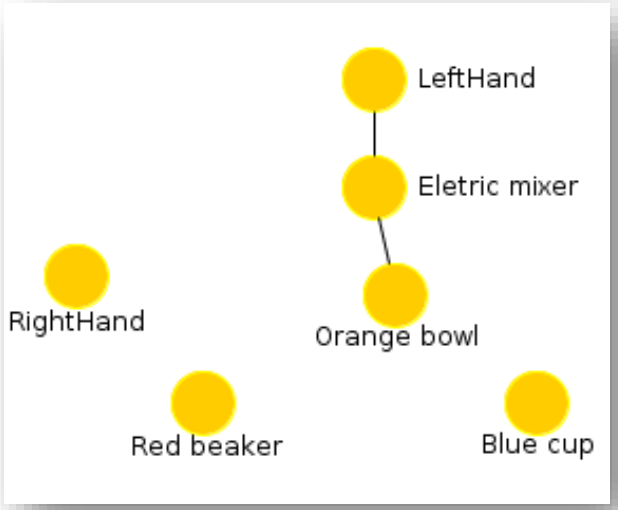
Aufgabensegmentierung: Idee

- Objektrelationen ermöglichen die Extraktion von Vorbedingungen und Effekten von Aktionen für **symbolische Planer**
  - Zustand am Segmentanfang: *LeftHand berührt nichts*
  - Zustand am Segmentende : *LeftHand berührt RoteTasse*

➔ *Vorbedingung: empty(LeftHand)*

➔ *Nachbedingung: in(RedCup,LeftHand), !empty (LeftHand)*



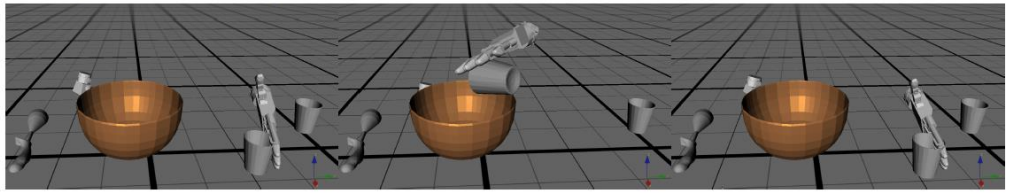


Hierarchische Aufgabensegmentierung

- **Hierarchische Aufgabensegmentierung** unter Berücksichtigung von **Bewegung** und relevanten **Objekten**
  - Semantische Segmentierung basierend auf **Objekt-Kontaktrelationen**
  - Bewegungssegmentierung basierend auf **Charakteristiken** von Trajektorien (Bewegungsdynamik)

Human Demonstration

Converted Demonstration

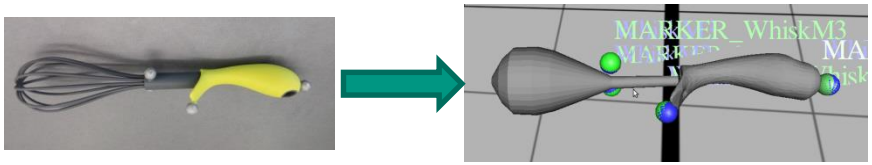


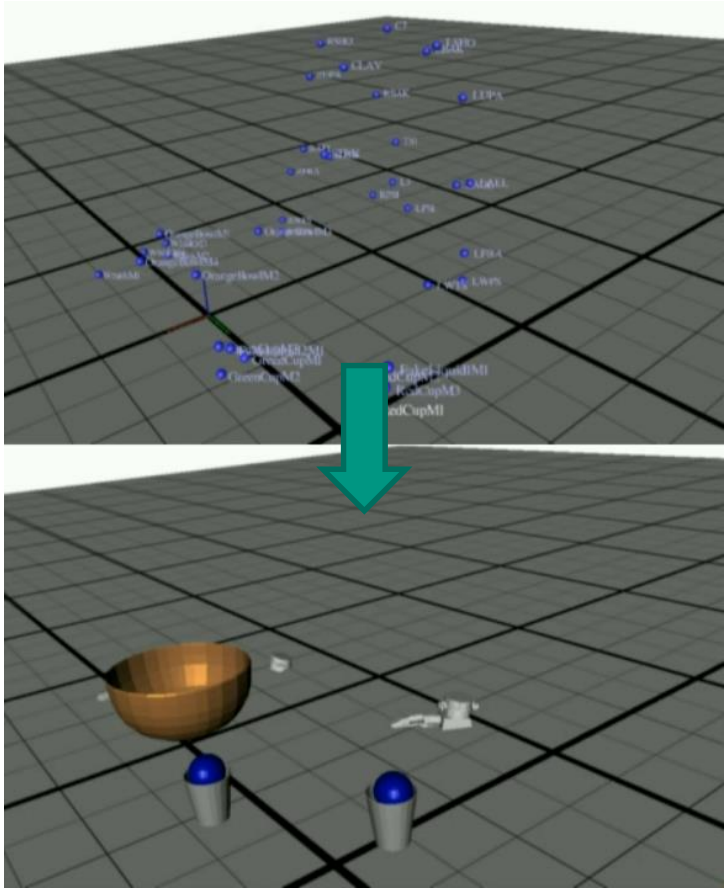
Hierarchical Segmentation

No contact	Cup in left hand			No contact
Grasp	Lift	Pour	Place	Retreat

Aufnahme der Demonstrationen

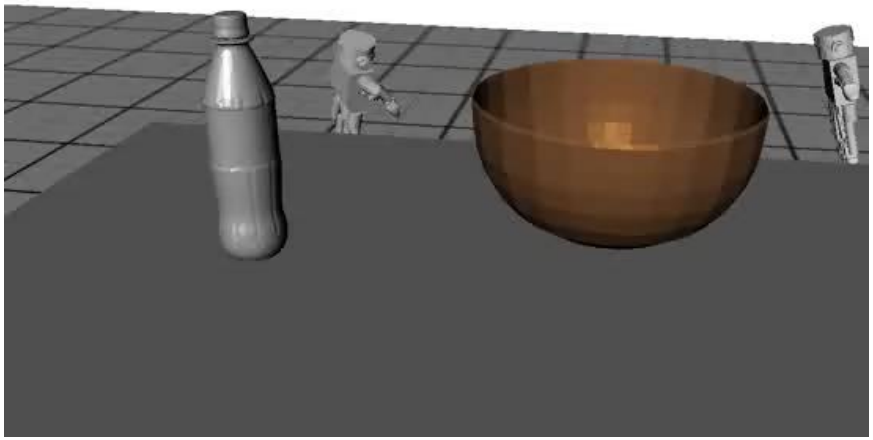
- Marker basiertes Motion-Capture von
  - Proband
  - Objekten
  - Umgebung
- Transformation zu 6D Posen Trajektorien
  - 3D Modelle mit virtuellen Markern
- Master Motor Map (MMM) Datenformat[1]

