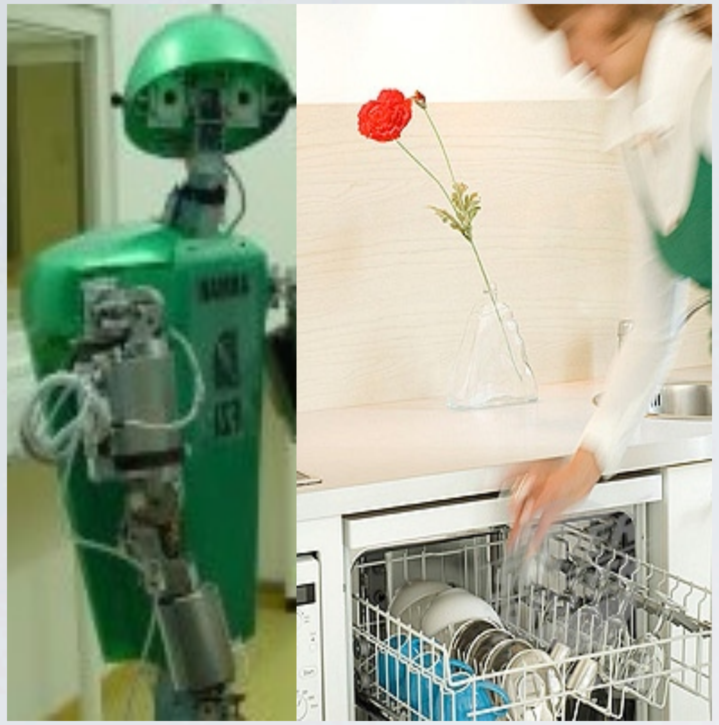
ON HUMAN ACTION

Volker Krüger
Dept. of Mechanical and Production Engineering
Aalborg University
vok@m-tech.aau.dk

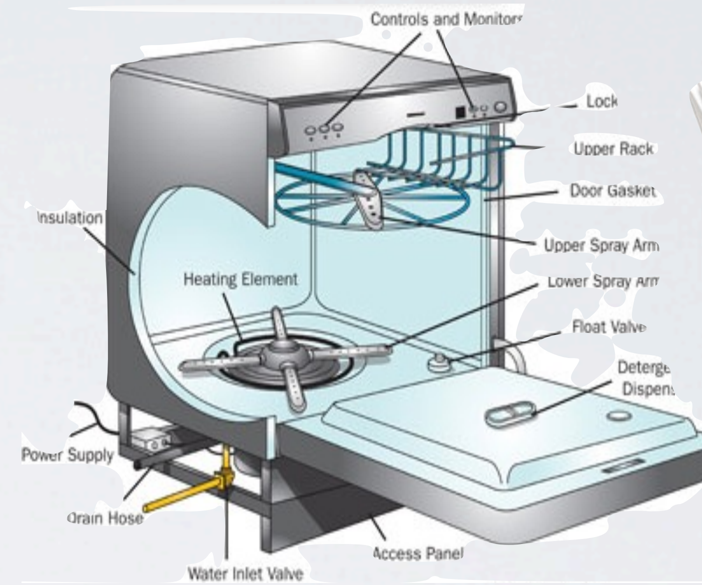
Institut for
Mekanik og Produktion

AALBORG UNIVERSITET

Cleaning the Kitchen



Observation

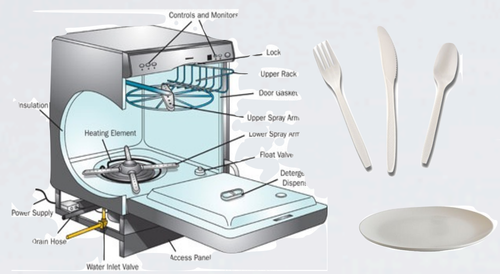


World Model

Reproduction/Recognition

Cleaning the Kitchen

Observation



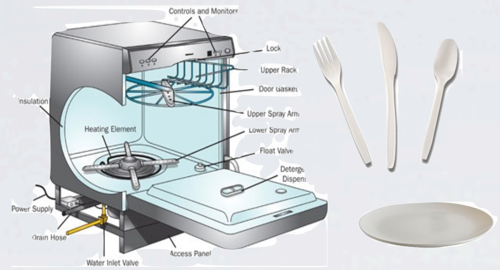
World Model

Reproduction/Recognition

What is the action?

Observation

Cleaning the Kitchen



World Model

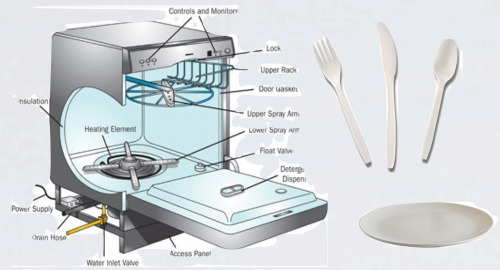
Reproduction/Recognition

What is the action?

Grasping a plate?

Cleaning the Kitchen

Observation



World Model

Reproduction/Recognition

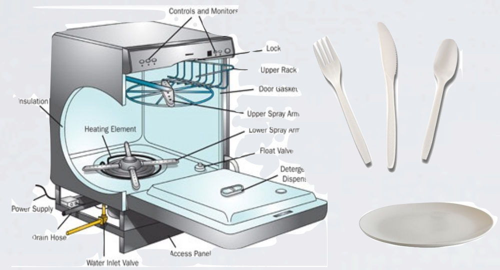
What is the action?

Grasping a plate?

Putting plates upright?

Cleaning the Kitchen


Observation


World Model

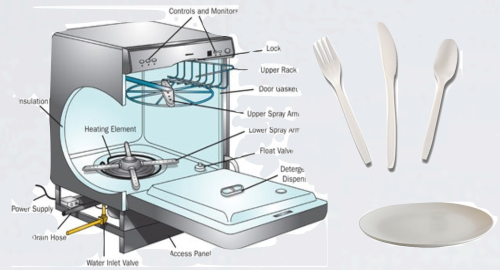

Reproduction/Recognition

What is the action?

Grasping a plate?
Putting plates upright?
Removing plates from the table?

Cleaning the Kitchen

Observation



World Model

Reproduction/Recognition

What is the action?

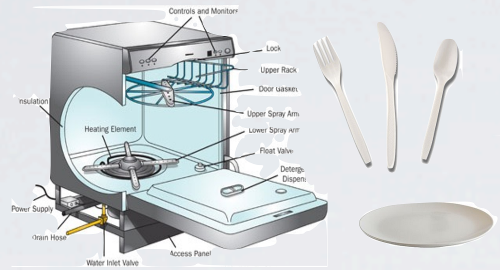
Grasping a plate?

Putting plates upright?

Removing plates from the table?

Filling the dish washer?

Observation



World Model

Reproduction/Recognition

Cleaning the Kitchen

What is the action?

Grasping a plate?

Putting plates upright?

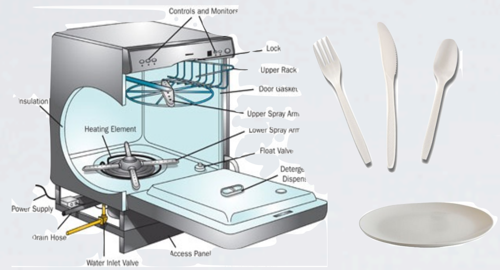
Removing plates from the table?

Filling the dish washer?

Cleaning the kitchen?

Cleaning the Kitchen

Observation



World Model

Reproduction/Recognition

What is the action?

Grasping a plate?

Putting plates upright?

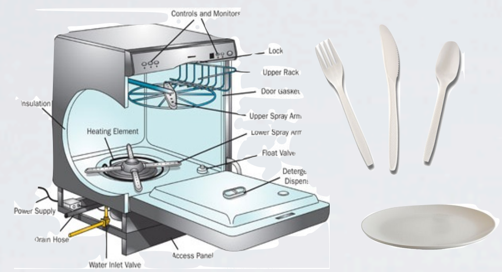
Removing plates from the table?

Filling the dish washer?

Cleaning the kitchen?

So what does it mean to *understand* the meaning of an action?

