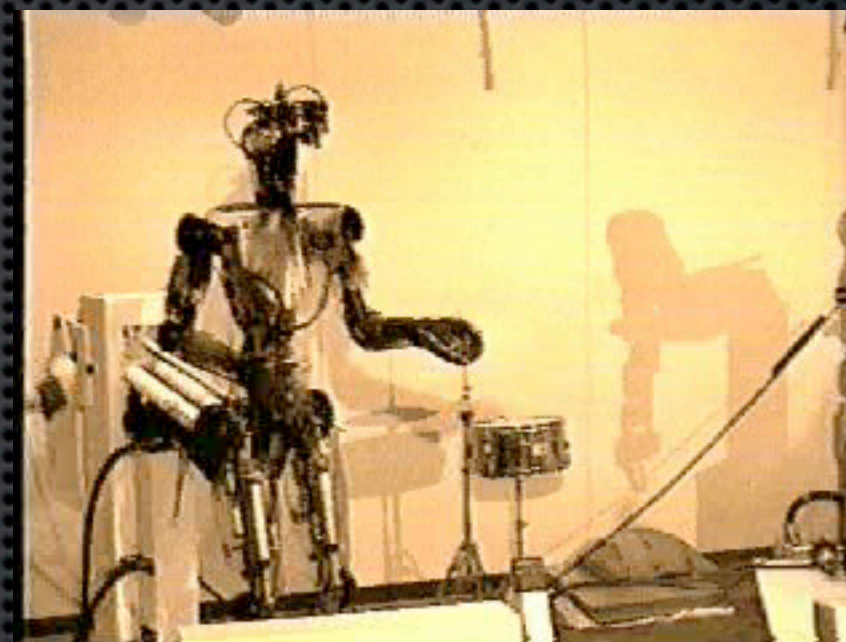
Grounded Humanoid Representations

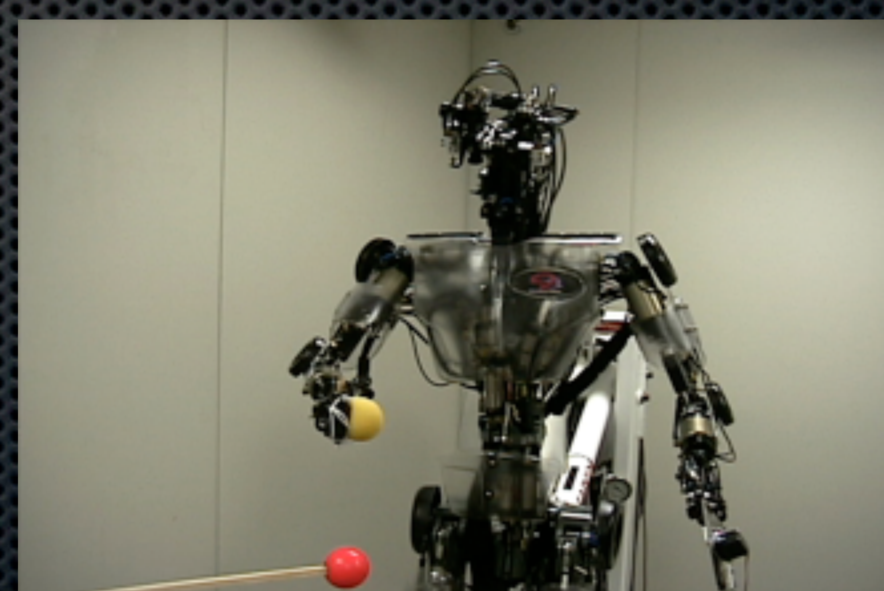
objects, actions and movements

Gordon Cheng@TUM

with contributions from: Michael Beetz (TUM), Marcia Riley (TUM), Federico Ruiz (TUM) and Ales Ude (IJS)

Humanoid Robots





Motivation

- We learn about our world with such ease
- Through exploration with our body upon actions on the real-world
- What does it mean for humanoid robots?
  - Having a body
  - Being able to interact with the world

Grounding movements/ actions

- Through observation then reproduction
- Refinements need to be accounted for
  - Through Exploratory
  - Through Instruction

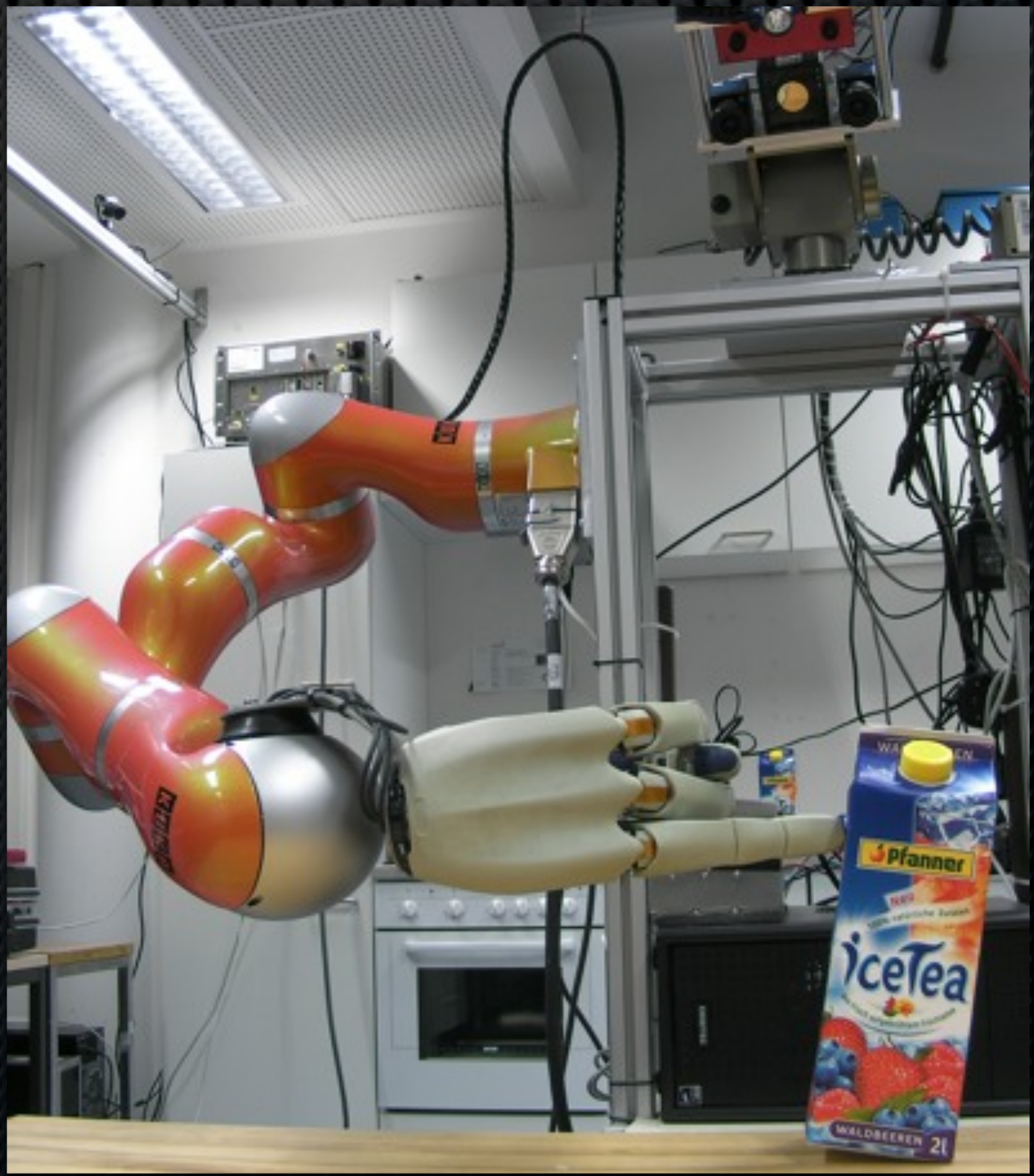
Action Language For Robot Control

Federico Ruiz-Ugalde (TUM), Michael Beetz (TUM),
Gordon Cheng, TUM

Action Language For Robot Control

There is a Strong connection between language and action.

- "Push Ice Tea while maintaining orientation"
- "Topple Ice Tea"
- "Touch Ice Tea, don't move it"