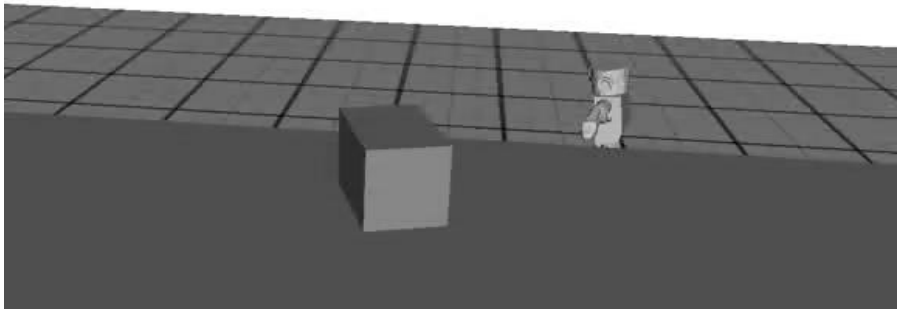


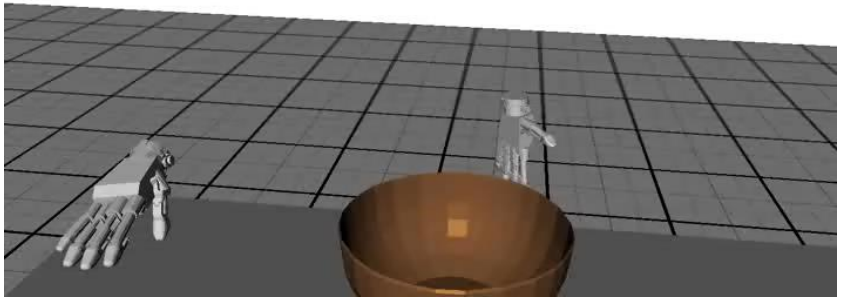


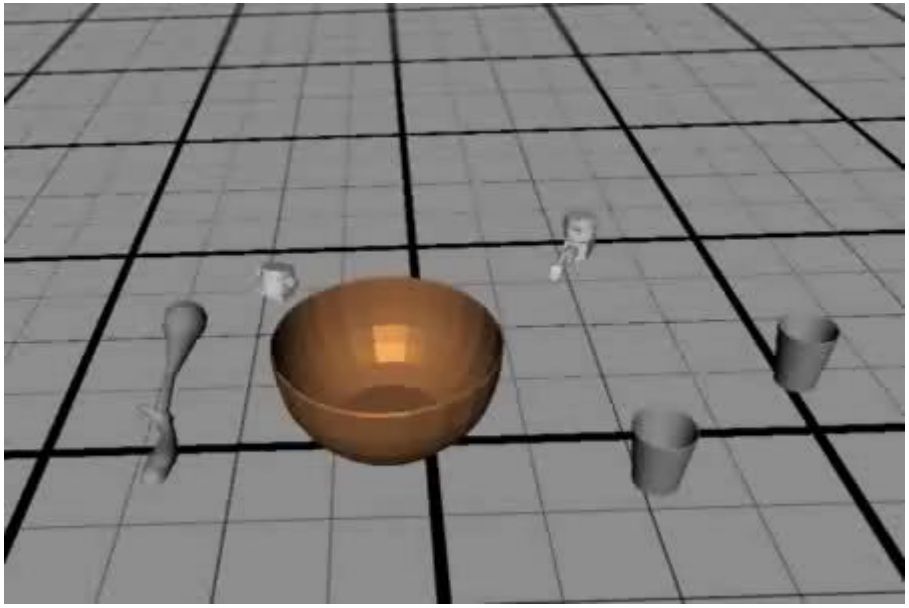
[1]https://mmm.humanoids.kit.edu/

Evaluation: Verschiedene Aktionssequenzen

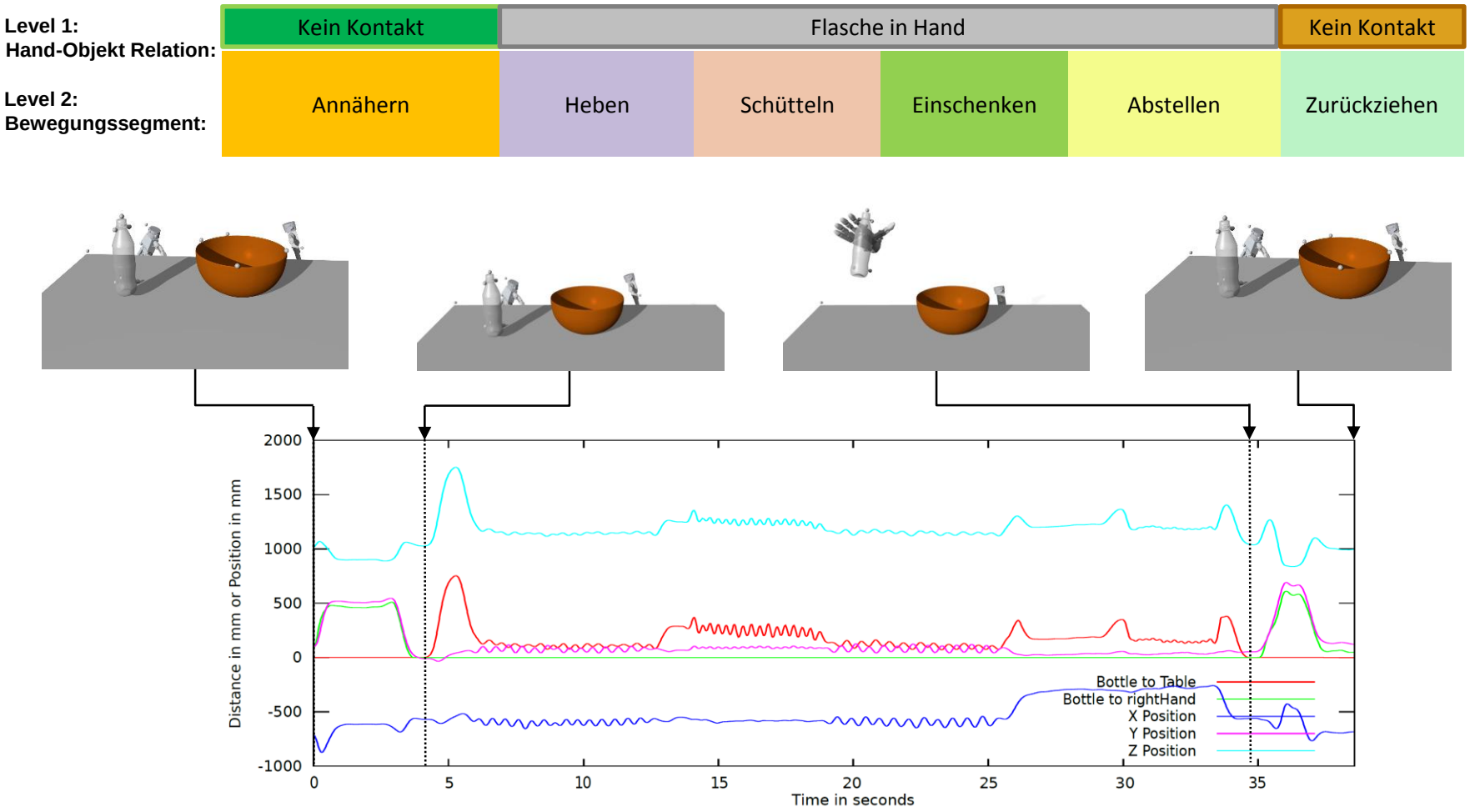








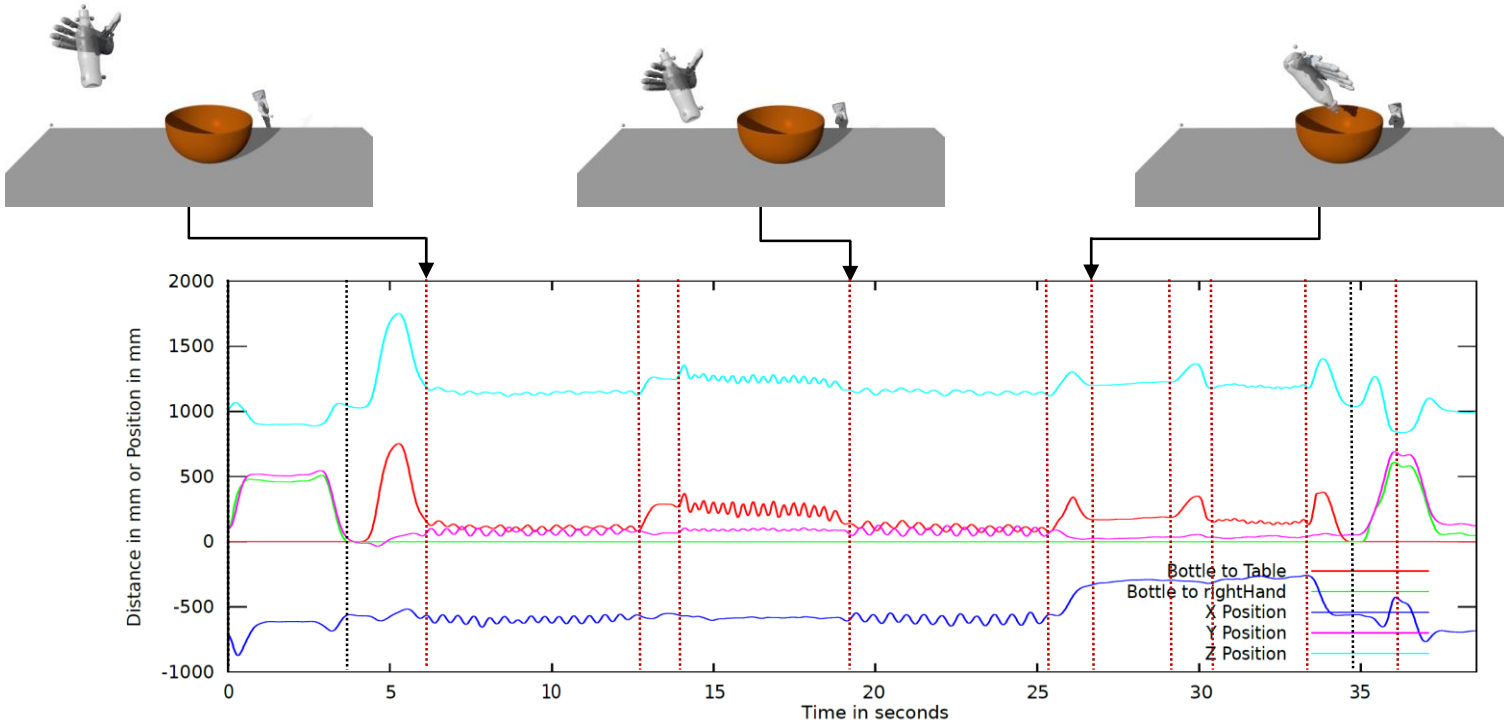
Level 1: Semantische Segmentierung



Level 2: Bewegungssegmentierung

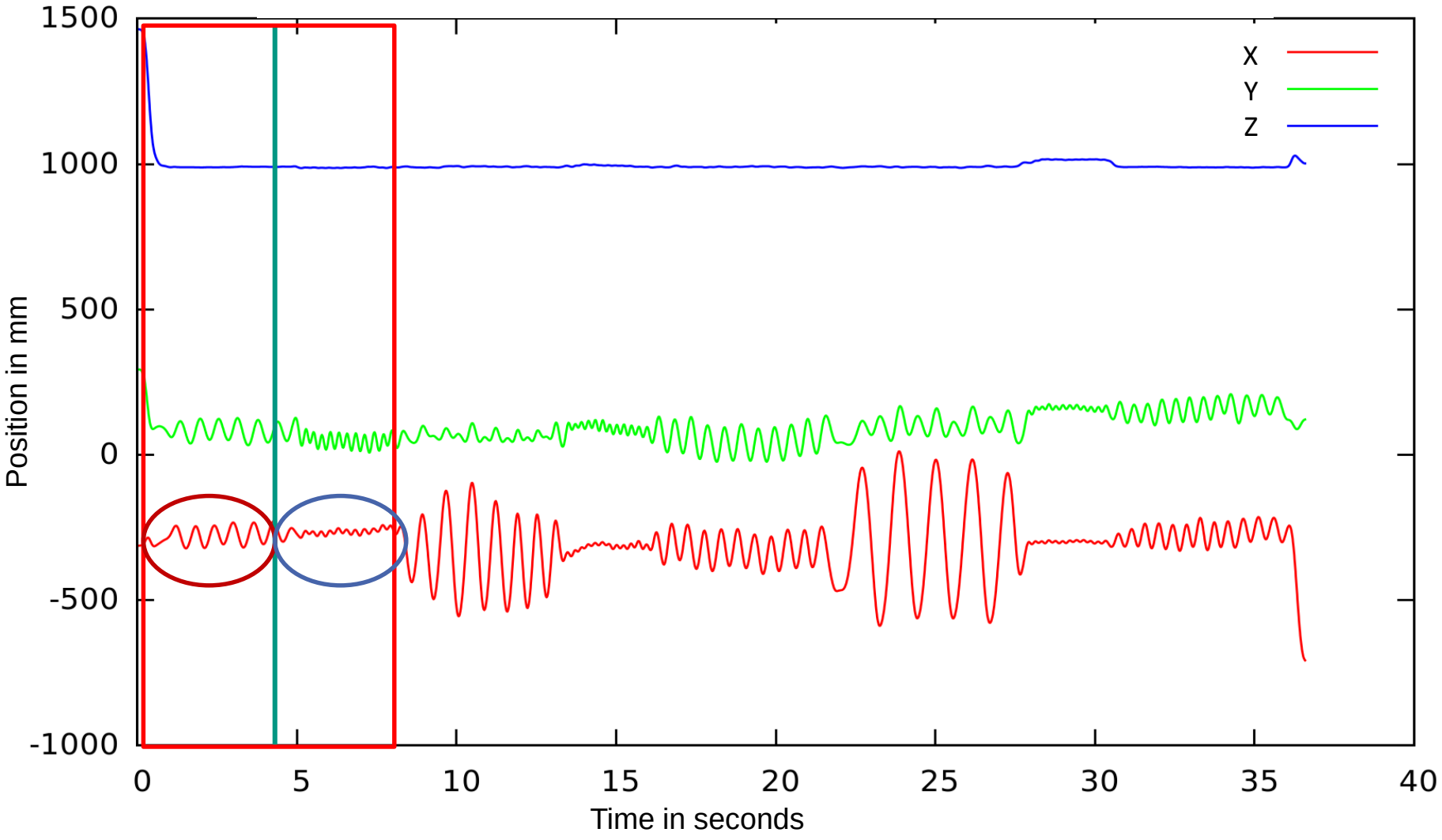
Hierarchische Segmentierung

Level 1: Hand-Objekt Relation:	Kein Kontakt	Flasche in Hand				Kein Kontakt