Observation



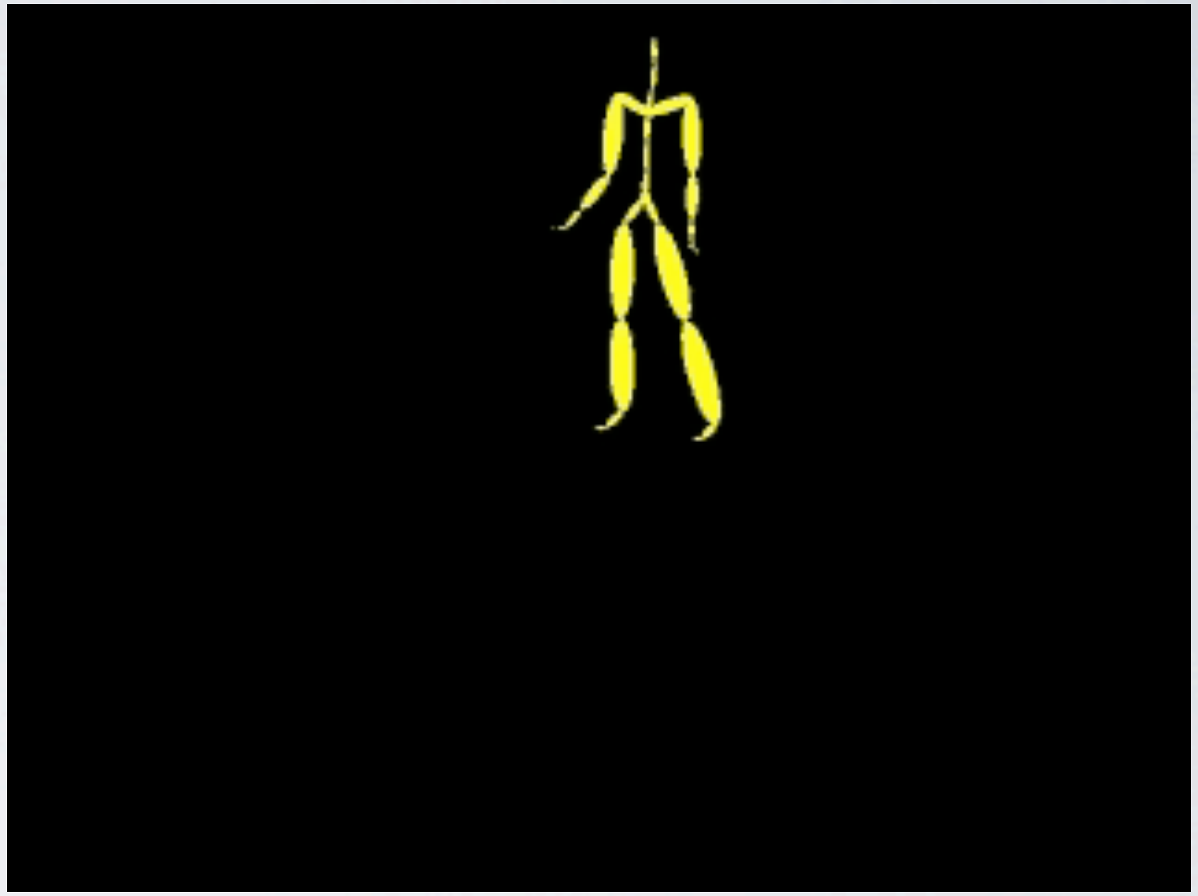
World Model

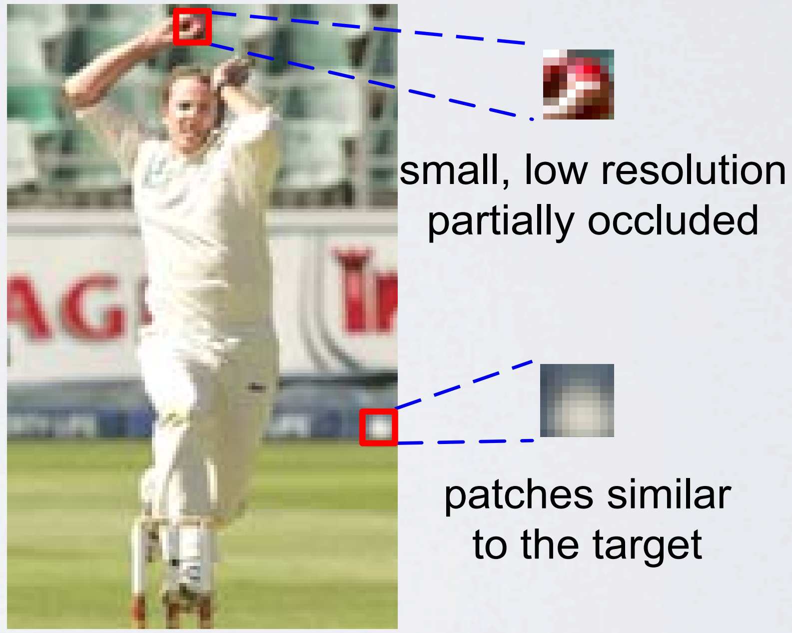
Reproduction/Recognition

Cleaning the Kitchen

The meaning of an action is the state change that the physical movement of an actor causes to the world state space.
That can be on different levels of abstraction. At least, this is the goal.

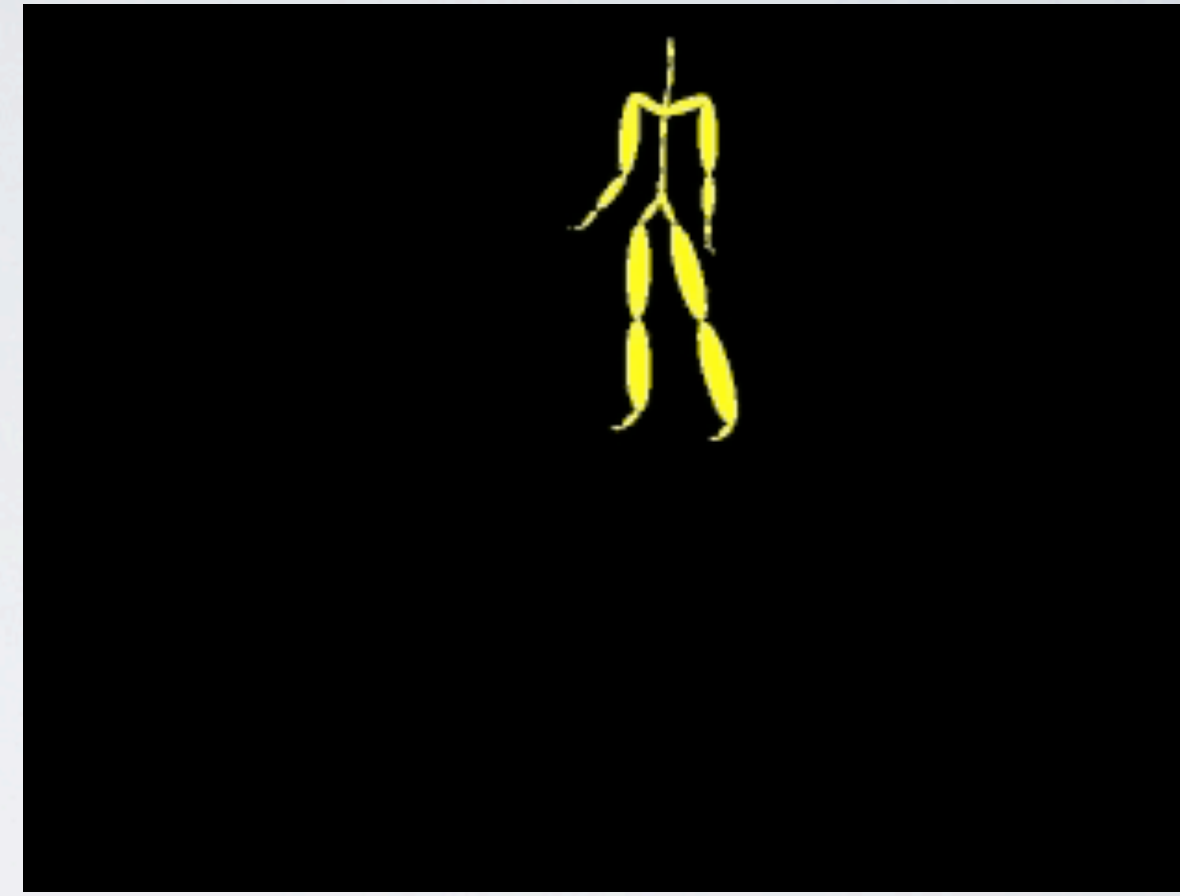
- What is the person doing?