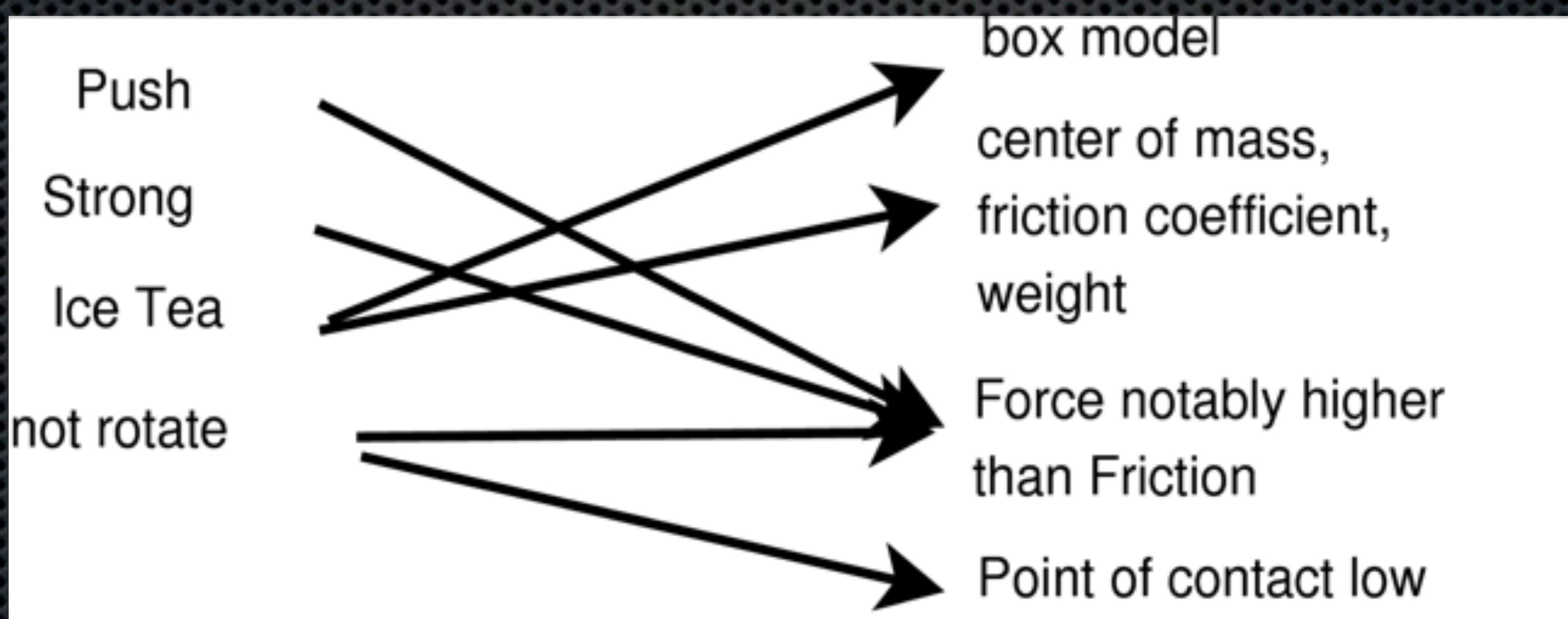


Action Language For Robot Control

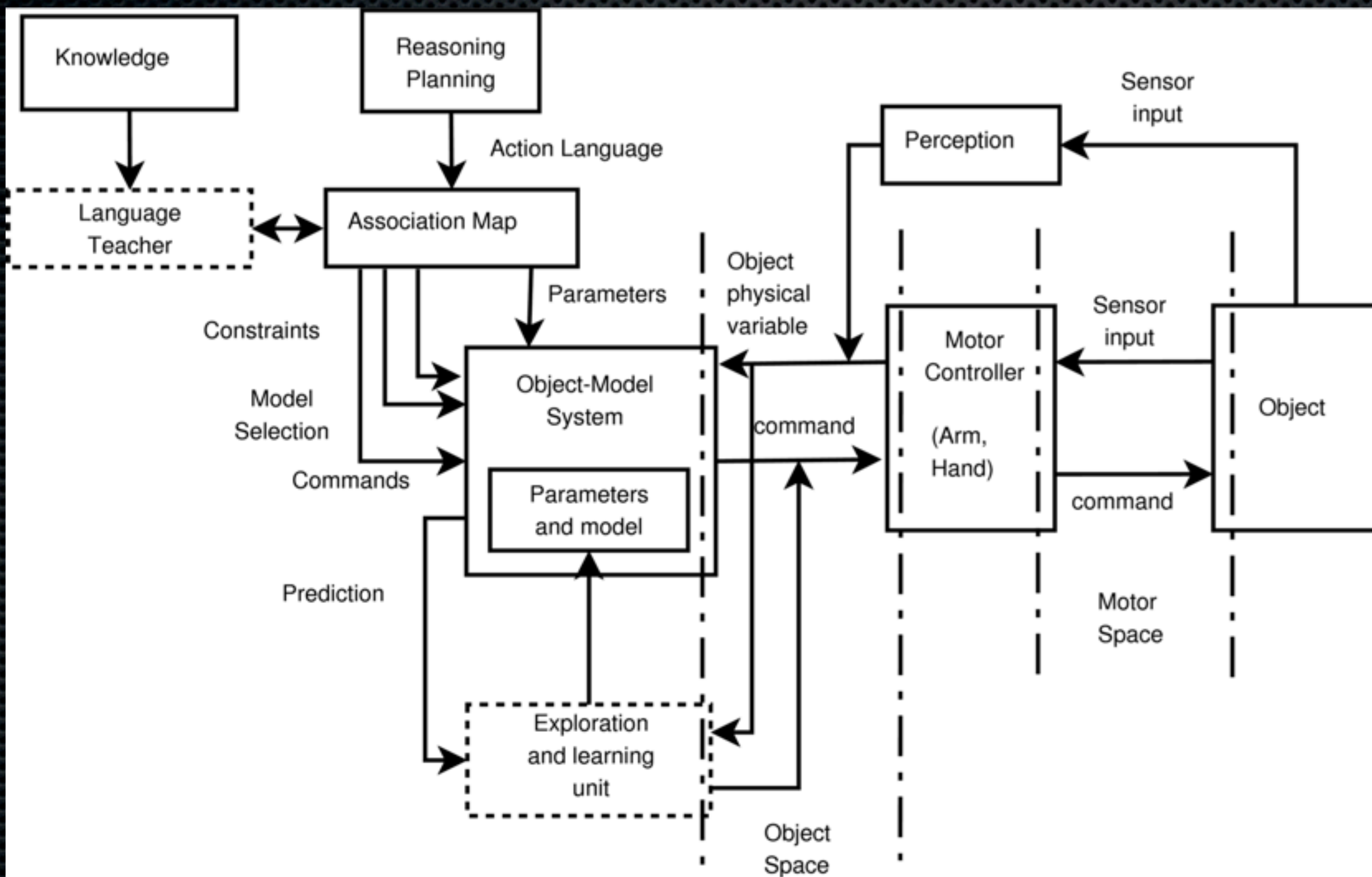
- We can use an action language to translate high level instructions to object control.
- Object control can translate object manipulation commands to motor commands
- Motor control can translate motor commands to move limbs, and exert forces on objects.
- Control is centered on the object (affordance) given the robot's capabilities (grounded).

Action Language

Example: “Push ice tea strongly while maintaining orientation”



System Together

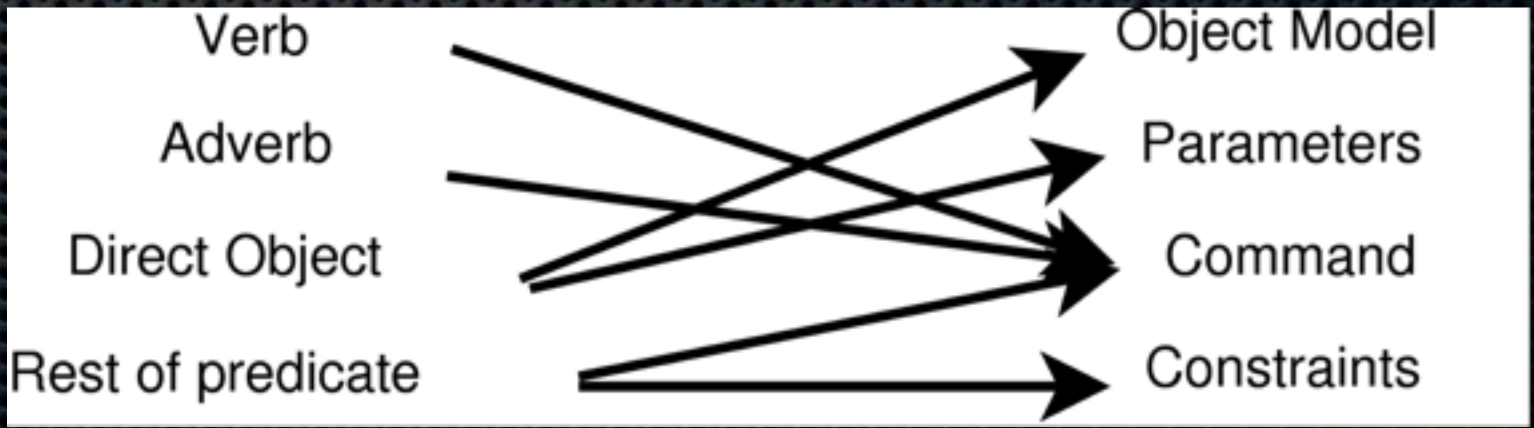


Reasoning and Planning

- Given a desired goal, it generates a plan which contains sentences using an action language.

Action Language

- Imperative.
  - "Open door", "Pour water into the mug"
- Subject is always the robot or the limb.
- Predicate (verb, adverb, direct object, rest of predicate) will determine the object model, parameters, commands and constraints. (Association map)

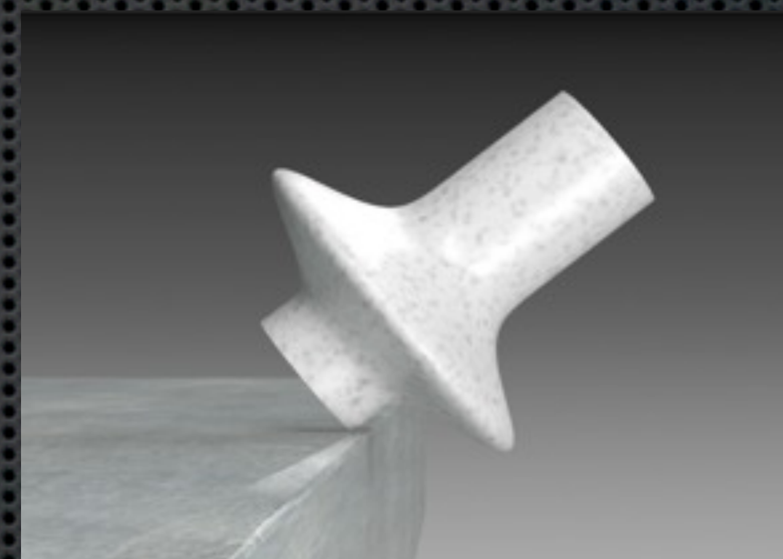


Finding An Association Map.

- Prior knowledge (web, other robots)
- **Teaching** (by action observation and execution)
- P(object, verb, adverb| model, params, command)
- Learn a complete enough association map to give our robot good manipulation capabilities.

Object Model System

- If we see this pictures what comes to our mind?



- What is about to happen.
- Prediction.

Object Control

"Open the door"