Level 2: Bewegungssegment:	Annähern	Heben	Schütteln	Einschenken	Abstellen	Zurückziehen

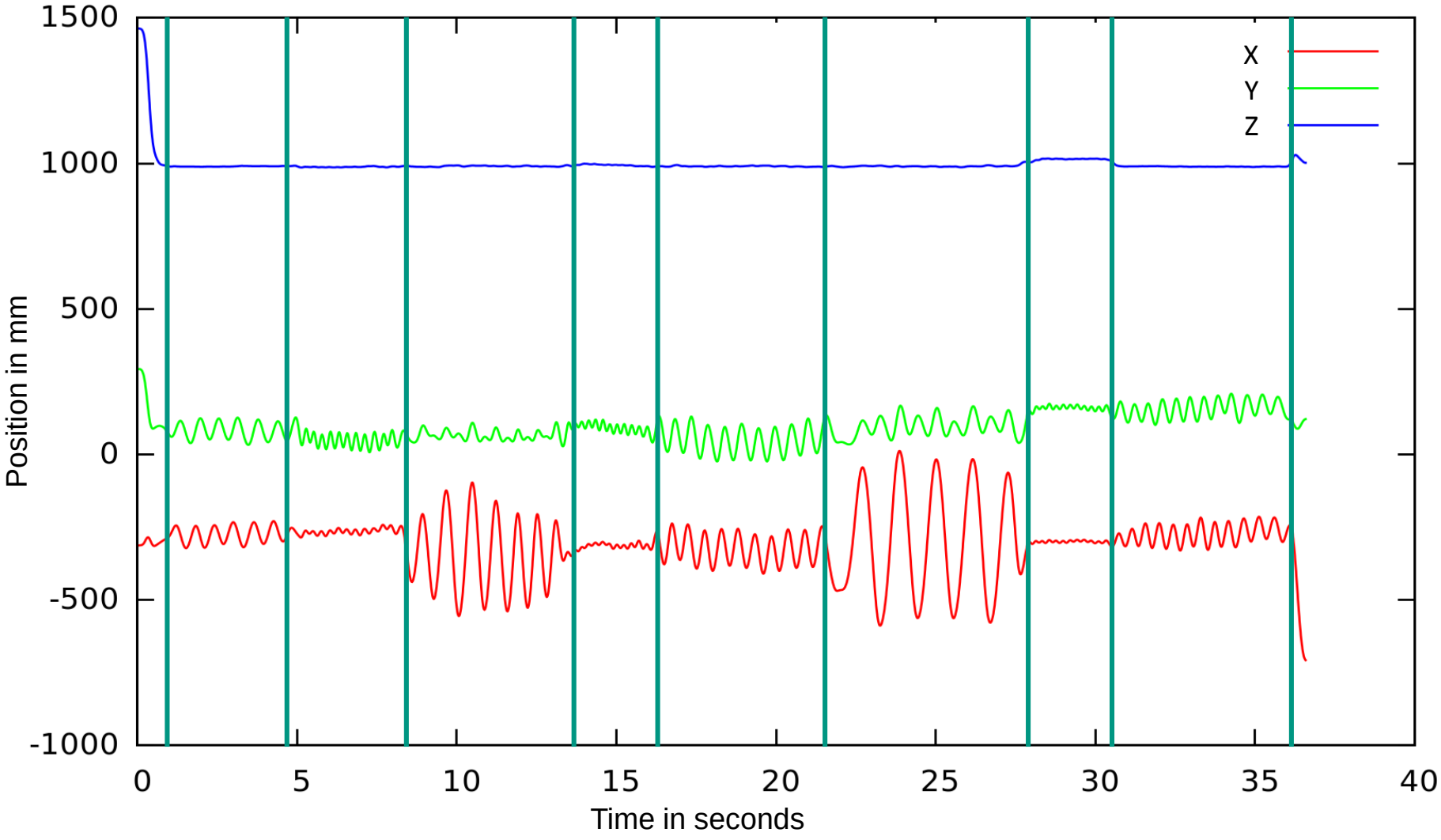


Bewegungscharakteristik Heuristik: Gleitendes Fenster

Frame score:



Bewegungscharakteristik Heuristik: Rekursive Suche



Bewegungscharakteristik Heuristik

- Heuristik basierend auf dem Beschleunigungsprofil
  - Messen der Länge der Beschleunigungskurve

$$s_{l,d}(t_c) = \int_{t_c-\frac{w}{2}}^{t_c-1} \sqrt{1 + a'_d(t)^2}\,dt \left(\frac{\hat{U}_l}{\hat{U}_r}\right)^2$$

$$s_{r,d}(t_c) = \int_{t_c}^{t_c+\frac{w}{2}-1} \sqrt{1 + a'_d(t)^2}\,dt \left(\frac{\hat{U}_r}{\hat{U}_l}\right)^2$$

- Iterative Suche des besten Keyframe Kandidaten mittels gleitendem Fenster
  - Vergleiche der Segmente links und rechts des Keyframe Kandidaten mit der Auswertfunktion *s*
  - Keyframe Qualität: $q_d = \begin{cases} \frac{s_{l,d}}{s_{r,d}} & s_{l,d} > s_{r,d} \\ \frac{s_{r,d}}{s_{l,d}} & s_{l,d} \leq s_{r,d} \end{cases}$
- Rekursive Unterteilung bis die Segmentgröße oder Qualität zu klein/niedrig ist