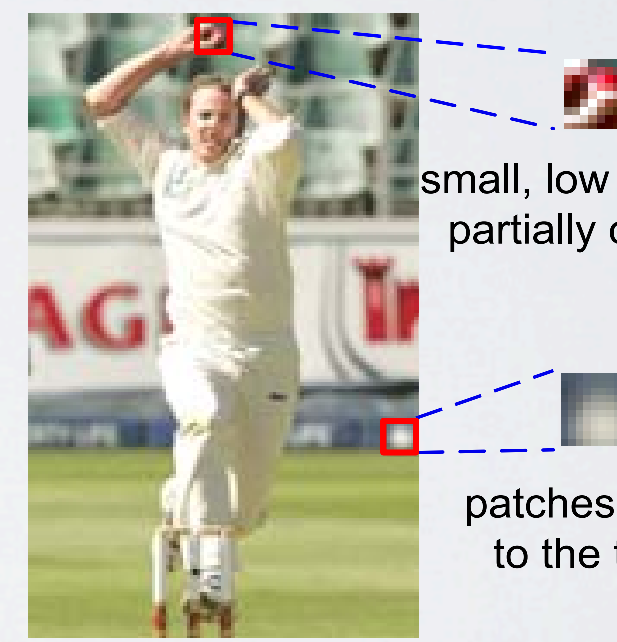




Li Fei-Fei, CVPR10

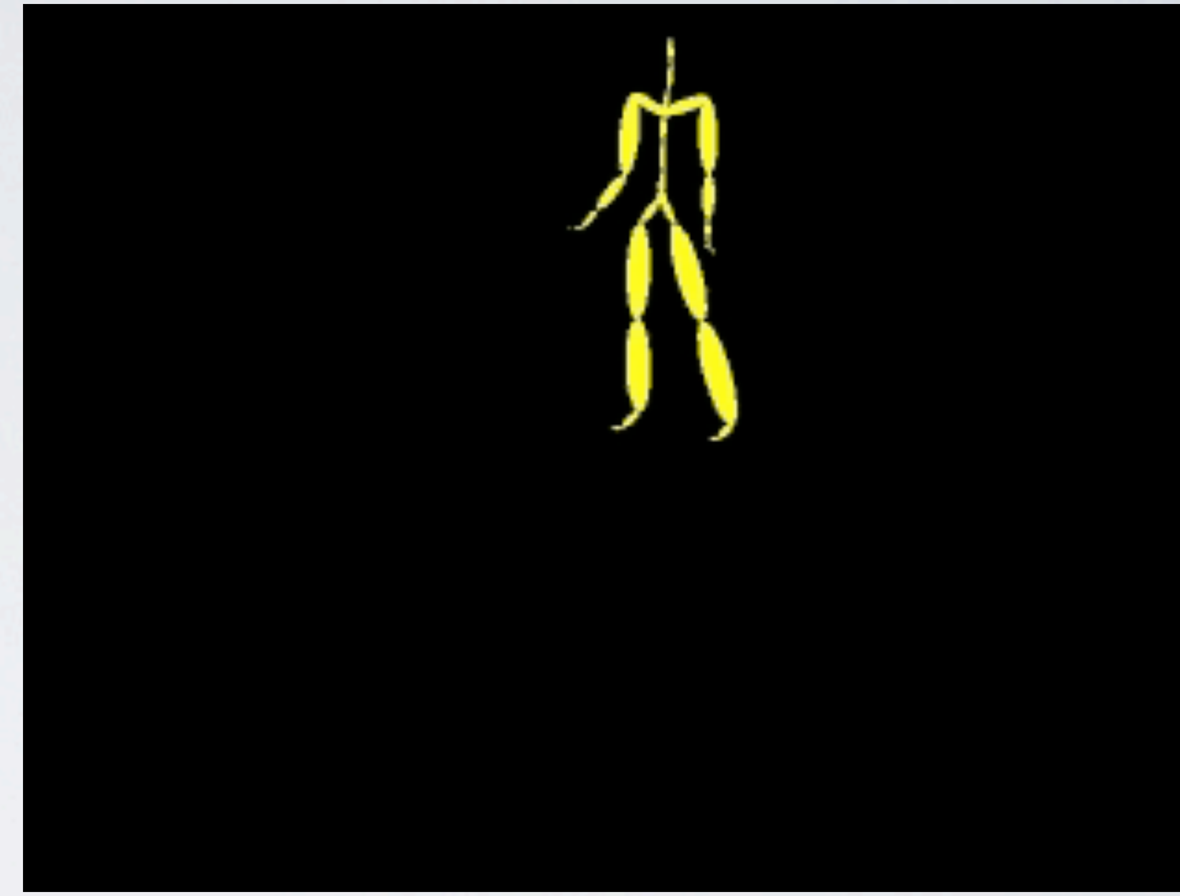
- What is the person doing?
- Objects and actions are intertwined

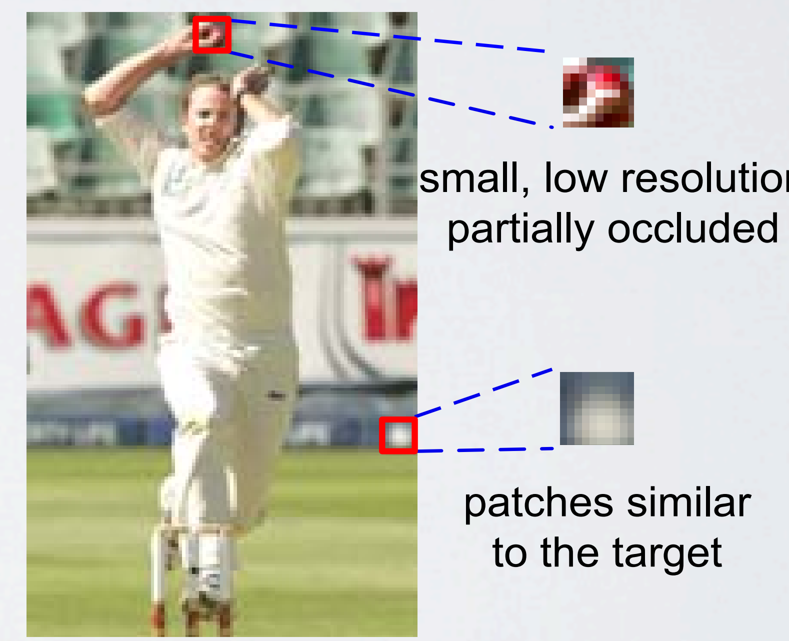




Li Fei-Fei, CVPR10

- What is the person doing?
- Objects and actions are intertwined
- Objects prime actions, actions prime objects





Li Fei-Fei, CVPR10

The world is perceived not only in terms of object shapes and spatial relationships but also in terms of object possibilities for action (***affordances***). **perception drives action**.

- **Gibson, J.J.** (1977). The theory of affordances. In R. Shaw & J. Bransford (eds.), Perceiving, Acting and Knowing. Hillsdale, NJ: Erlbaum.
- **Norman, D.** (1988). The Psychology of Everyday Things. New York, Basic Books, pp. 87-92.
- **Humphreys, G. *et al.*** The interaction of attention and action: From seeing action to acting on perception. British Journal of Psychology (2010), 101, 185–206

perception and action share the same symbolic structure

- Gallese et al. ”Action Recognition in the premotor cortex”, Brain, vol. 119, no. 2, 1996.
- Nishitani et al. ”Broca’s Region: From Action to Language” Physiology, vol. 20, 2005.
- Rizzolatti et al. ”Neurophysiological Mechanisms Underlying the Unterstanding and Imitation of Action” Nature Reviews, vol 2, 2001.
- **Newtson**: “The Objective Basis of Behavior Units”, Journal of Personality and Social Psychology, vol 35(12), 1977.

perception and action share the same symbolic structure
spoken language and visible movements use same cognitive substrate

- Gallese et al. "Action Recognition in the premotor cortex", Brain, vol. 119, no. 2, 1996.
- Nishitani et al. "Broca's Region: From Action to Language" Physiology, vol. 20, 2005.
- Rizzolatti et al. "Neurophysiological Mechanisms Underlying the Unterstanding and Imitation of Action" Nature Reviews, vol 2, 2001.
- **Newtson**: "The Objective Basis of Behavior Units", Journal of Personality and Social Psychology, vol 35(12), 1977.

OBJECT ACTION COMPLEXES (OACS)

- Objects and Actions are inseparably intertwined.

paco | plus

Institut for
Mekanik og Produktion

AALBORG UNIVERSITET

OBJECT ACTION COMPLEXES (OACS)

- Objects and Actions are inseparably intertwined.
- Categories are determined (and also limited) by the action an agent can perform and by the attributes of the world it can perceive;

paco | plus

Institut for
Mekanik og Produktion

AALBORG UNIVERSITET

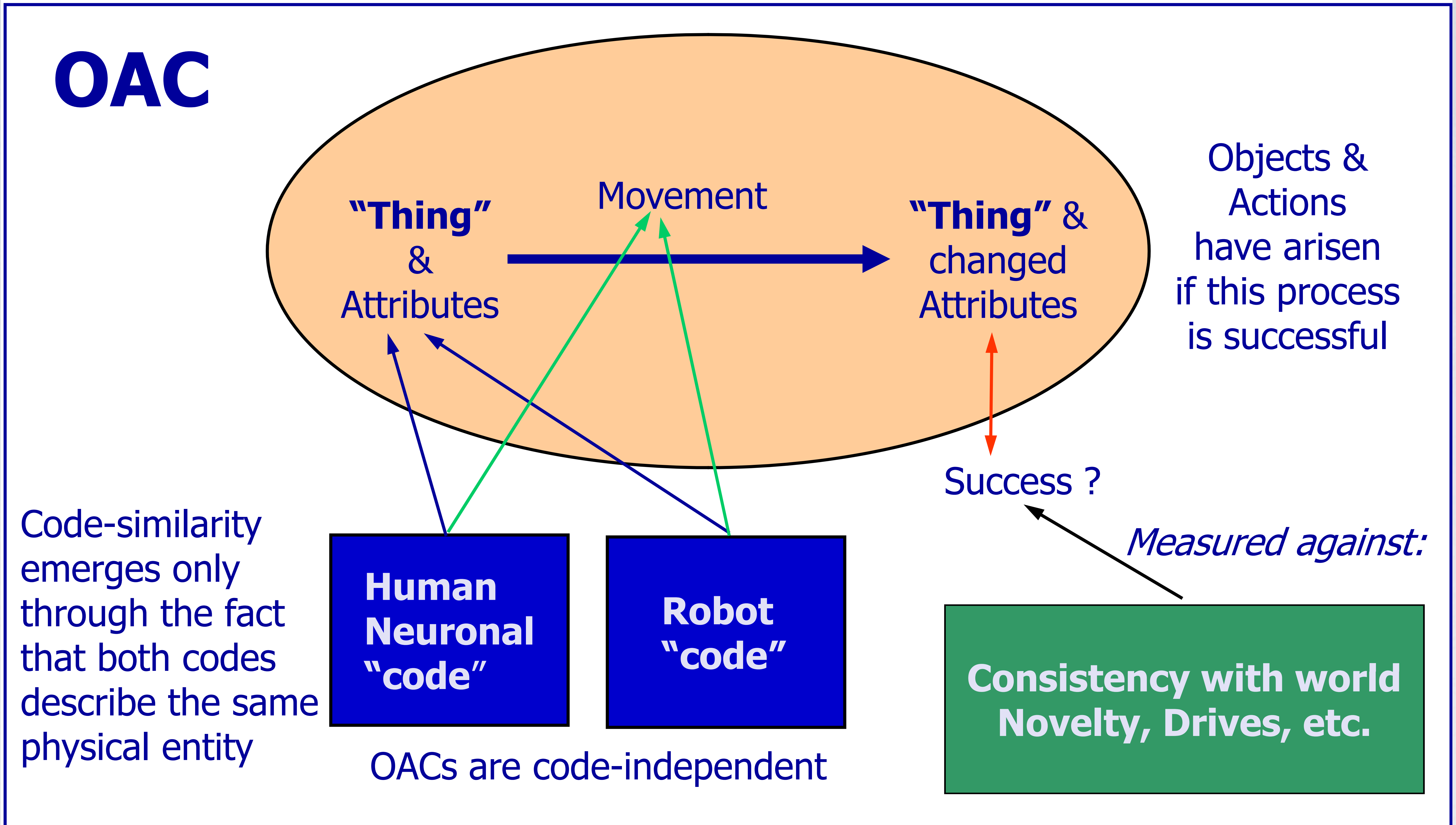
OBJECT ACTION COMPLEXES (OACS)

- Objects and Actions are inseparably intertwined.
- Categories are determined (and also limited) by the action an agent can perform and by the attributes of the world it can perceive;
- Entities "things" in the world of a robot (or a human) will only become semantically useful "objects" through the action that the agent can/will perform on them.