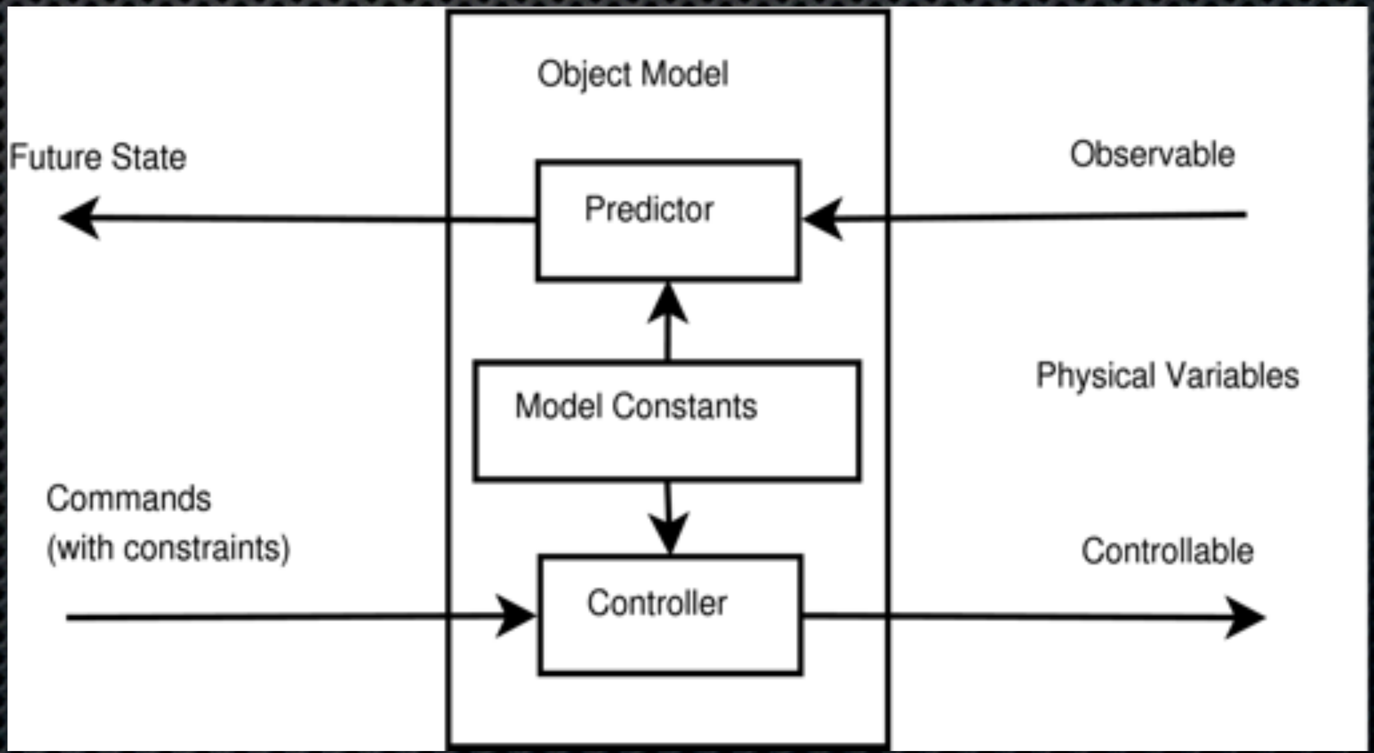
- Given the condition that the door is initially closed.
- We know (from prediction) what is going to happen when we rotate the handle.
- What do I have to do to open the door? Inverse problem.

Object Control

- The inverse problem is easy when the door is closed. We can only open the door by rotating the handle and pulling from it.
- But if the door is already a bit open and we want to open the door fully, then we can pull the door not only from the handle but also from the door itself from infinite points in the door.
- Optimize. (Put more constraints to the problem)
  - Use less energy, faster, and so on until we find a unique solution.

Object Control.

- Control is centered on the object (affordances) given the robot's capabilities (grounded)
- We use a multiple paired forward (predictor) and inverse (controller) model system to control the objects.



Models And Parameters

- Mechanics “Classical Mechanics” (rigid body, force balance, kinematics, dynamics, fluids)
- Machine Learning.

Exploration and Learning

- The robot can learn the parameters and also the forward and inverse model of the objects, letting the robot play with the objects.
- Exploration can be guided to minimize time or effort to find the parameters.

Videos of friction and playing

Motor Control

- Translates internal object space variables to external motor control signals. (e.g. inverse-forward kinematics and dynamics)
  - Measured torques in the arm joints are translated to estimated forces in the end-effector.
- Models the robot limbs. (arm, hands, legs, head)
  - **Kinematic chains, weights, inertial-tensors, joint friction, actuator model**.
- It translates the internal object model system signals into motor commands.

Perception

- It provides the object model system with more object state information.
- It does this by translating between camera signals to the internal object space representation.

An Example: Sliding A Box.

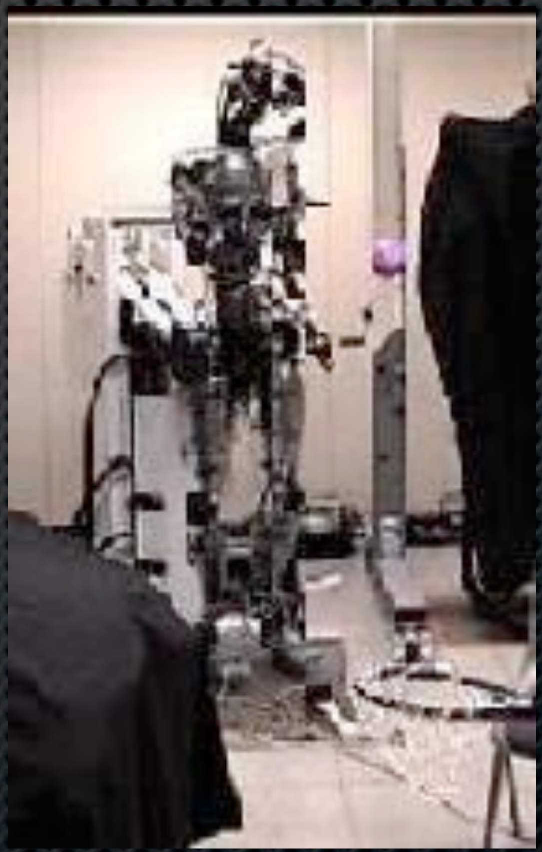
- Model of a box.
- Friction: $F_f \leq \mu_s F_W$
- It will slide if $F_{ext} \leq F_f$, it will not move if $F_{ext} \leq F_f$
- Constant: μ_s, which is specific to object instance.

Toppling An Ice Tea Box

The predictor must answer:

- Where can the box rotate? Around A or B?
  - Model: Iterative algorithm (torques on base vertices)
  - Relevant parameter: Base Shape
- Will it rotate? How strong?
  - Model: Forces balance.
  - Relevant parameter: Center of mass.

Grounding humanoid bodily motion



Marcia Riley, TUM
Ales Ude, Jozef Stefan Institute
Gordon Cheng, TUM

Coaching Introduction

- Human motor skill learning models & human coaching
- Adapt appropriate formalisms to humanoid robot coaching
- Experiment is shown

Humanoid Robot Coaching Motivation

Coaching paradigm: robot acquires motor skills with the aid of a human coach
Modelled on human-skill transfer between a coach and student

Motivation: reduce time and ease of creating robot behaviors

Efficient

Proven merits in accelerating human learning
Does this efficient learning have applications in humanoid domain
Constrains the solution space for the behavior
Provides critical evaluation and guidance to reach a correct solution faster than can realized alone

Intuitive

Uses a familiar human paradigm for skill transfer
Not necessary for person to learn a new skill set to coach a robot (they have experience from their own lives)

Human Learning Models

Gentile model (1972, 1987, 2000)

- 2-stage model with respect to **Goals**
  - **Initial**: acquire movement patterns
  - **Later**: capability to adapt patterns to specific situations increase consistency & economy of effort

(Dave Thompson)

Human Learning Models

Fitts & Posner learning model (1967)

- Cognitive (verbal) phase of learning
  - beginner
  - patterns of coordination in new task acquisition
  - rapid improvement
- Associative phase
  - subtle adjustments, gradual improvement
  - development of internal reference of correctness
- Autonomous stages
  - expert who is ready to cope with strategies
  - performance is automatic, minimal improvement (months or years)

(University of Sydney, School of Exercise and Sport Science, Dave Thompson)

Human Learning Models

- Strategies are applicable to the autonomous phase of learning thinking is that we need superior skill to assess strengths & weaknesses
  - ourselves
  - our opponents
- Performers in autonomous phase are experts:
  - less need of conscious task attention
  - better problem solving, adaptability
  - attends to relevant features quickly
  - makes decision with less information recognizes patterns sooner
  - better use of visual information as action predictors
- Experts require 10 years of intense practice (Ericsson, Krampe, Tesch-Romer 1993)
- Requires deliberate intense practice including instruction

Human Learning Models

Example: expert attending to relevant features quickly (Savelsberg et al. 2002)

Expert soccer goalkeepers

- more accurate in predicting direction of penalty kick
- took more time before initiating a movement
- made fewer corrective movement

Novices: looked longer at trunk, arms, hips

Experts: attended more to kicking leg, non-kicking leg, ball areas,

especially as impact approached

(University of Sydney, School of Exercise and Sport Science)

Information Feedback

Intrinsic
- kinesthetic information from performing
- relevant cues when performing (lines on a tennis court)

Artificial
- augmented feedback: giving additional information during or immediately after performance (Rushall, 1972, sports education)

Terminal feedback or KR (knowledge of results)
- from a completed action (making a jump shot)

Coaching is artificial concurrent IF.
Useful if leads to learning of intrinsic cues for success.
(Can you perform successfully when the coach is not around?)

(University of Sydney, School of Exercise and Sport Science,
Kelso, *Human Motor Behavior*, 1982)

Model for Humanoid Coaching

Fitts & Posner learning model (1967)

Cognitive (verbal) phase of learning

- beginner
- patterns of coordination in new task acquisition
- rapid improvement

Humanoid Robot Coaching Model

Associative phase

- subtle adjustments, gradual improvement
- development of internal reference of correctness

Ongoing Work

Autonomous stages

- expert who is ready to cope with strategies

(University of Sydney, School of Exercise and Sport Science, Dave Thompson)

Coaching in Human Skill Transfer

A coach is an expert who improves student performance.

How does the coach communicate relevant information to the student?

Type and timing of information are key.

Type: demonstration and verbal commands (most common methods)

Much more effective when used together

- - In showing videos of complex movements, performance actually decreases if no
- verbal information accompanies video (Schmidt & Lee, 1999)
- **Explanation**: too much simultaneous information is presented to make
- correct correspondence between actions and goals

Relevant information is hidden among irrelevant information

- best performances occurred when specific feedback was given

(Why is perception alone not sufficient for learning complex tasks?)