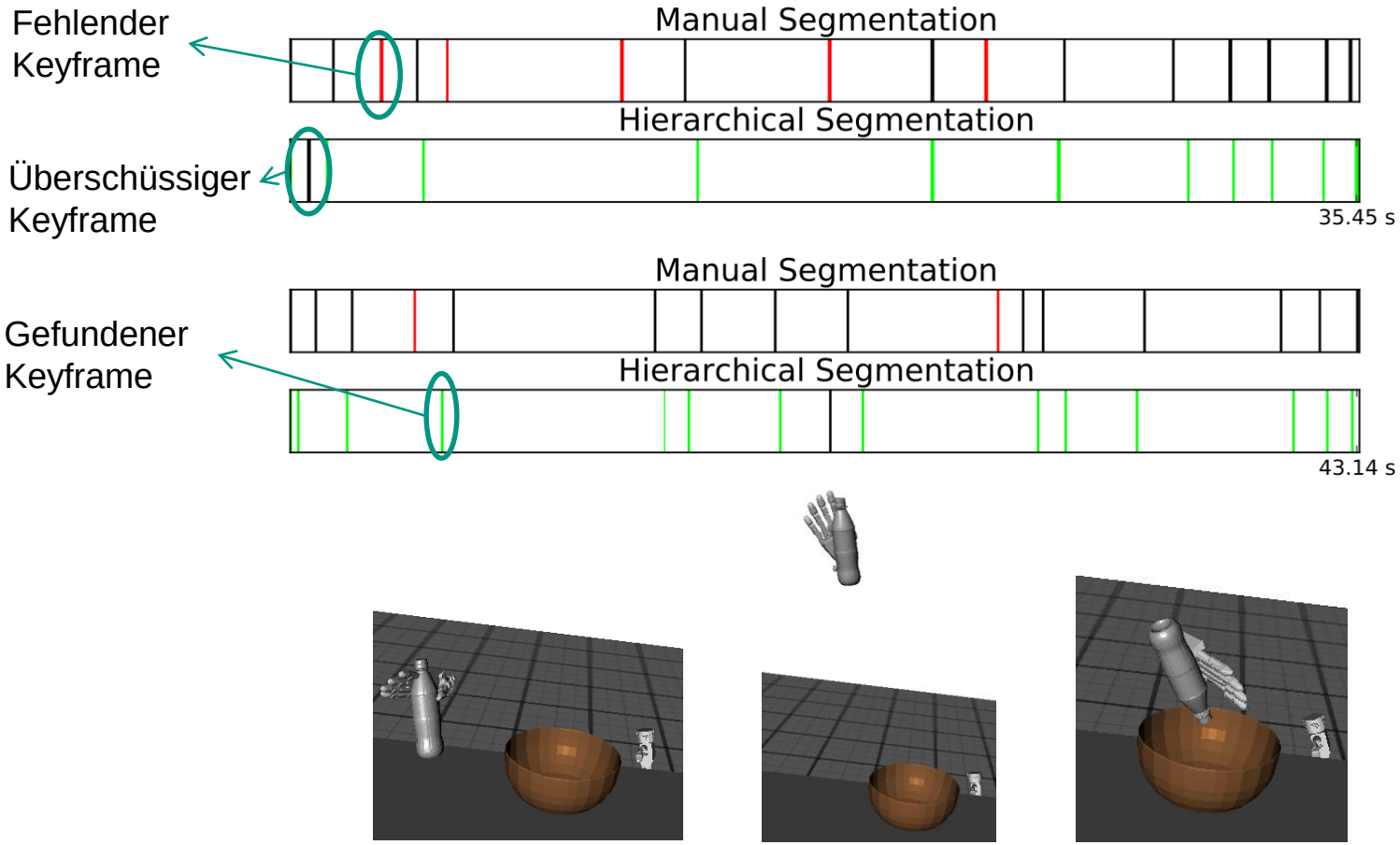
Evaluation der Segmentierung

- Neue Metrik für die Segmentierungsergebnisse
  - Bestrafung für fehlende und zusätzliche Keyframes

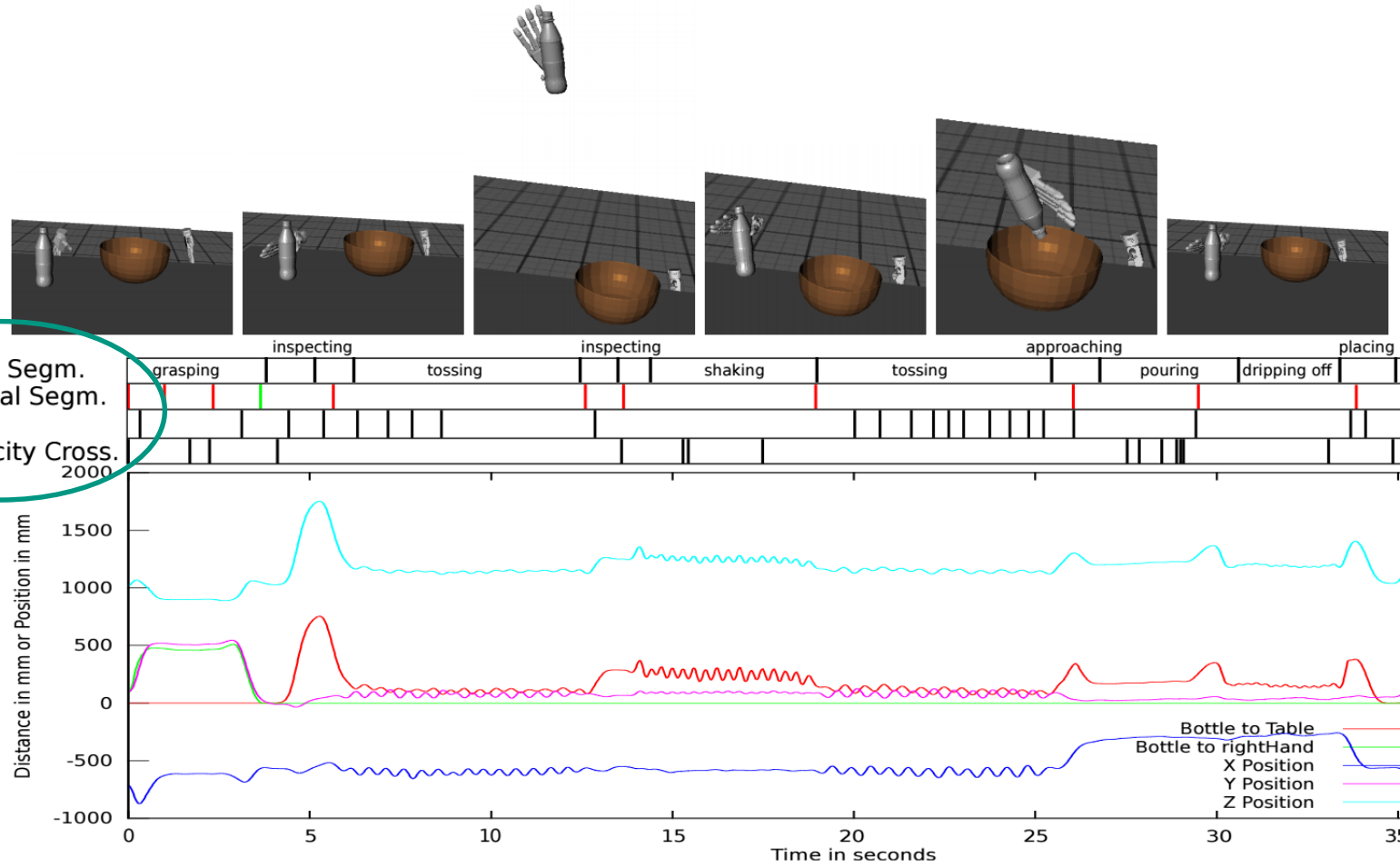
$$e = \underbrace{(m + f) * p}_{\text{Bestrafung für fehlende und zusätzliche Keyframes}} + \underbrace{\sum_{i} \min_{j}(k_{r,i} - k_{r,j})^2}_{\text{Mittlerer quadratischer Fehler zum nächsten Keyframe}}$$