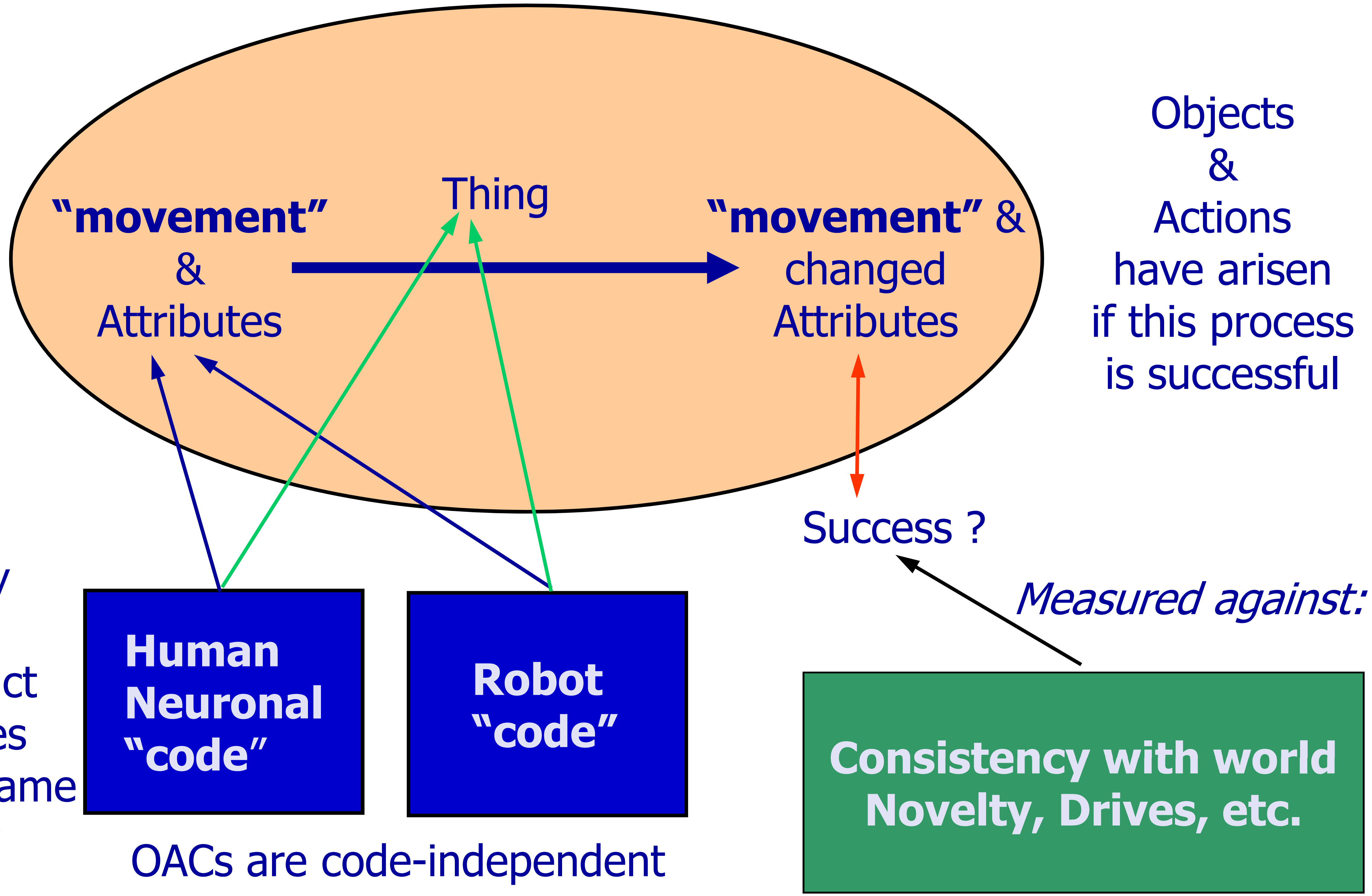
paco | plus

Institut for
Mekanik og Produktion

AALBORG UNIVERSITET

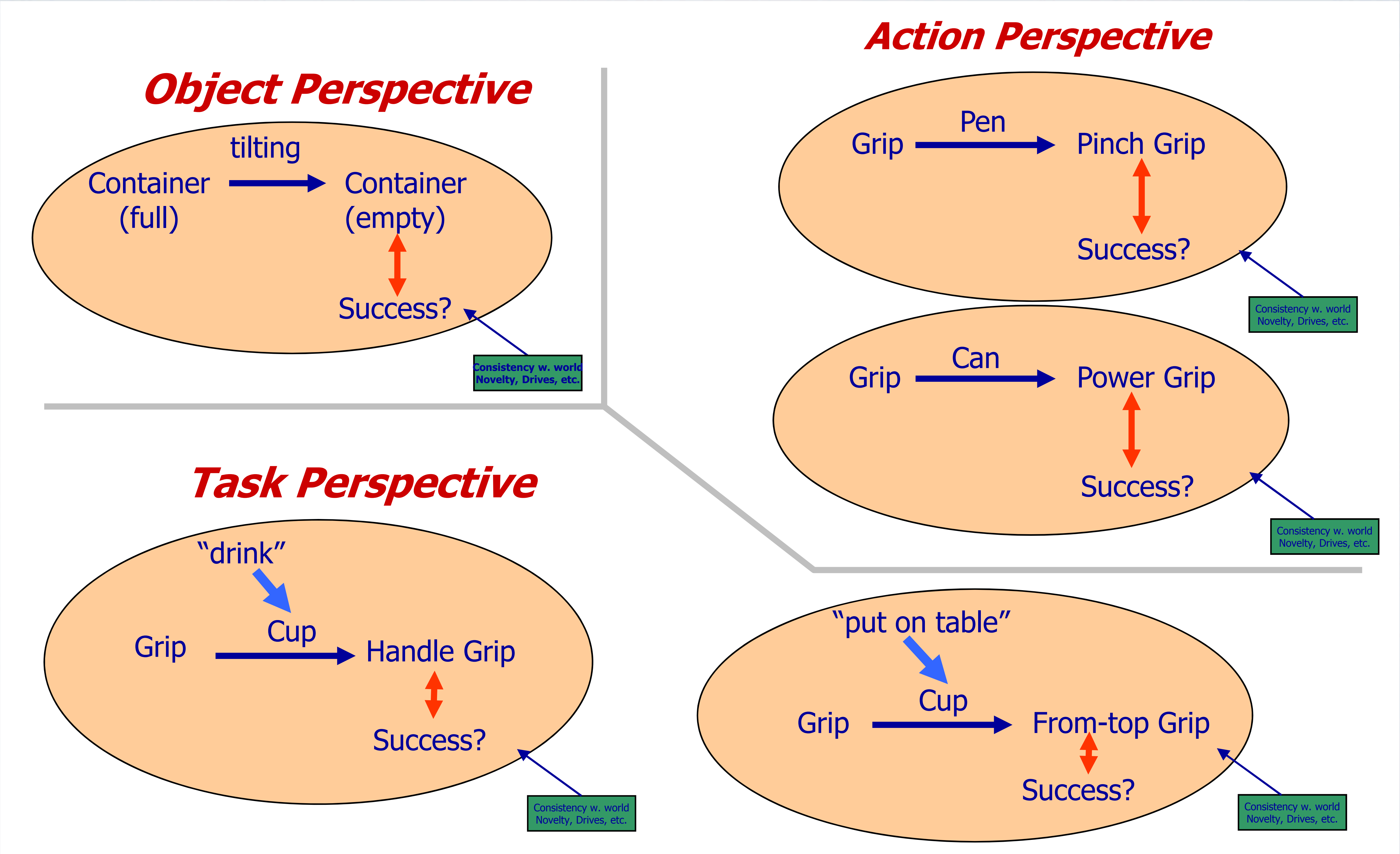

OAC
"Thing" & Attributes
Movement
"Thing" & changed Attributes
Objects & Actions have arisen if this process is successful
Success ?
Measured against:
Code-similarity emerges only through the fact that both codes describe the same physical entity
Human Neuronal "code"
Robot "code"
OACs are code-independent
Consistency with world Novelty, Drives, etc.