Types of Information in Coaching

Demonstration includes:

- performing correct movement (mirror neurons)
- physically guiding student through movement

provides kinesthetic information from performing (intrinsic feedback)

Common Verbal Commands: kinematic descriptions of motion

Coaches are especially good at identifying and correcting kinematic errors

"bend your knees when you land"

Patterns of coordination

Position

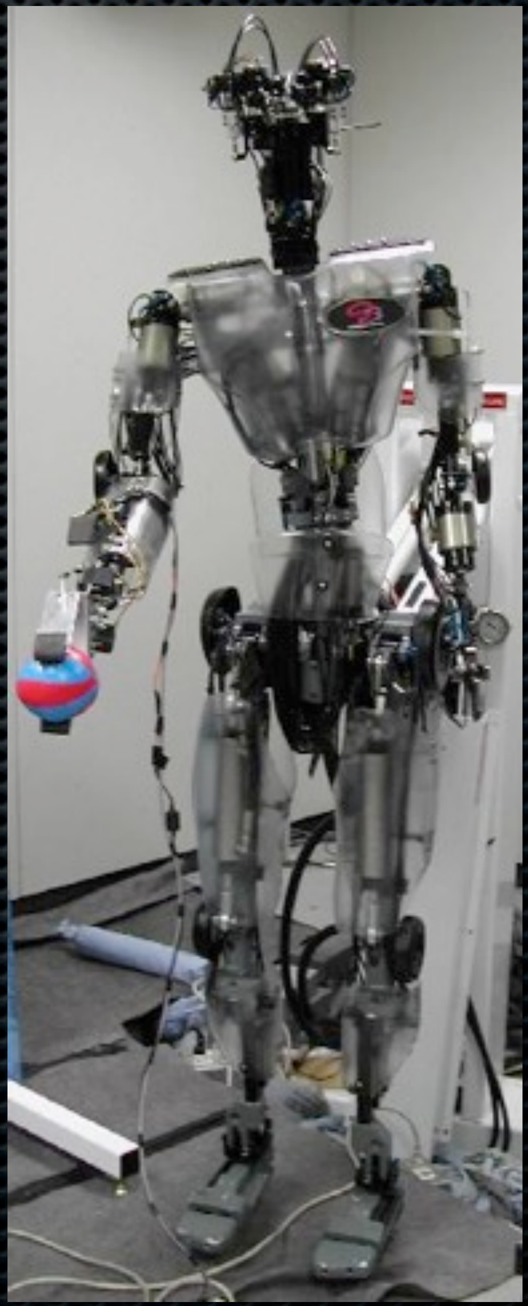
Velocity

Acceleration

Types of Information in Human Coaching

- Evidence that people use kinematic planning:
  - Kinematic trajectory planning in the **parietal cortex** (Kalaska, 1991)
  - Inverse dynamics models found in the **cerebellum** (Schweighofer et al.,1998)
  - Motor equivalence (Kelso, 1982; Bernstein, 1967)

Formalisms used in Humanoid Robot Coaching



Summary of Formalisms useful in the robot domain:

Transmit Information by:

Demonstration	**Verbal communication**
Performance, guiding	kinematic instructions

Useful coaching formalism applicable to humanoid robot domain:

- New motor knowledge (patterns of coordination)
- Focus attention on relevant task features for learning of critical task aspects
- Assign priorities among goals
- Gives specific feedback to improve performance
- Iteratively define characteristics of success

Timing of commands is important, as is the tight coupling of performance, evaluation and instruction.

Humanoid Robot Coaching

Adaptations to humanoid robot coaching system:

New motor knowledge by **demonstration**: imitation and physical guiding

Vocabulary for coaching instructions: kinematic commands used for motor skills

Transformation functions containing domain-specific knowledge to effect specific changes to a motor skill

Ability to **focus attention** on specific parts of a behavioral for refinement: body and time segmentation

A student-initiated **dialogue** to resolve ambiguities

Constraint: real-time interactive system that preserves tight coupling found in human coaching among effort, evaluation and guidance

Real-time full-body Imitation

Imitation as a means of learning from demonstration

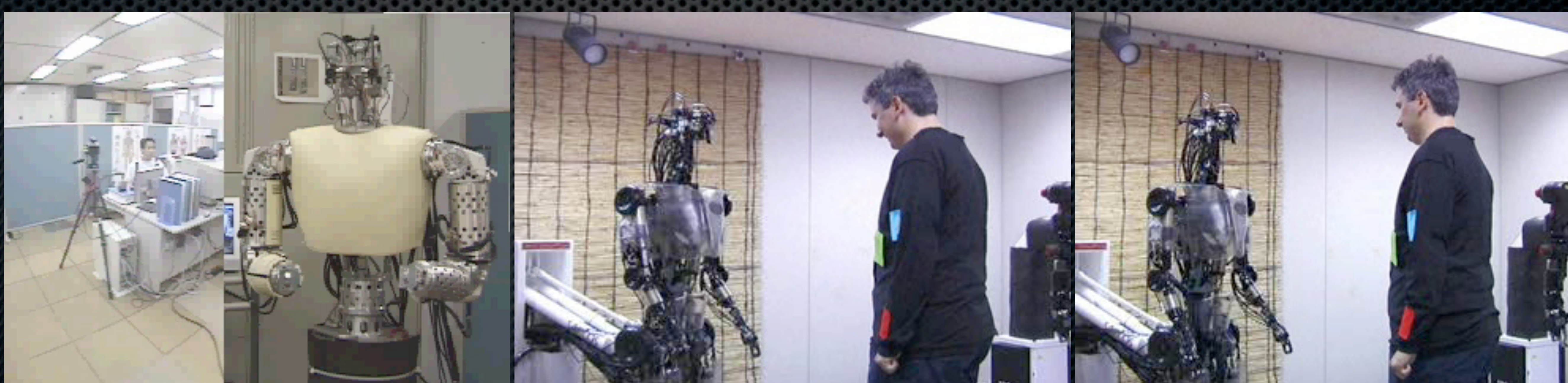
efficient way to acquire and modify skills

retaining human characteristics of behavior

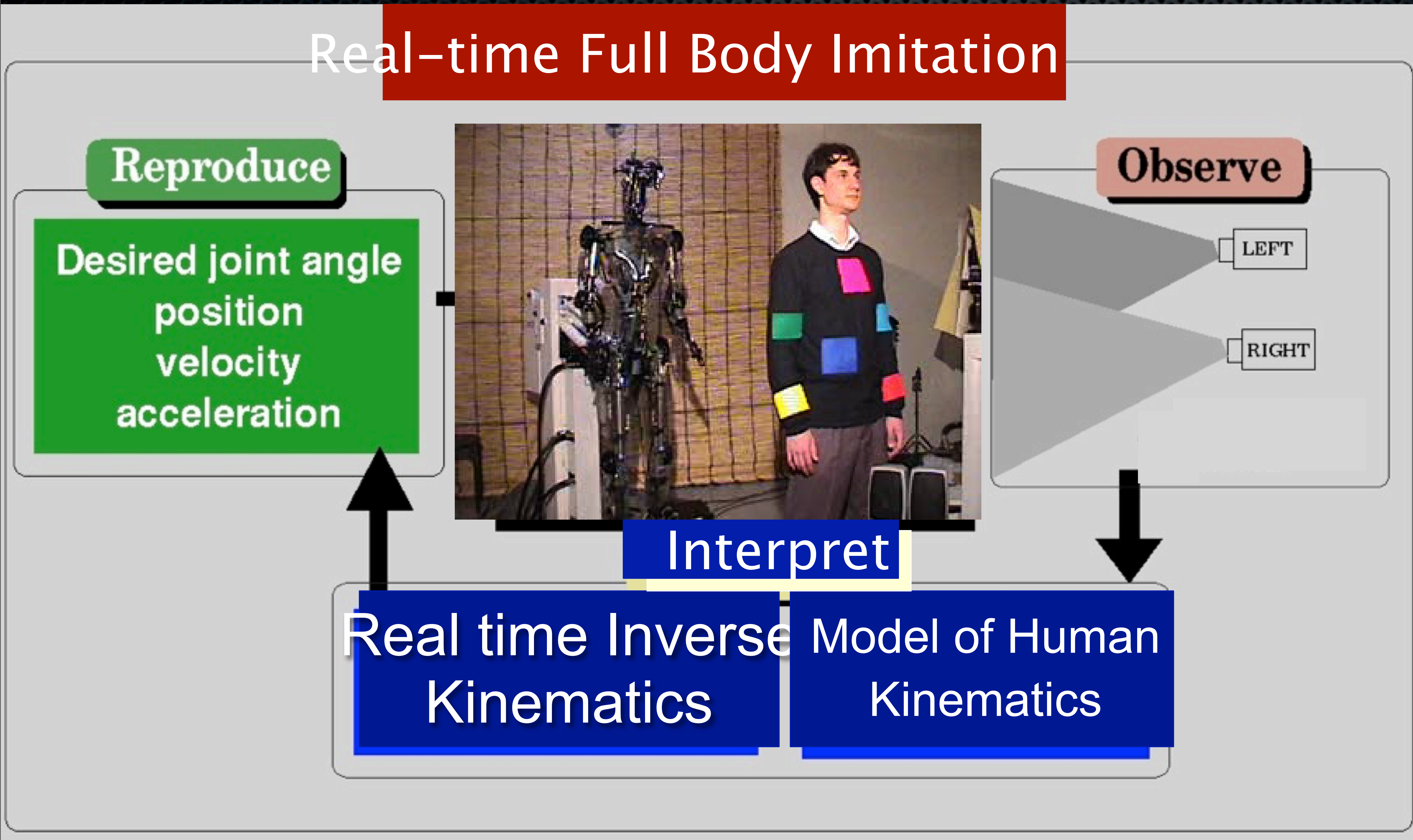
Provide interaction in a natural context

Strictly low-level imitation: the only goal is to match the movement of the coach or teacher as closely as possible

Use this to bootstrap new behaviors in real time in our coaching system



Demonstration: Approach to Imitation



Humanoid Robot Coaching System

- **New motor knowledge** by **demonstration**: ***imitation*** and **physical *guiding***
- Full-body real-time imitation method to bootstrap behaviors.
- Guide the robot through a motion by lowering gains and capturing joint angles