Evaluation: Vergleich zu einer Referenzsegmentierung

- 2 Wiederholungen von Aktionssequenzen von Einschenken-Schütteln-Aktionen



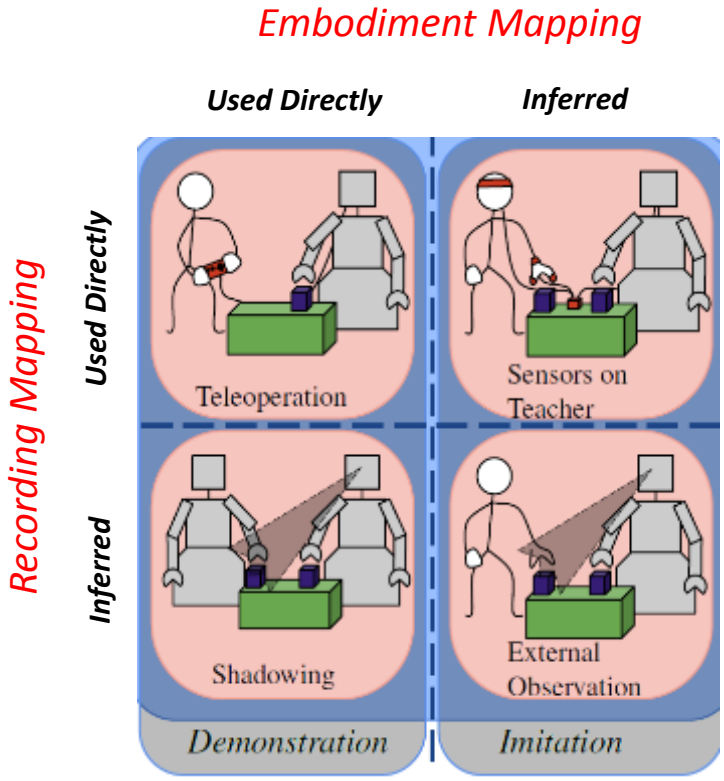
Vergleich mit anderen Segmentierungsalgorithmen



Hierarchical Segmentation of Manipulation Actions based on Object Relations and Motion Characteristics

Mirko Wächter, Tamim Asfour

Demonstrationsverarbeitung

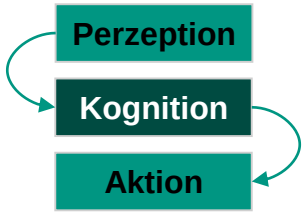


[Argall et al. 2009]
[Romero 2011]

Zwei Möglichkeiten Aktionen zu verarbeiten:

- *Batch Lernen*
  - Die Aktion wird gelernt wenn alle Aktionen/Wiederholungen aufgenommen wurden
- *Inkrementelles Lernen*
  - Die Aktionsrepräsentation wird nach jeder Aktion/Wiederholung gelernt/aktualisiert

Aktionsrepräsentation auf Trajektorienebene



- Lernen von Aktionen und Repräsentation geschieht oft zusammen
  - Repräsentation ist stark an der Lernverfahren gekoppelt
- Populäre Verfahren
  - Hidden Markov Models (HMM)
  - Dynamic Movement Primitives (DMP)
  - Gaussian Mixture Models (GMM)
- Verfeinerung von gelernten Trajektorien
  - Reinforcement Learning
- **Mehr dazu in der Vorlesung „Robotik-II: Humanoide Robotik“ im SS**
  - Hier: eine kleine Einführung

Dynamic Movement Primitives (DMP)

- Gegeben
  - Demonstration
    - Für jeden Zeitpunkt:
      - $\boldsymbol{y}_D$: Position
      - $\boldsymbol{v}_D$: Geschwindigkeit
  - Ausführung
    - Anfangsposition y_0
      Anfangsgeschwindigkeit v_0
    - Zielposition y_Z
    - Zeitdauer t

- Gesucht
  - Eine Trajektorie von y_0 bis zu y_Z
    ähnlich zu Demonstration $\boldsymbol{y}_D$