OAC



Code-similarity emerges only through the fact that both codes describe the same physical entity

OACs are code-independent


Object Perspective
tilting
Container (full)
Container (empty)
Success?
Consistency w. world Novelty, Drives, etc.
Action Perspective
Pen
Grip
Pinch Grip
Success?
Consistency w. world Novelty, Drives, etc.
Can
Grip
Power Grip
Success?
Consistency w. world Novelty, Drives, etc.
Task Perspective
"drink"
Cup
Grip
Handle Grip
Success?
Consistency w. world Novelty, Drives, etc.
"put on table"
Cup
Grip
From-top Grip
Success?
Consistency w. world Novelty, Drives, etc.

- Object Action Complexes (OACs)
  - Actions define the meaning of Objects
  - Objects suggest Actions (affordance)

- Object Action Complexes (OACs)
  - Actions define the meaning of Objects
  - Objects suggest Actions (affordance)
- OACs are associations of objects and affordances
  - Affordances can be expressed by STRIPS like-rules

- Object Action Complexes (OACs)
  - Actions define the meaning of Objects
  - Objects suggest Actions (affordance)
- OACs are associations of objects and affordances
  - Affordances can be expressed by STRIPS like-rules
- Associative memory ensures that
  - Object representations (and other preconditions) evoke affordances
  - Representations of affordances (and other preconditions) evoke objects

OACS VS. AFFORDANCES

- Affordances are "unidirectional": Objects affords actions
- OACs are "bidirectional": Object affords actions ↔ Actions suggest objects
- OACs can be chained (new complex OACs from simpler OACs "Tasks from skills = Planning")

ACTION PRIMITIVES WITHIN OACS

- **action hierarchy**
- Actions involve objects, Movements do not
- Action primitives are the atomic entities
- vital due to computational / combinatorial aspects

Activities
Actions
Action Primitives
Movements

ACTION PRIMITIVES WITHIN OACS

- **action hierarchy**
- Actions involve objects, Movements do not
- Action primitives are the atomic entities
- vital due to computational / combinatorial aspects

Activities
Actions
Action Primitives
Movements

action primitives are atomic building blocks of actions. They

ACTION PRIMITIVES WITHIN OACS

- **action hierarchy**
- Actions involve objects, Movements do not
- Action primitives are the atomic entities
- vital due to computational / combinatorial aspects

Activities
Actions
Action Primitives
Movements

action primitives are atomic building blocks of actions. They
- are meant to change the world state in a specific manner

ACTION PRIMITIVES WITHIN OACS

- **action hierarchy**
- Actions involve objects, Movements do not
- Action primitives are the atomic entities
- vital due to computational / combinatorial aspects

Activities
Actions
Action Primitives
Movements

action primitives are atomic building blocks of actions. They

- are meant to change the world state in a specific manner
- require a certain world state

ACTION PRIMITIVES WITHIN OACS

- **action hierarchy**
- Actions involve objects, Movements do not
- Action primitives are the atomic entities
- vital due to computational / combinatorial aspects

Activities
Actions
Action Primitives
Movements

action primitives are atomic building blocks of actions. They
- are meant to change the world state in a specific manner
- require a certain world state
- can generate a specific change to world state

ACTION PRIMITIVES WITHIN OACS

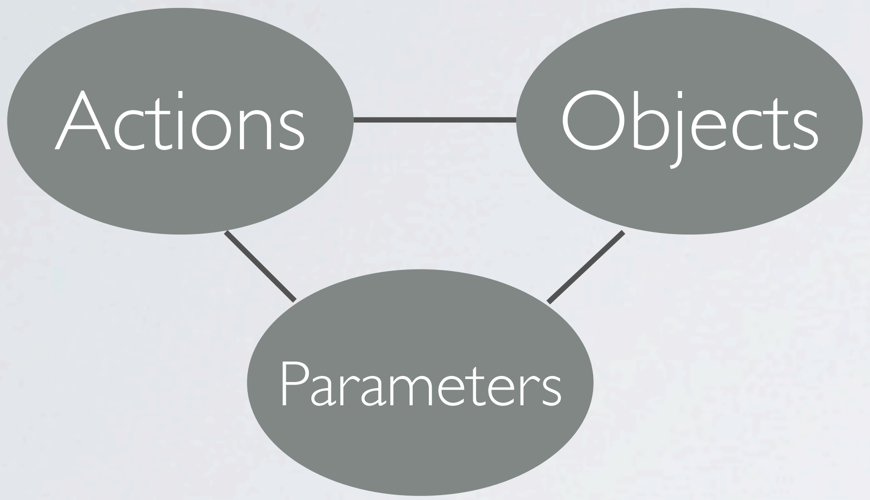
- **action hierarchy**
- Actions involve objects, Movements do not
- Action primitives are the atomic entities
- vital due to computational / combinatorial aspects

Activities
Actions
Action Primitives
Movements

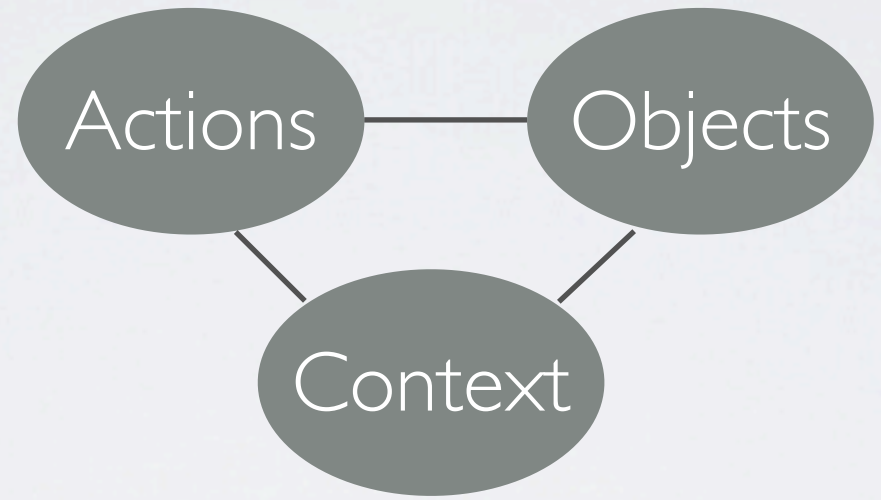
action primitives are atomic building blocks of actions. They
- are meant to change the world state in a specific manner
- require a certain world state
- can generate a specific change to world state

OACS contain the sensing capabilities (visual, haptic, force torque)