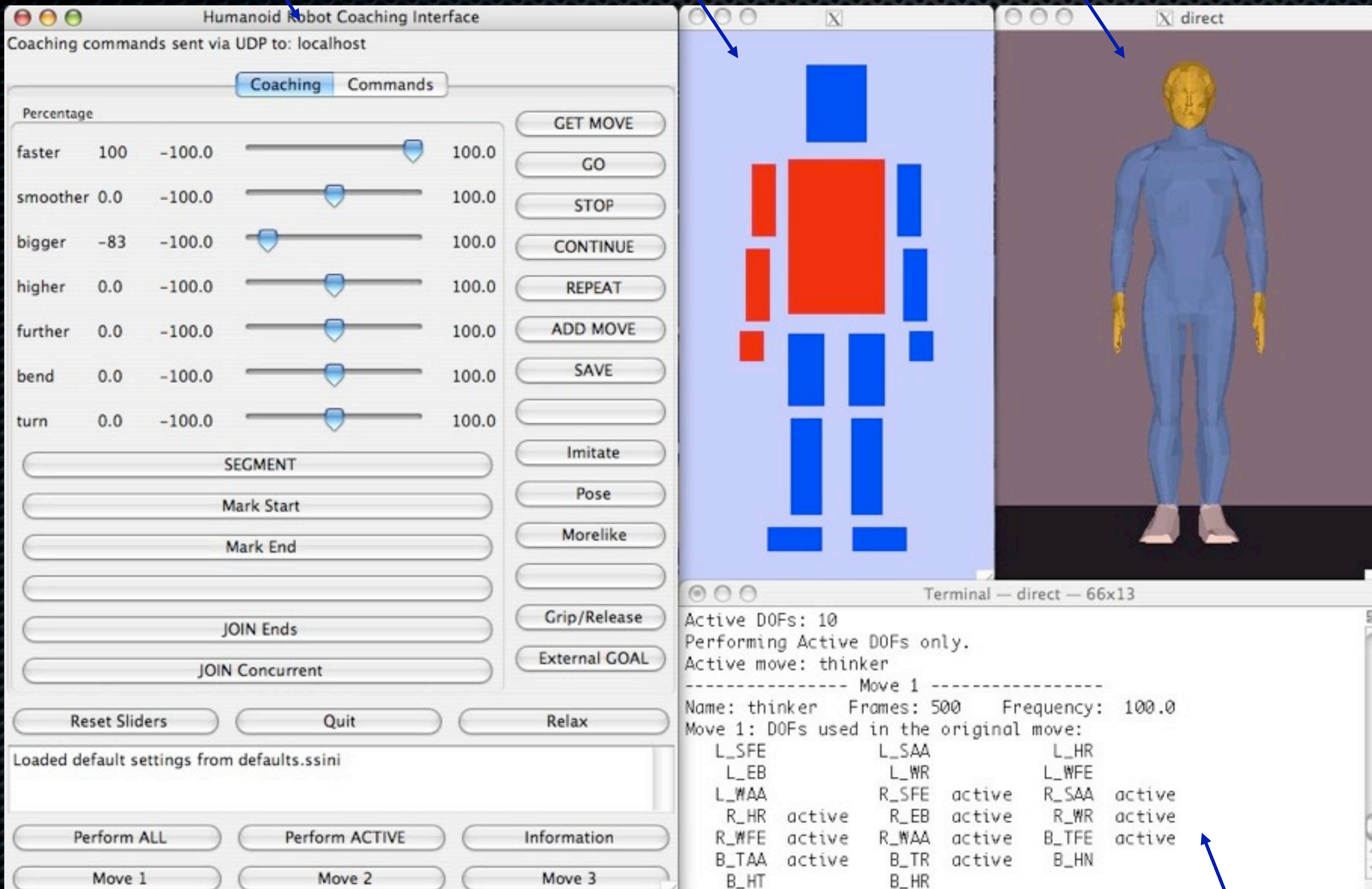
Related Work

- Nakatani, Suzuki & Hashimoto, “Subjective-evaluation oriented teaching scheme for a biped humanoid robot”, Conf. HR, 2003.
- Nicolescu & Mataric, “Natural methods for robot task learning: Instructive demonstrations, generalization and practice.”, Conf. on AAMA 2003.
- Takagi, “Interactive evolutionary computation: Fusion of the capabilities of EC optimization and human evaluation”, Proc. of IEEE, vol.89, no.9, 2001.
- Kuroki et al., “Motion creating system for a small biped entertainment robot”, IROS 2003
- Interactive Evolutionary Computation (IEC) is a technique that evolutionary computation consisting of genetic algorithms (GA), evolutionary strategy (ES), evolutionary programming (EP), and genetic programming (GP) optimizes the target system based human subjective evaluation.

Classic Interface

2D Body Part Model

3D Humanoid Model

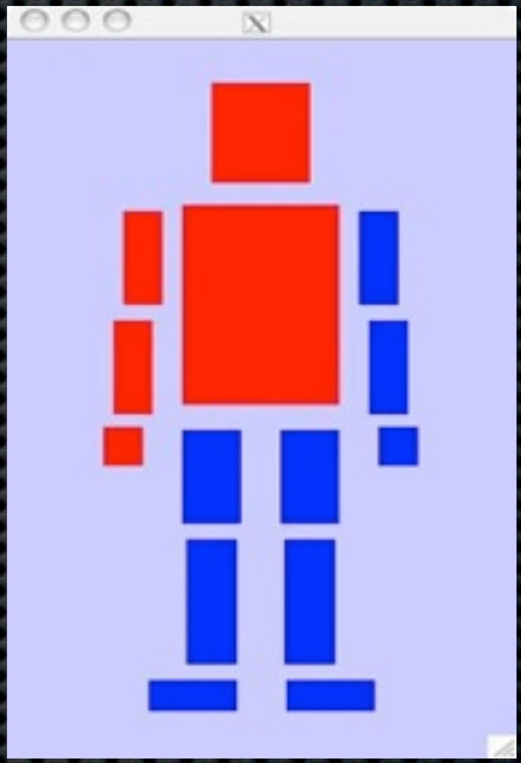


Interactive Text Window

Humanoid Robot Coaching System

- Vocabulary for coaching instructions:
  - Reflects verbal instructions coaches commonly use. These commands center around **kinematic** descriptions of motion, such as higher, bend, and bigger used often in teaching motor skills.
- These domain-specific commands comprise the system primitives
- Vocabulary also used to describe body

Humanoid Robot Coaching System



- Ability to focus attention in body and time
  - body space: concentrate and refine one part of the movement (arms, leave the legs for later)
  - time: segment the movements into sequences of smaller movements (split, join ends, join concurrent)

Humanoid Robot Coaching System

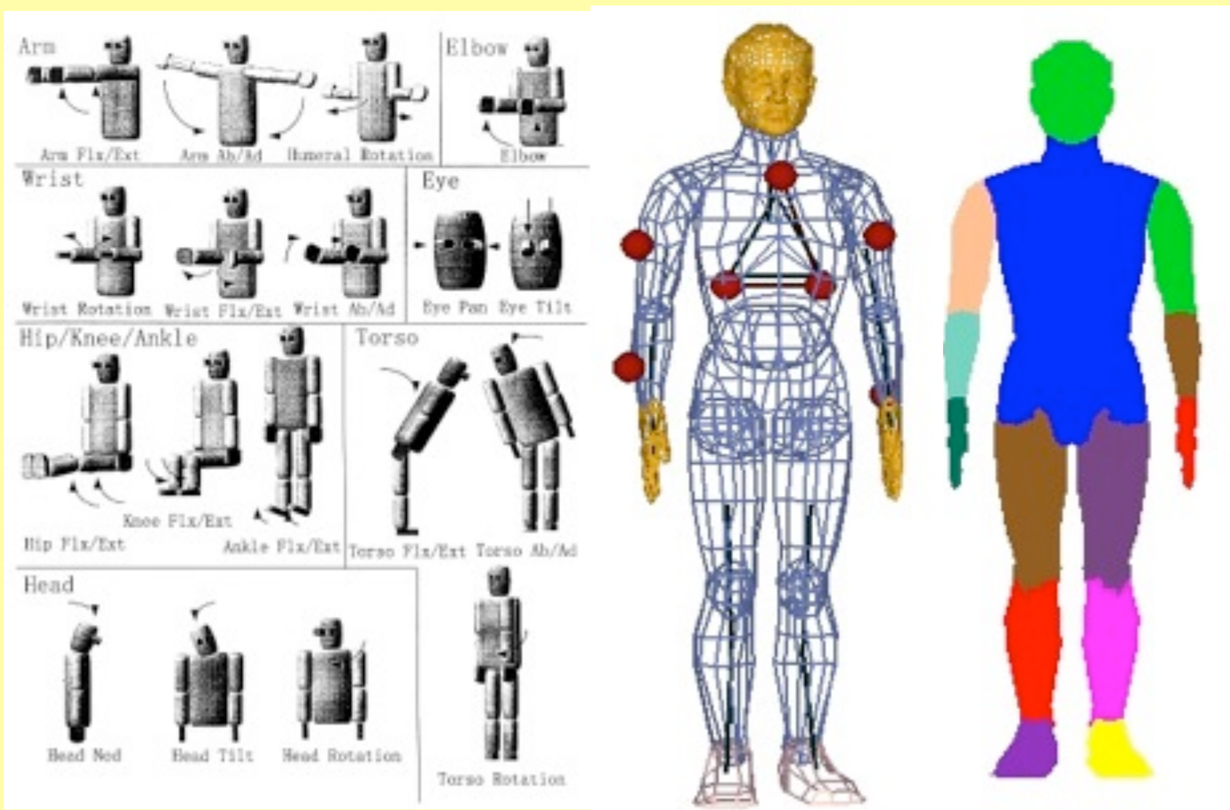
- Transformation functions containing domain-specific knowledge to effect specific changes to a motor skill
- A ***TF*** is comprised of a label, the coaching command that invokes it, and a set of criteria that defines the high level command in terms of low level behavioral criteria. Label and criteria comprise a function that ultimately effects changes to the appropriate behavioral parameters

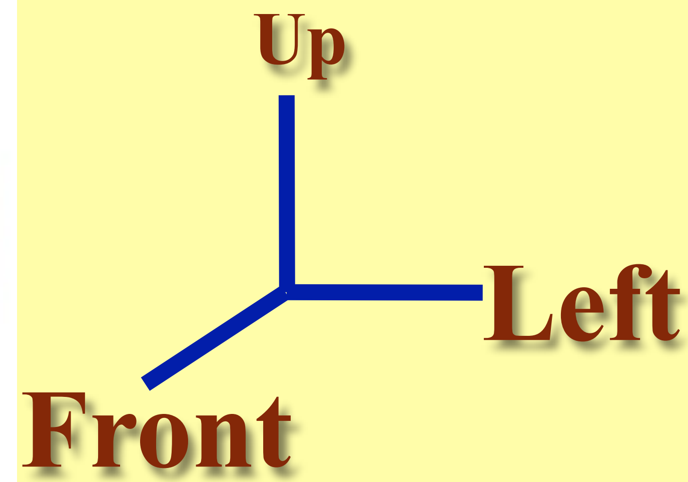
Using knowledge to find solutions

- Need knowledge relevant to behavior domain to establish criteria for transformation functions.
- We seek a minimal knowledge representation that affords the robot the same type of understanding of its body and the world as an infant has.
  - body, connectivity (reaching, torso may help extend the arm)
  - world (external objects exist, my body is somewhere in world)
- In addition, we have domain level knowledge of common motion descriptors like ***higher, bigger, faster.***