- Lösung: **Feder-Masse-Dämpfer System** mit dem **Perturbationsterm**

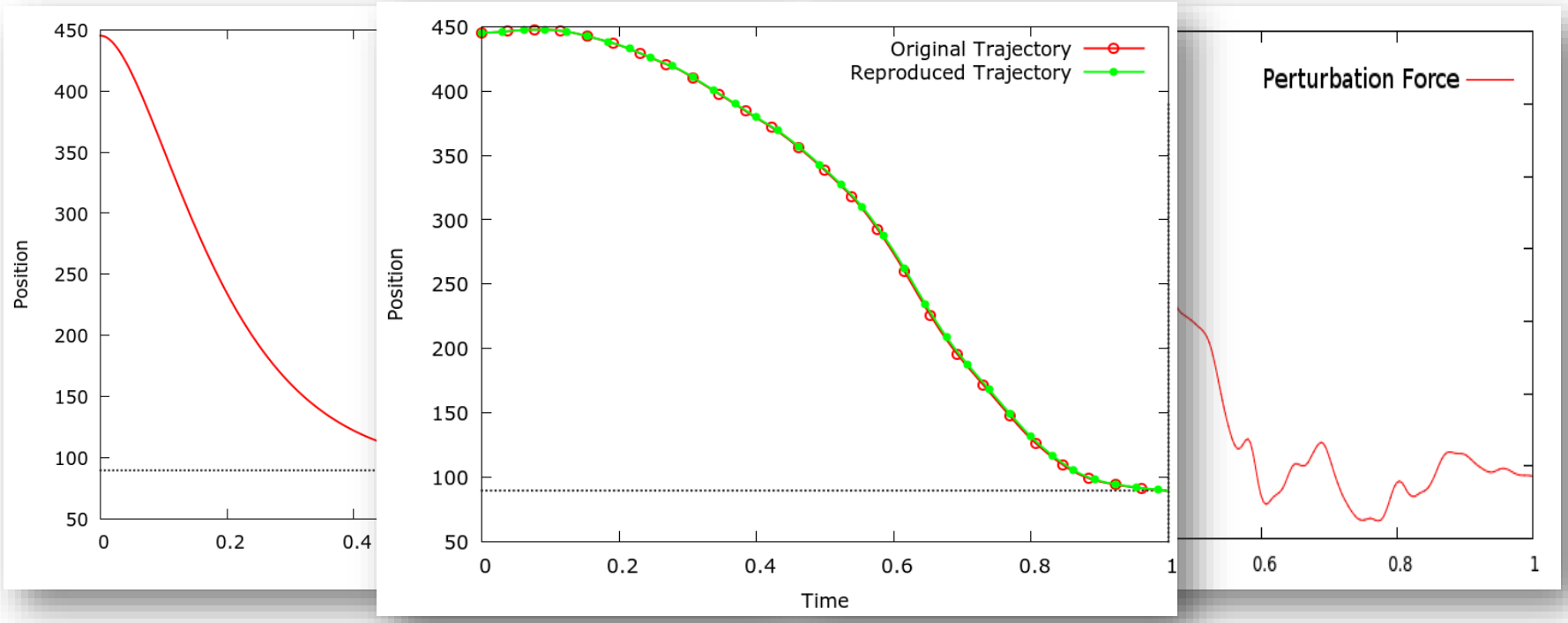
Dynamic Movement Primitives

- Feder-Masse-Dämpfer System mit dem Perturbationsterm

$$\tau\ddot{y} = K(g - y) - D\dot{y} + f_\theta(g - y_0)$$

Feder-Masse-Dämpfer System : Ziel Attraktor

Perturbationsterm: Gestalt Attraktor



Dynamic Movement Primitives

- Feder-Masse-Dämpfer System mit dem Perturbationsterm: Transformationssystem

$$\tau\ddot{y} = K(g - y) - D\dot{y} + f_\theta(x)(g - y_0)$$
$$\tau\dot{x} = -\alpha x$$

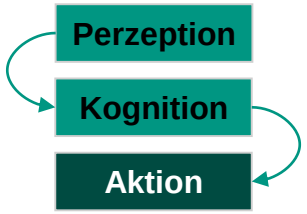
- $f_\theta(x)$wird durch die Beobachtung von y_D gelernt.
- Hyperparameter für die Anpassung:
  - τ: Zeitlicher Faktor -> Anpassung der Geschwindigkeit
  - g: Ziel -> Anpassung der Zielposition
  - y_0: Skalierungsfaktor -> Anpassung der Anfangsposition

- Kanonisches System: $\tau\dot{x} = -\alpha x$
  - Das Transformationssystem ist hiermit nicht unmittelbar von der Zeit abhängig

Inhalt

- Übersicht
- Kernfragen von Programmieren durch Vormachen
- **Perzeption:** Arten der Demonstration
  - Kinästhetisches Lernen, Teleoperation, Shadowing, Beobachtung
- **Kognition:** Lernen und Repräsentation von Fähigkeiten und Aufgaben
- **Aktion:** Ausführung von gelernten Fähigkeiten und Aufgaben
  - Ausführung von Aktionen mit online Adaption
  - Ausführung von komplexen Tasks in dynamischen Umgebungen

Trajektorienausführung



- Generierter Trajektorie $Y = (y_t)_{t=1}^{T}$ folgen
  - z.B. PD Regler:

$$u_t = k_p(y_t - y) - k_d v$$

k_p: Proportionalerbeiwert
k_d: Differenzierbeiwert
y: aktuelle Position
v: aktuelle Geschwindigkeit

- Direktes Kontrollsignal (speziell für DMP)

$$u_t = \ddot{y} = \frac{1}{\tau}(K(g - y) - D\dot{y} + f_\theta(x_t)(g - y_0))$$

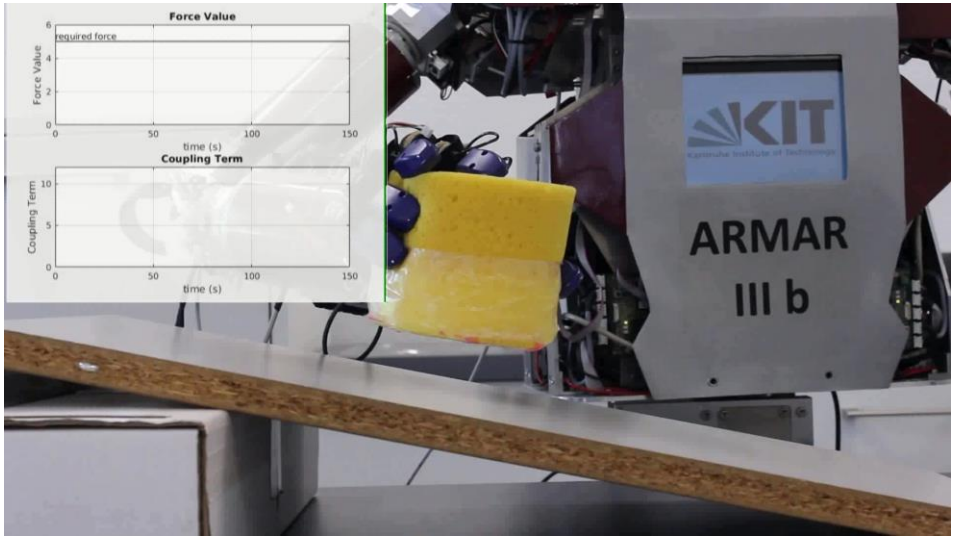
Lösung vom kanonischen System zum Zeitpunkt t

Online Anpassung bei der Ausführung

- Online Anpassung
  - Online Anpassung ist möglich durch den Kopplungsterm in DMP
  - Online Anpassung ist schwierig für GMM und HMM, weil beide nicht dynamisch sind

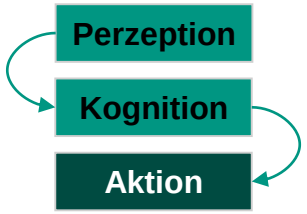
- DMP Online Anpassung $\tau\ddot{y} = K(g - y) - D\dot{y} + f_\theta(x)(g - y_0) + C$