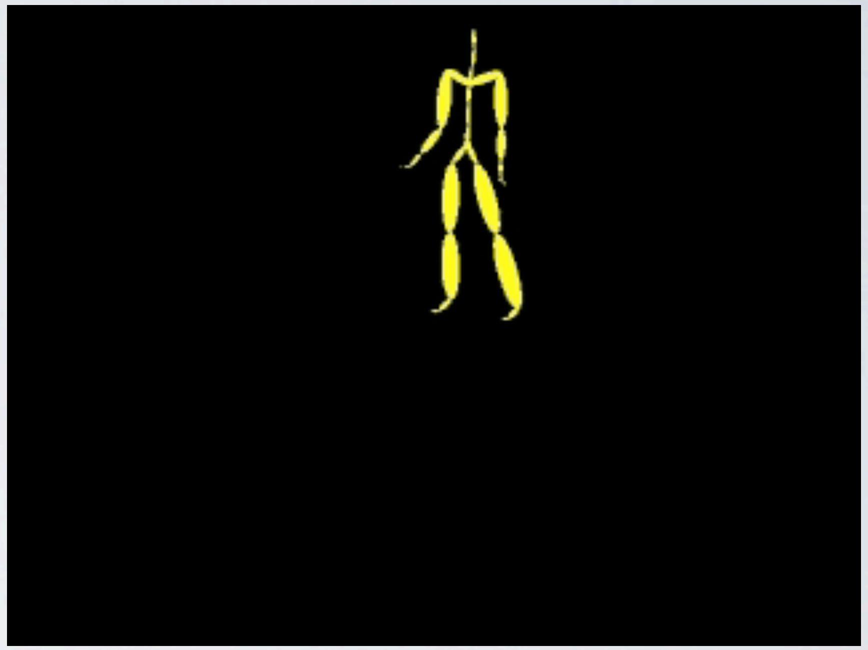
ATTEMPT TO IMPLEMENT OACS

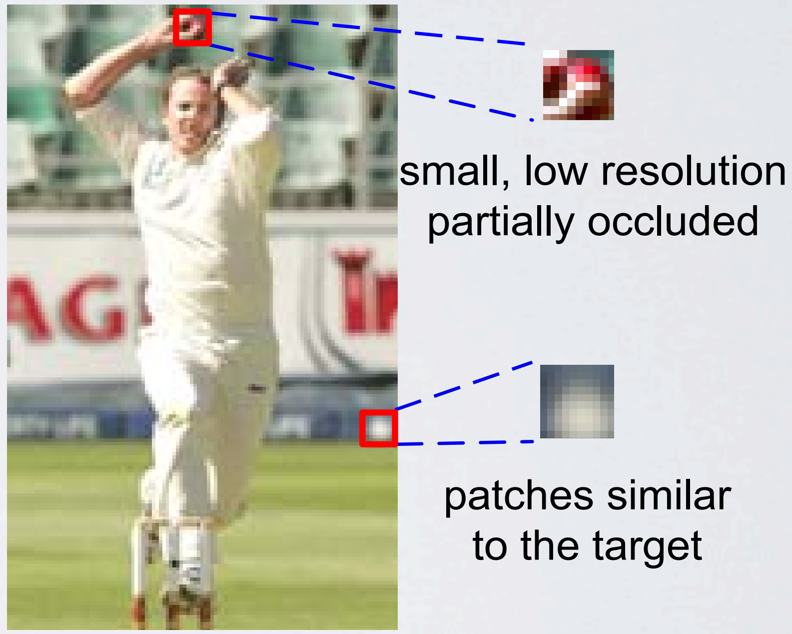


$$P(o, a, w) \equiv P(a, w|o)$$



$$P(o, a, c)$$





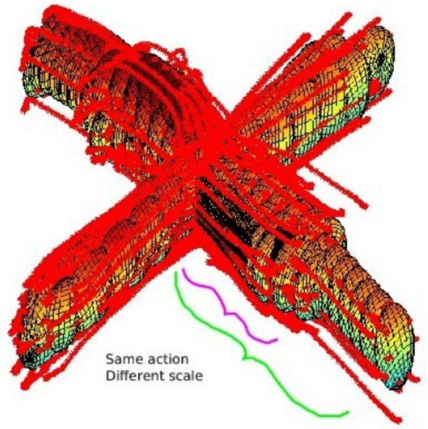
Li Fei-Fei, CVPR10

Institut for
Mekanik og Produktion

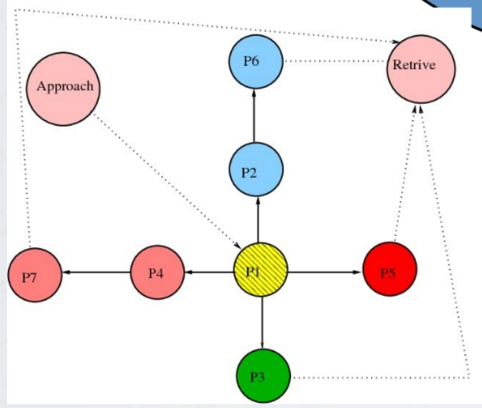
AALBORG UNIVERSITET

Observations

Non-supervised learning of action primitives

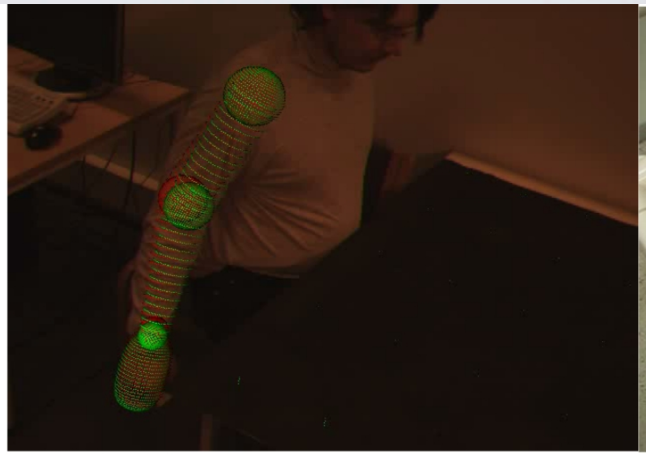


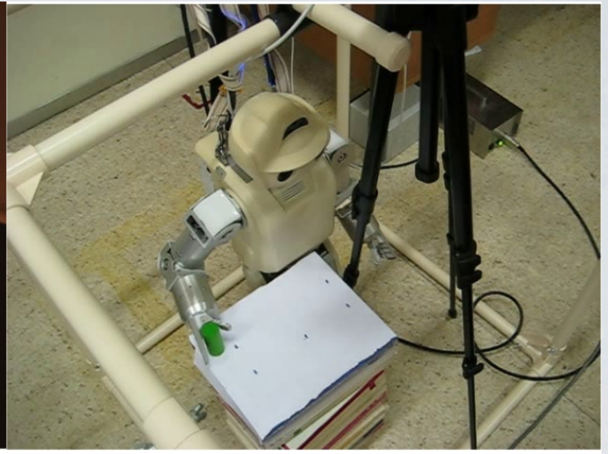
human actions classes according to their effect on objects



Build PHMM for each action class

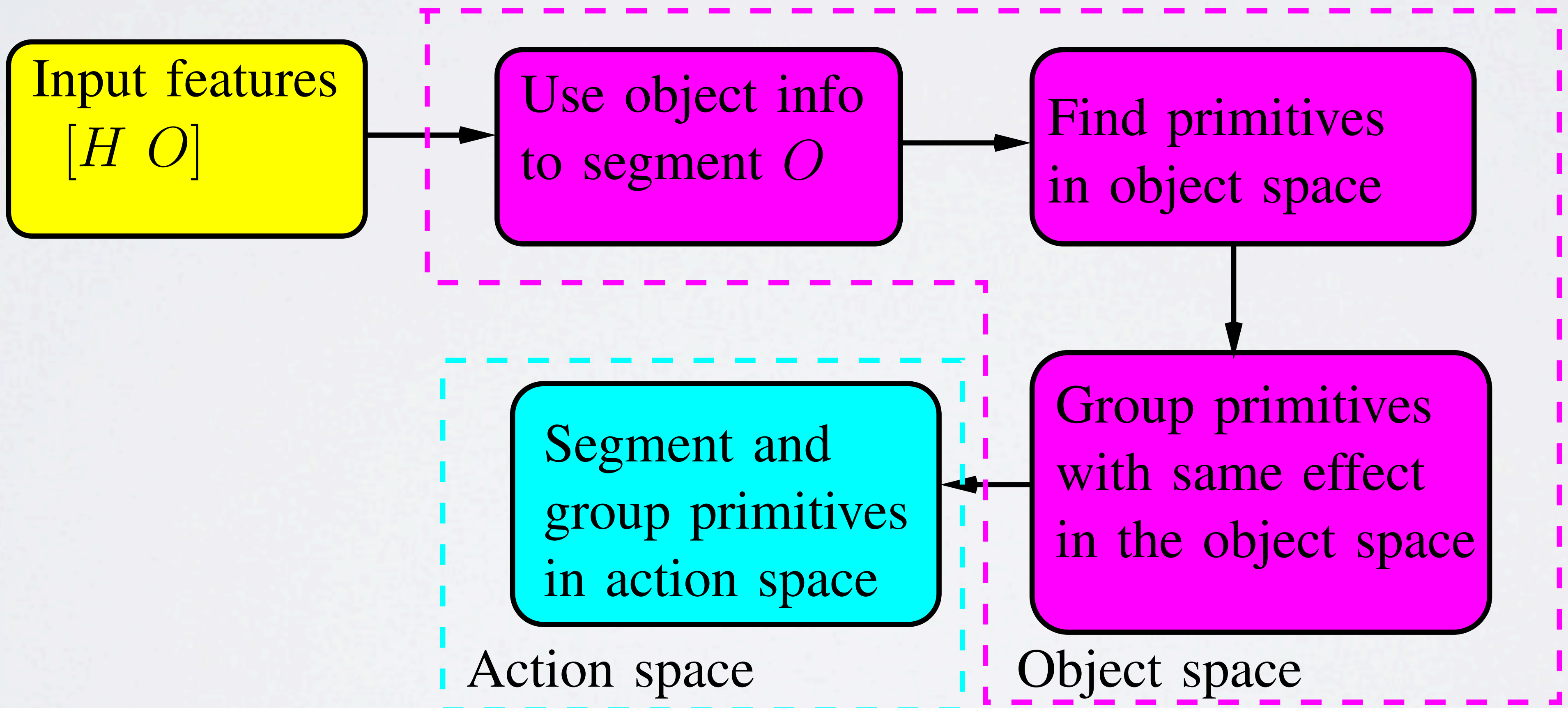
Generate actions on robots, recognize actions on humans