Knowledge

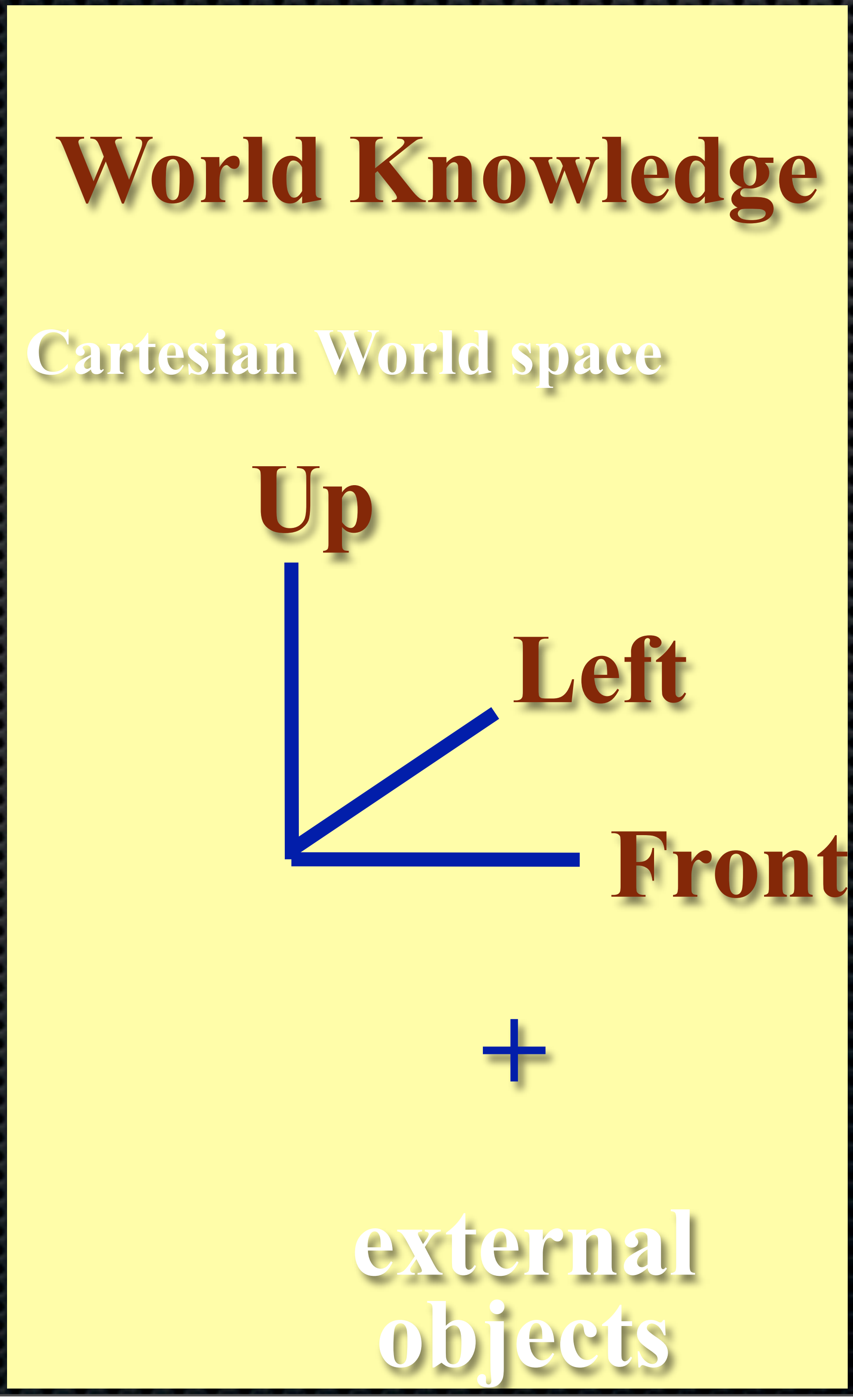
Body Knowledge





Cartesian body space

Knowledge


World Knowledge
Cartesian World space
Up
Left
Front
+
external
objects

Representations

Points **P,**

Movements **M,**

Transformations **T,**

$\mathbf{M}(arg_1, arg_2, \ldots, arg_n)$

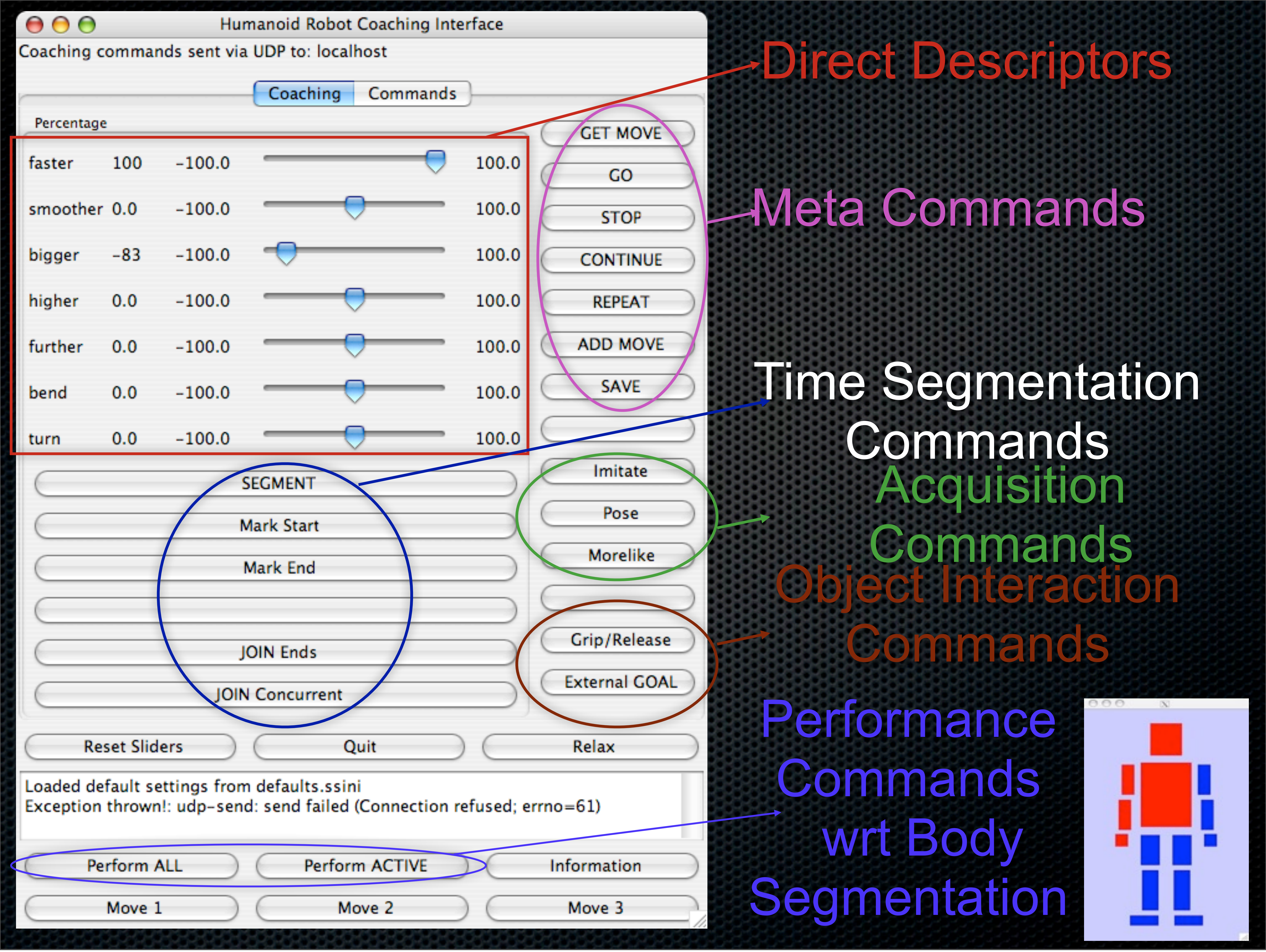
$\mathbf{M}_i \xrightarrow{\mathbf{T}} \mathbf{M}_j$ $\mathbf{T}(arg_1, arg_2, \ldots, arg_m)$

Labels **Word or Phrase**

Word	word	word	phrase	phrase
↓	↓	↓	↓	↓
P	**M**	**T**	**M**(args)	$\mathbf{M}_i \xrightarrow{\mathbf{T}_2} \mathbf{M}_j$, $i = 1$

Using knowledge to find solutions

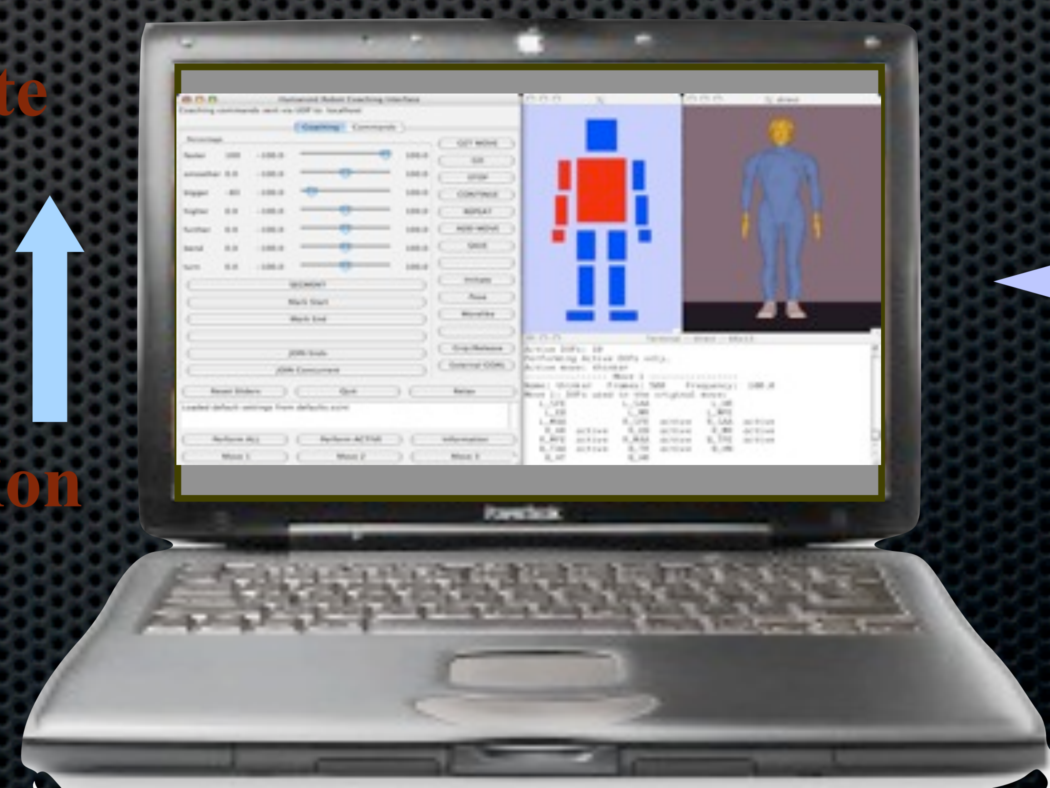
- We can exploit this knowledge to enable the robot to find its own solutions in response to commands.
- Robot determines which DOFs would help with a ***higher*** command.
    - candidate DOFs are determined
    - each is tested with a virtual move using forward kinematics always starting from the current position
    - the change in position is compared to the criteria for the ***TF***
    - if it matches, robot suggests using this DOF
    - may make other suggestions knowing its connectivity