Online Kraft-Anpassung



Quelle: Zhou et al., 2016

Ausführung von komplexen Aufgaben



- Repräsentation von komplexen Aufgaben mit Aktionssequenzen
- Ausführung von Aktionssequenzen in dynamischen Umgebungen
  - Aktionen einzeln programmiert oder gelernt
  - Aktionssequenzen hängen vom Weltzustand ab
    → Einsatz von symbolischen Planern möglich
- Zustand der Umgebung muss wahrgenommen werden
  - Selbstlokalisierung (z.B. mit Laserscanner)
  - Objektlokalisierung (z.B. Stereo-RGB, RGBD)
  - Symbolisches Umgebungswissen
    - Landmarken
    - Zustand der Umgebung (Tür offen/geschlossen)

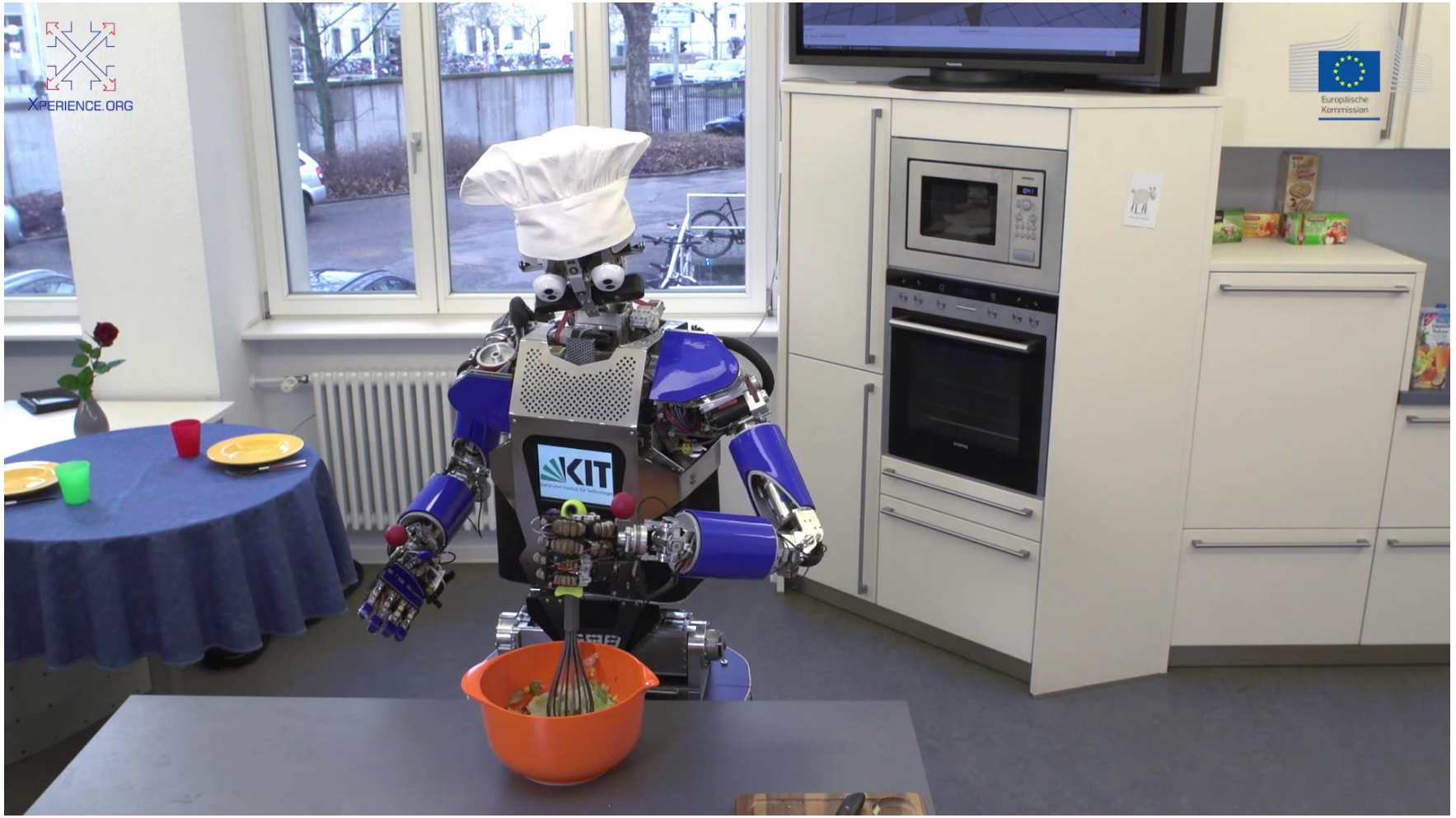
  → Abbildung von kontinuierlichem Wissen auf symbolisches Wissen
  z.B. 6D Objektpose → $\mathrm{auf}(\mathrm{Objekt}, \mathrm{Tisch})$

Ausführung von komplexen Aufgaben (2)

- Initialer Weltzustand
  - Klassische Planer benötigen vollständigen Weltzustand
  - **Problem**: Roboter weiß zunächst nicht, wo die Objekte sind
  - **Lösung**: Hypothesen generieren anhand von Roboterwissen aus
    - Erfahrung
      - gewonnen aus früheren Ausführungen
    - Allgemeines Objektwissen
      - z.B. Data-Mining, Semantische Netze

- Fehlerbehandlung
  - Erkennen von Problemen
    - Fehlerhafter Weltzustand beim Planen (bei der Ausführung entdeckt)
    - Fehlerhafte Ausführung
  - Lösungen:
    - Neu planen auf dem aktuellen Weltzustand
    - Wenn nicht möglich: Suche nach Alternativen
      - Gemeinsame Affordanzen: Soda nicht verfügbar -> Soda ist trinkbar -> Orangensaft
      - Ähnliche Funktion aus visuellen Merkmalen (Form der Objekte)
      - etc.

Ausführung komplexer Aufgaben mit symbolischer Planung und Objektersetzung



Zusammenfassung

- PdV ermöglicht es Fähigkeiten zu transferieren
- PdV kann den Lernprozess beschleunigen (im Vergleich zu Reinforcement Learning bzw. Versuch-und-Irrtum Methoden)
- PdV liefert Vorschläge für die optimale Lösung für eine Aufgabe zu ermitteln
- Beim Imitationslernen können Positiv- und Negativbeispiele verwendet werden
- Der Roboter muss über eine Bibliothek an generischen Fähigkeiten (skills) verfügen und in der Lage sein diese an den Kontext anzupassen
- Imitation ist
  - *Nicht geeignet* wenn eine gute Lösung des Lehrers keine gute Lösung des Nachahmers ist
  - *Nicht geeignet* wenn die Demonstrationen des Lehrers stark verrauscht sind

A. Billard, S. Calinon, R. Dillmann and S. Schaal, *Handbook of Robotics*, Springer, pp. 1371-1394, 2008.

B. Argall, S. Chernova, M. Veloso and B. Browning, *A survey of robot learning from demonstration*, Robotics and Autonomous Systems, 2009.

J. Romero, *From human to robot grasping*, PhD Thesis, 2011

Mehr zu dem Thema insgesamt in der
Vorlesung „Robotik-II: Humanoide Robotik“
im SS