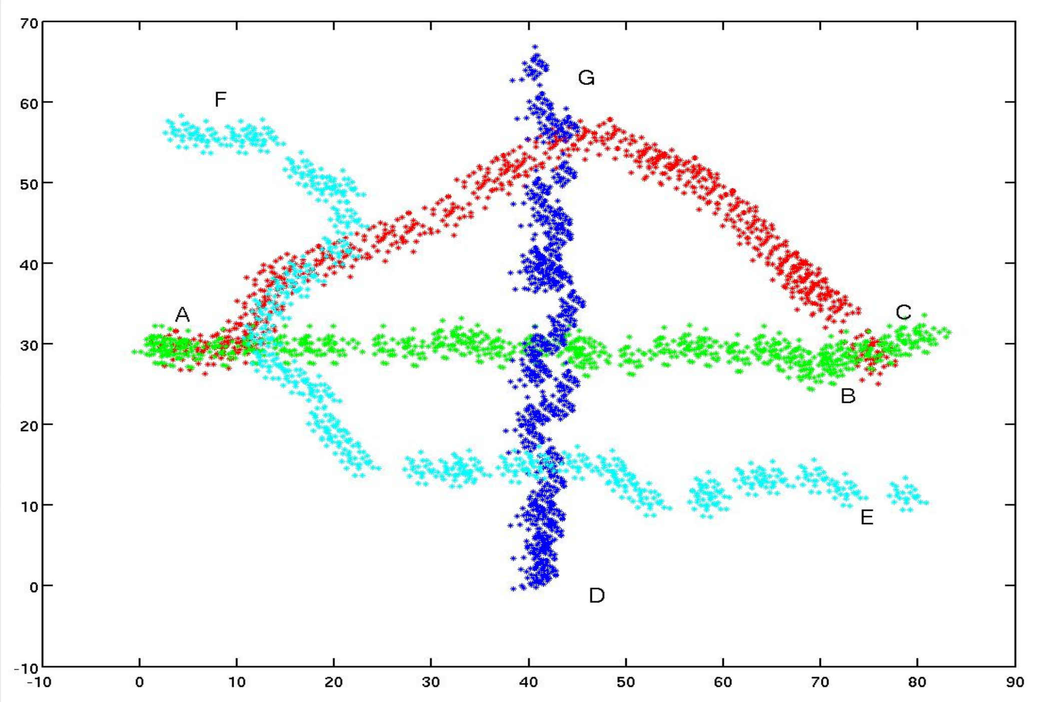


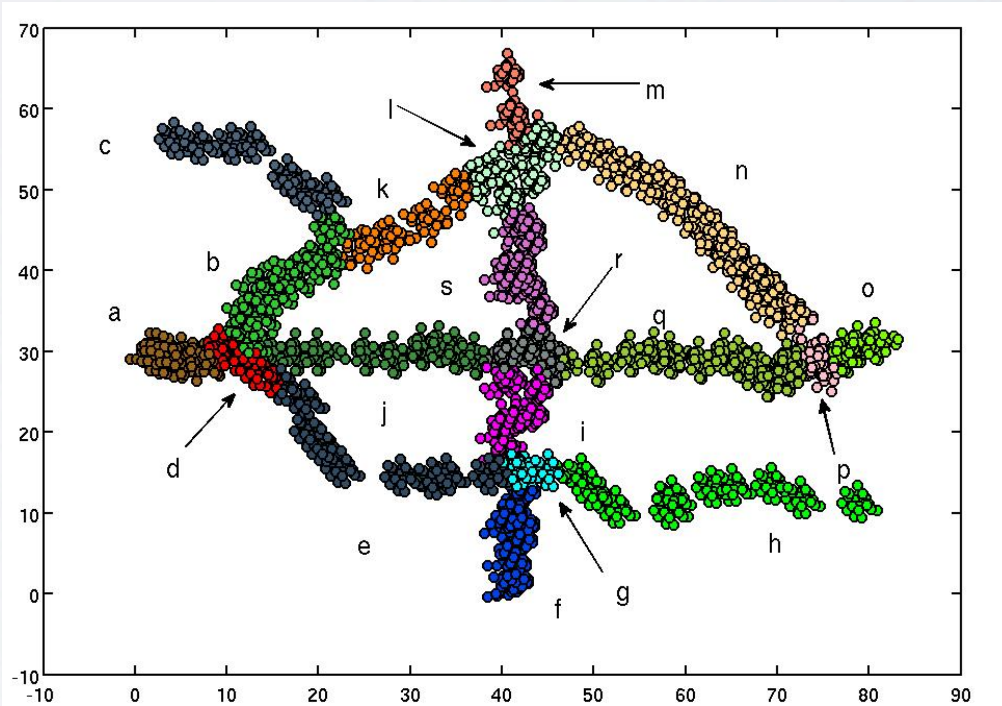
UNSUPERVISED LEARNING OF WORLD STATE SPACE

- Identify statistics in the **effect space O**
- Propagate the clustering of the **effect space** to the human **action space *H***



- **Parameterization is here object location + (speed and direction).**
- Unsupervised learning of context-free grammar
  - recursive construction of HMM
  - Dirichlet Process

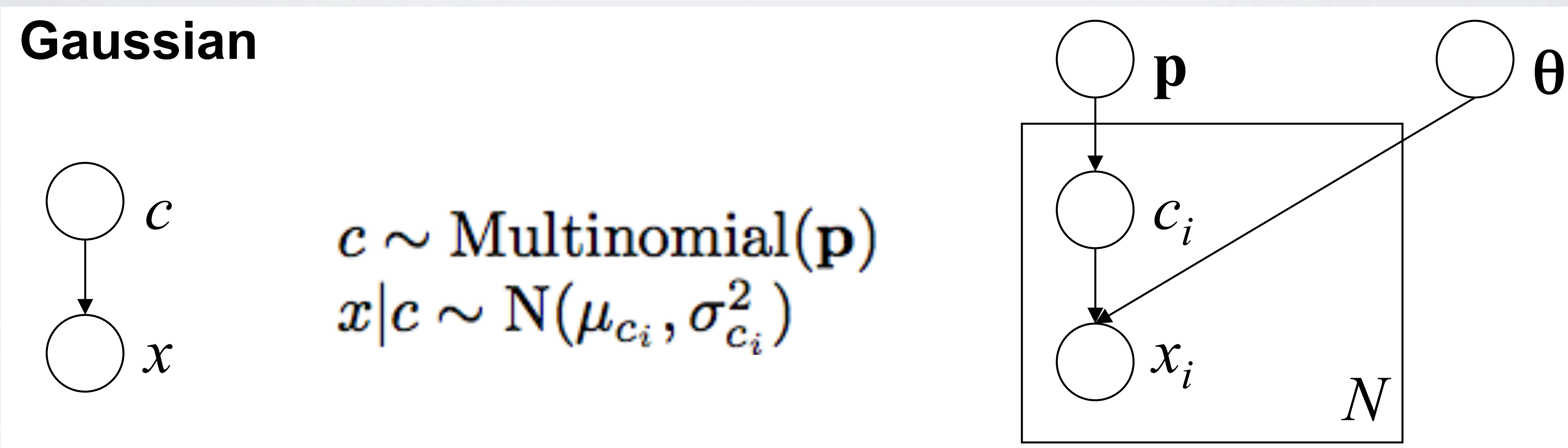




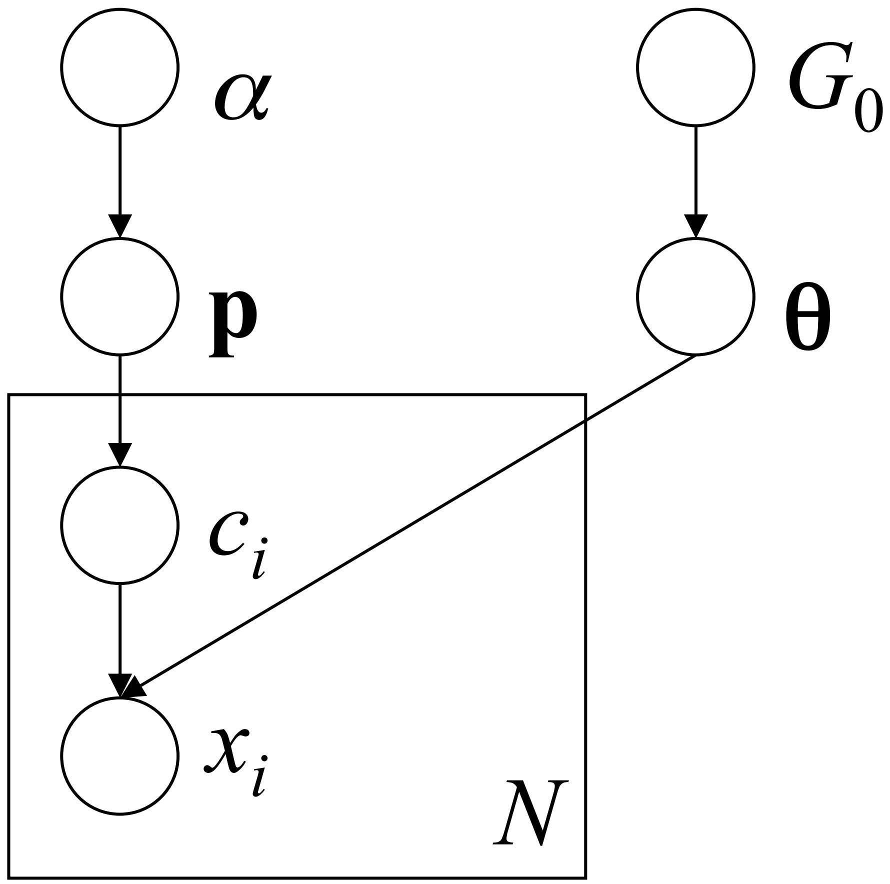
Institut for
Mekanik og Produktion

AALBORG UNIVERSITET

- Dirichlet Processes generalize finite mixture models to infinite mixture models
- choice of mixture number is data-driven, similar to k-means clustering
- Dirichlet Process find the number of mixtures automatically.
- DPs and HDPs are unsupervised.



Slide partially borrowed from Teg Grenager



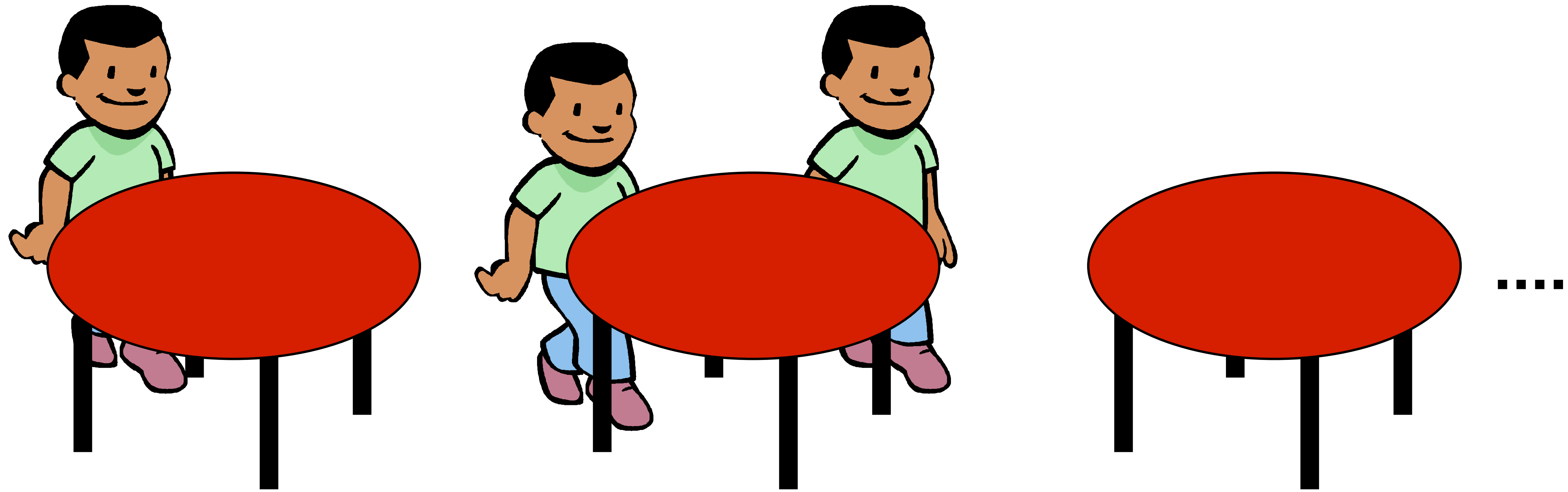
$$P(c_i = c | \mathbf{c}_{-i}) = \frac{\sum_{j \neq i} \mathbf{1}(c_j = c) + \frac{\alpha}{K}}{N - 1 + \alpha}$$