Humanoid Robot Coaching Interface
Coaching commands sent via UDP to: localhost
Coaching
Commands
Percentage
faster 100 -100.0 100.0
smoother 0.0 -100.0 100.0
bigger -83 -100.0 100.0
higher 0.0 -100.0 100.0
further 0.0 -100.0 100.0
bend 0.0 -100.0 100.0
turn 0.0 -100.0 100.0
GET MOVE
GO
STOP
CONTINUE
REPEAT
ADD MOVE
SAVE
SEGMENT
Mark Start
Mark End
JOIN Ends
JOIN Concurrent
Imitate
Pose
Morelike
Grip/Release
External GOAL
Reset Sliders
Quit
Relax
Loaded default settings from defaults.ssini
Exception thrown!: udp-send: send failed (Connection refused; errno=61)
Perform ALL
Perform ACTIVE
Information
Move 1
Move 2
Move 3
Direct Descriptors
Meta Commands
Time Segmentation Commands
Acquisition Commands
Object Interaction Commands
Performance Commands wrt Body Segmentation

Low Level System Implementation

Robot State

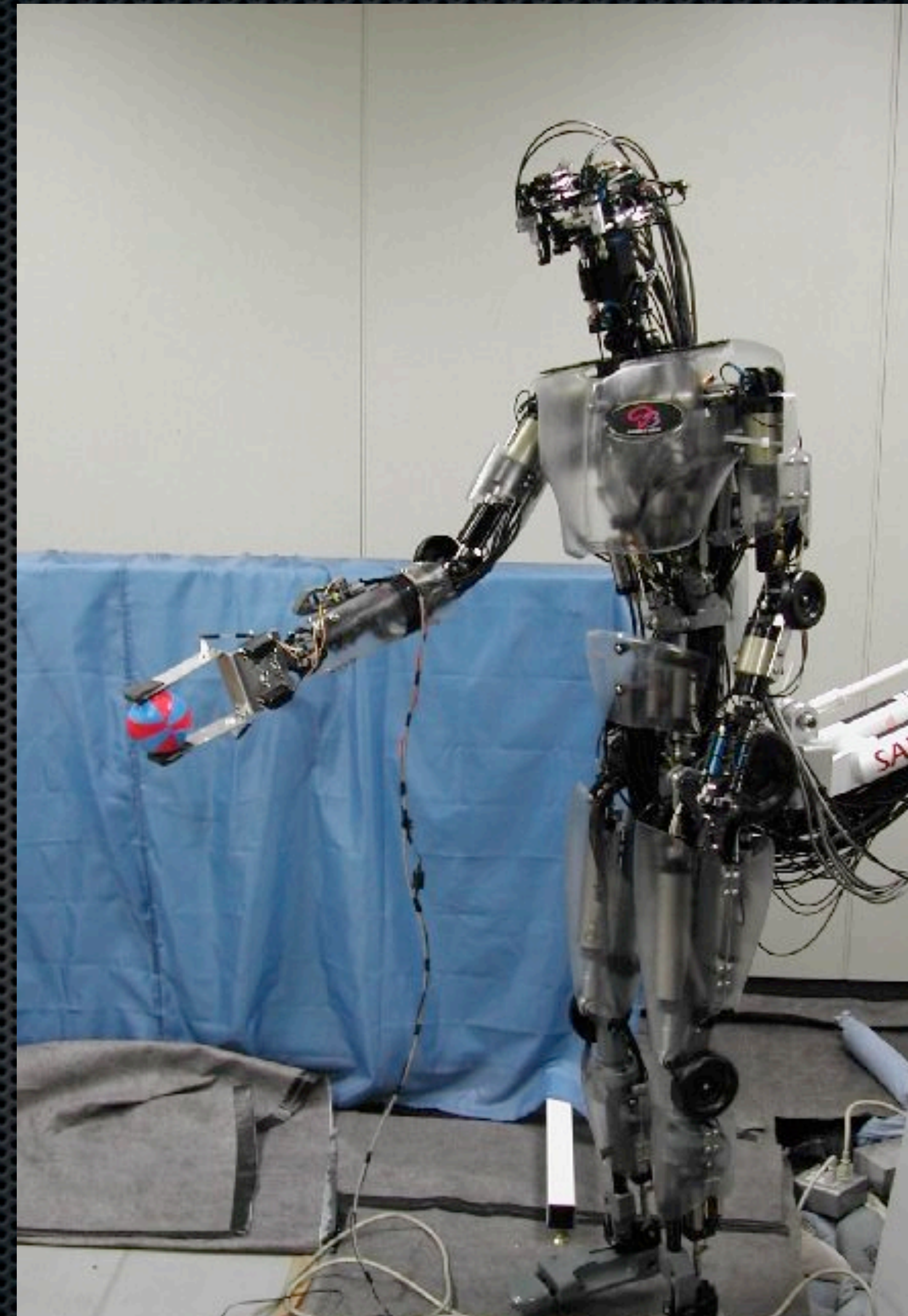
Joint Position, Velocity, Acceleration via UDP



3D Vision data via UDP

Acquiring a throwing movement

Add a gripper



Seeding the throwing movement from demonstration

Experimental Parameters

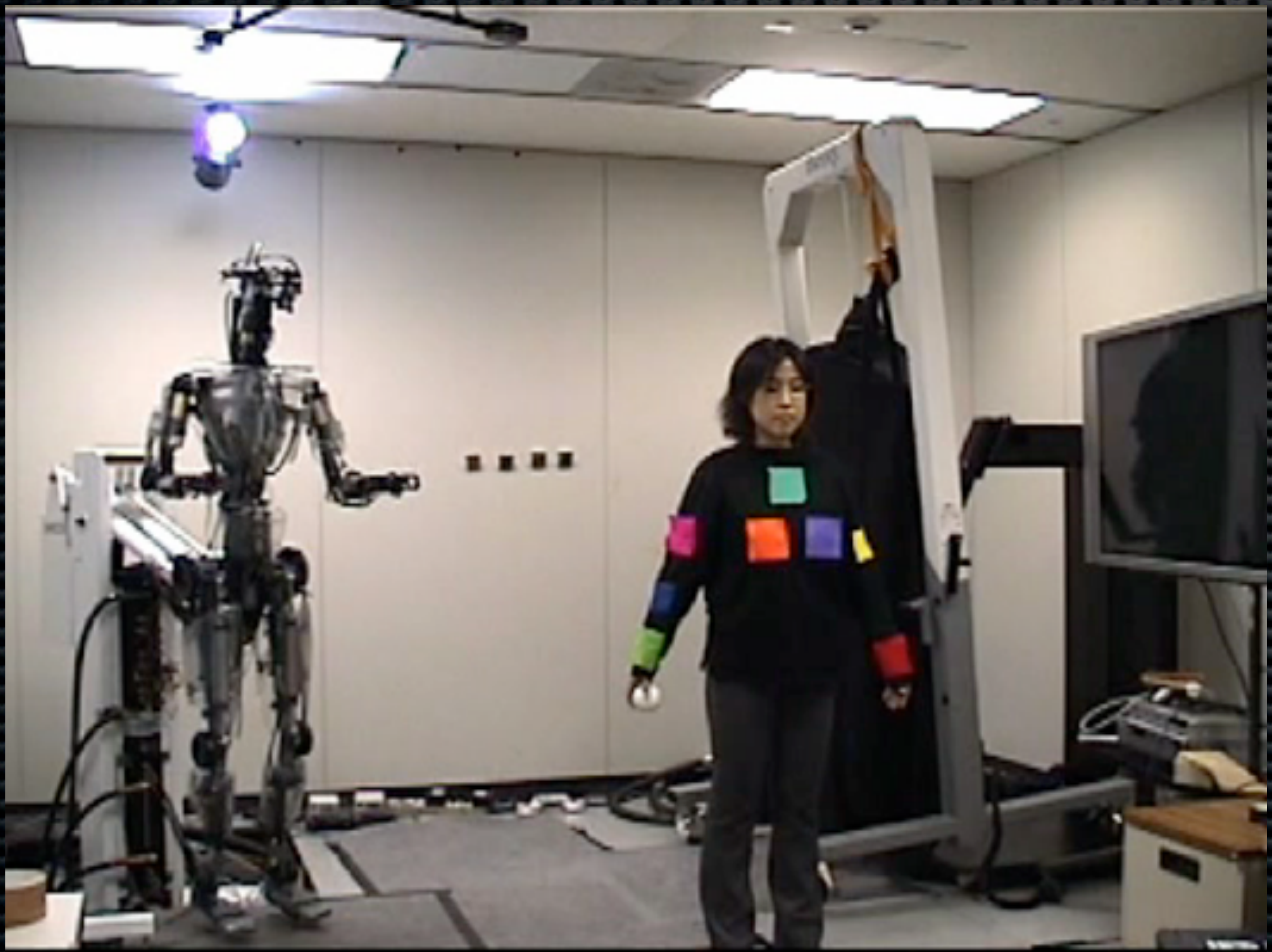
Acquires new motor knowledge about task from coach's demonstration
Coach gives specific feedback to improve performance

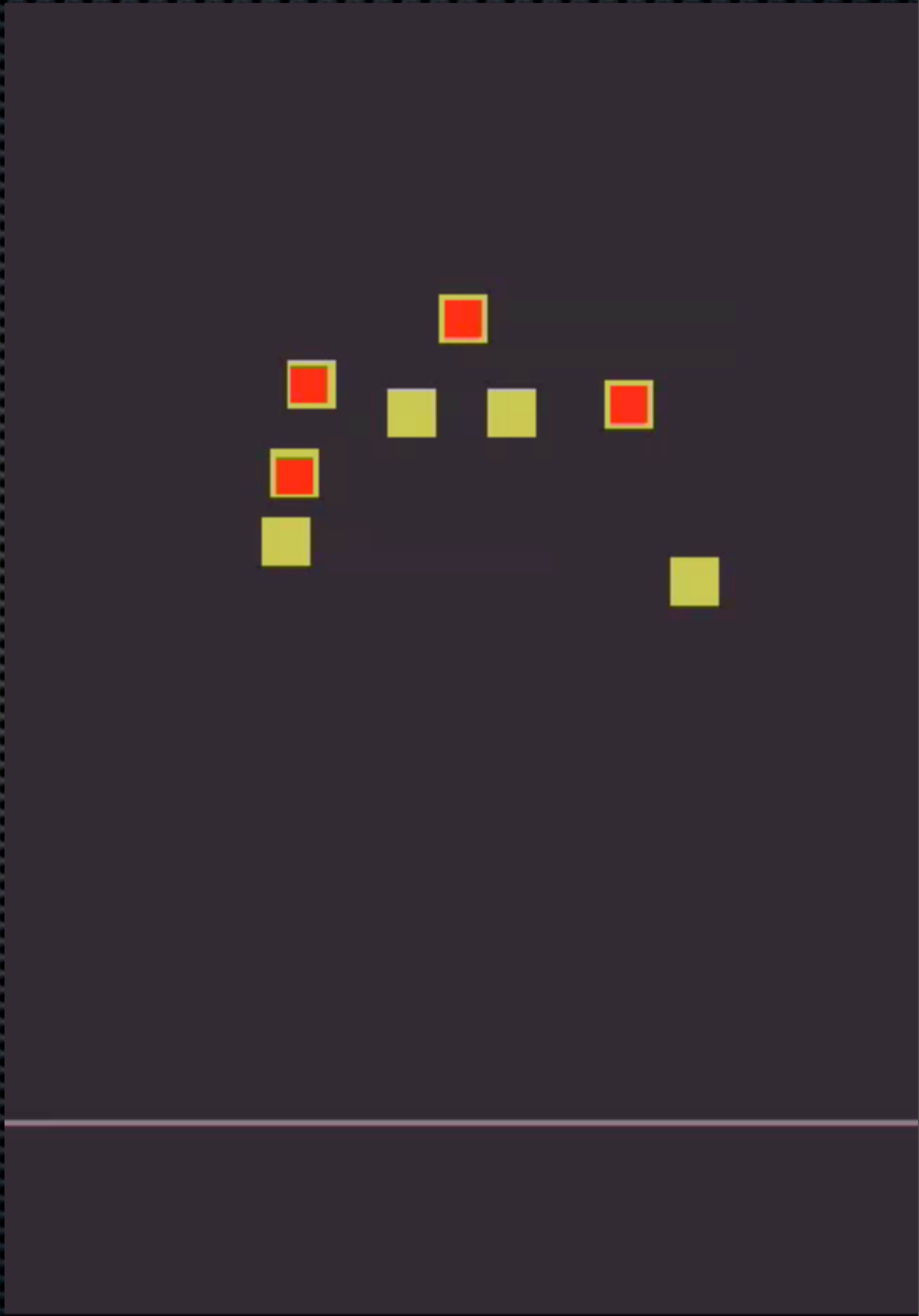
puparmr_fe	60 8
puparmr_aa	14 3 1 2 3
plowarmr_fe	2 1 4
puparml_fe	1 1 5
puparml_aa	5 1 6
plowarml_fe	4 2 7 8

pchest_fe
pchest_aa
pchest_r

3 points for world pos and orientation mapping

Experimental Parameters

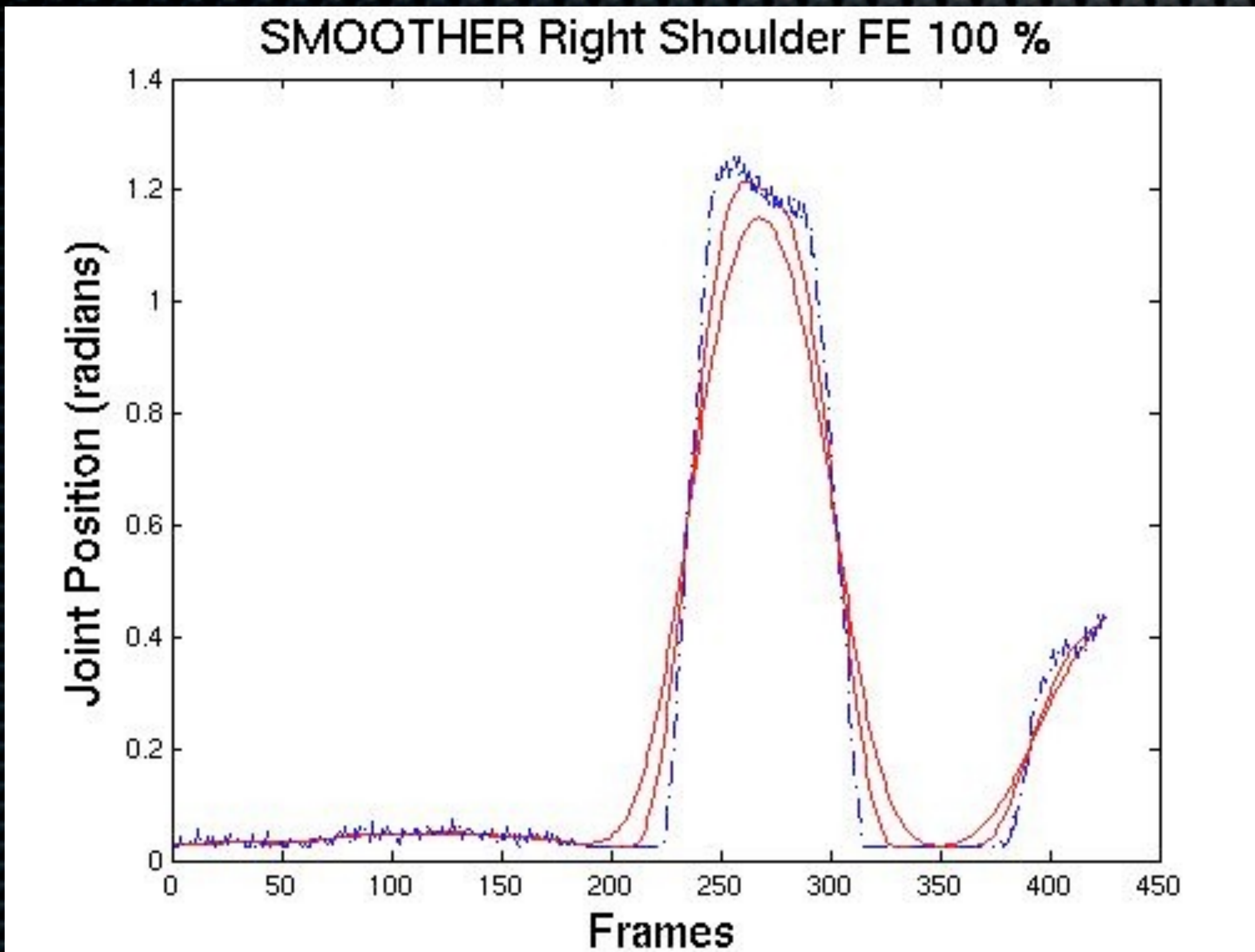


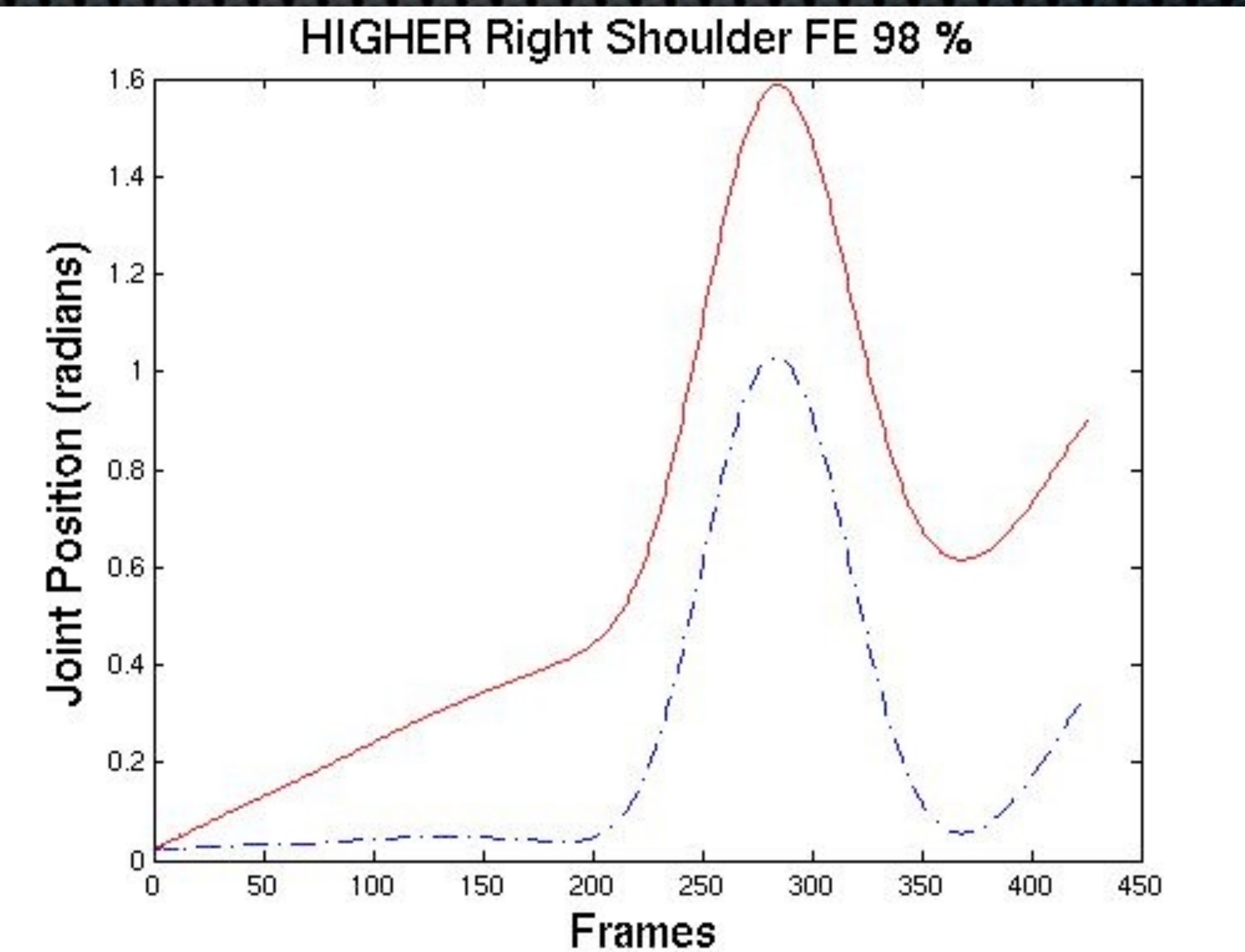


Seeding the throwing movement from demonstration

Coaching:
An Approach to Efficiently and Intuitively Create Humanoid Robot Behaviors

Transformation Functions





Original and modified trajectories for two iterations of the *smoother* transformation implemented with a moving average filter.

Original and modified trajectories showing modification by the *higher* transformation function after using *smoother*.

Comments

- Coaching paradigm: robot acquires motor skills with the aid of a human coach

Modeled on human-skill transfer between a coach and student

Reduce time and ease of creating robot behaviors by constraining
the solution space for a given behavior.

Coaching does this by providing critical evaluation and guidance to reach a
correct solution faster than can realized with no guidance.

Coaching affords:
High level control of complex robots
Eliminates need to program each behavior
Affords flexibility in changing goals or focus of attention during a behavior
Enables non-specialists to participate more fully in creating robot behaviors

Comments

- Coaching does not obviate the need for low level control algorithms
- Instead, we want to look at potential role of introducing interactive high-level instruction and interactive goal specification used so successfully by people in improving the overall efficiency of creating new robot behaviors.

Future (and Current) Work

- Remember and re-use strategies
  - recognizing which primitives are useful in a given situation
  - representation of task and goal in order to recognize similar tasks
- Learning Transformation functions
  - adding new primitives to the system without programming

The End

- Thank you for your attention….