$$\lim_{K \to \infty} P(c_i = c | \mathbf{c}_{-i}) = \frac{\sum_{j \neq i} \mathbf{1}(c_j = c)}{N - 1 + \alpha}$$

$$P(c_i \neq c_j \forall j \neq i | \mathbf{c}_{-i}) = \frac{\alpha}{N - 1 + \alpha}$$

$$\mathbf{p} \sim \text{Dirichlet}(\tfrac{\alpha}{K}, \ldots, \tfrac{\alpha}{K})$$
$$\theta_c \sim G_0$$
$$c_i | \mathbf{p} \sim \text{Multinomial}(\mathbf{p})$$
$$x_i | c_i, \theta \sim F(\theta_{c_i})$$

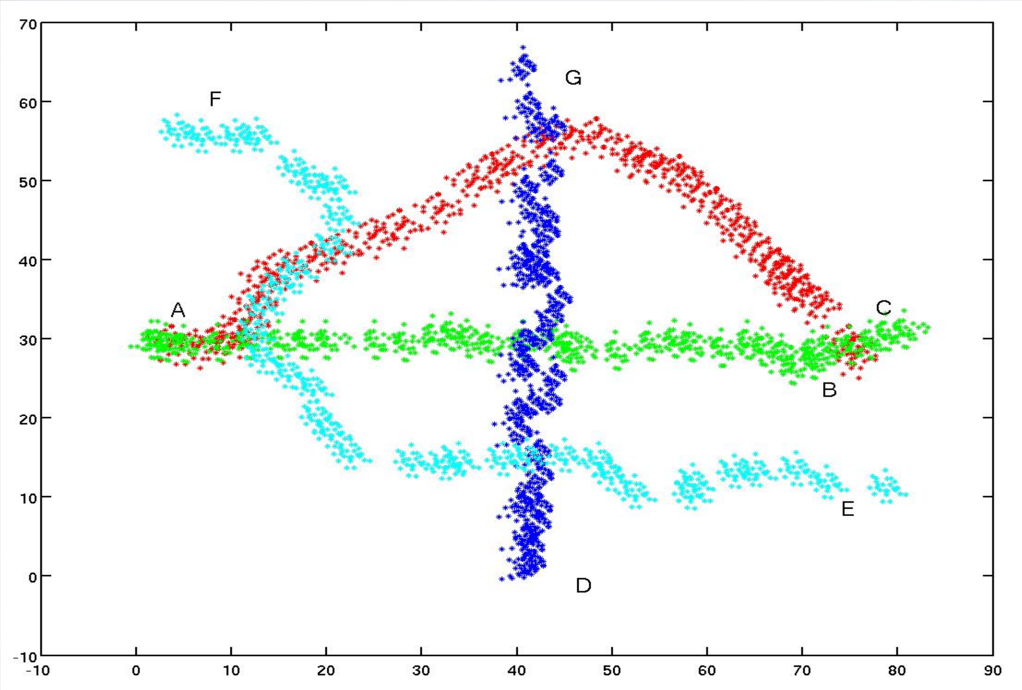
Slide partially borrowed from Teg Grenager

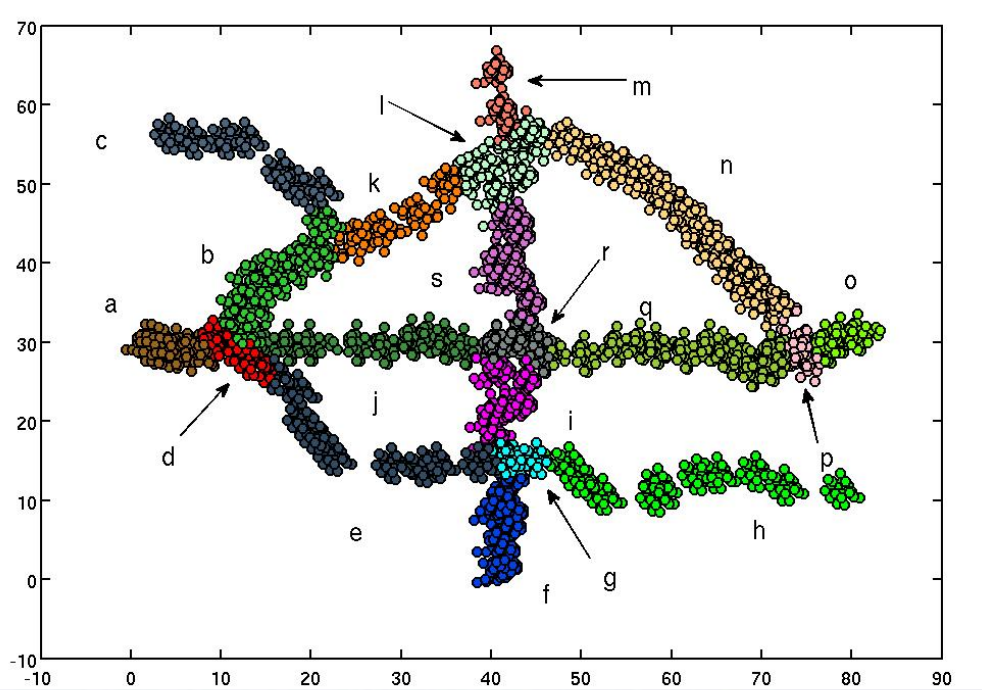


$$
\begin{array}{llll}
\mathrm{P}(c_i = c \mid \mathbf{c}_{-i}) = & 1 & 0 & 0 \\
& \frac{1}{1+\alpha} & \frac{\alpha}{1+\alpha} & 0 \\
& \frac{1}{2+\alpha} & \frac{1}{2+\alpha} & \frac{\alpha}{2+\alpha} \\
& \frac{1}{3+\alpha} & \frac{2}{3+\alpha} & \frac{\alpha}{3+\alpha}
\end{array}
$$

Slide partially borrowed from Teg Grenager

- What we have now is
  - **states:** clusters of trajectories that all have the same effect
  - detecting their **grammatical relationship** is trivial

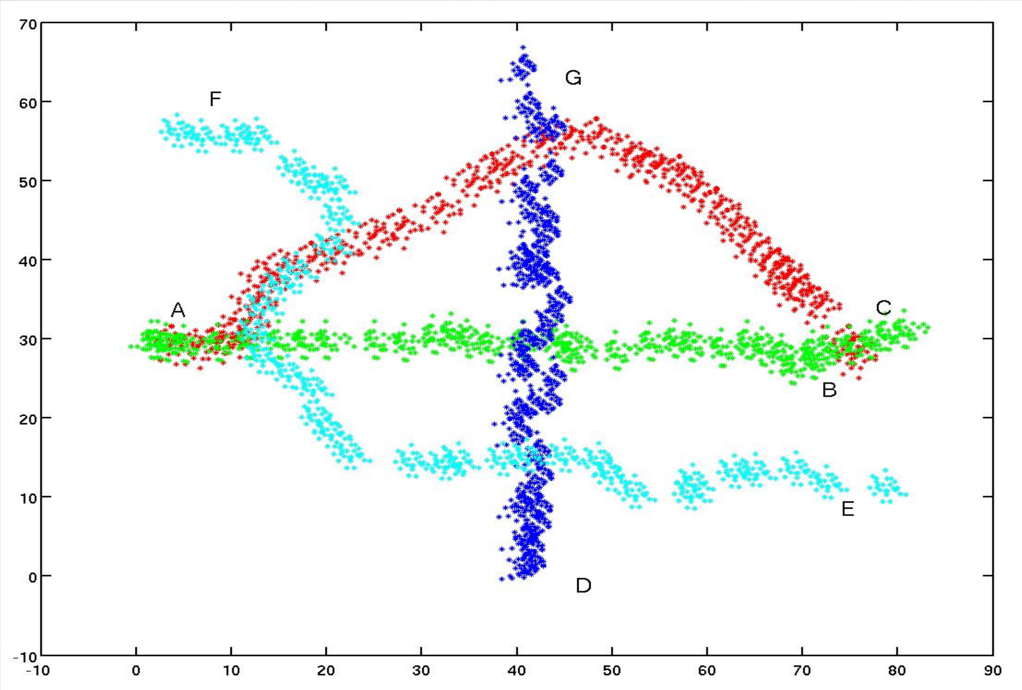


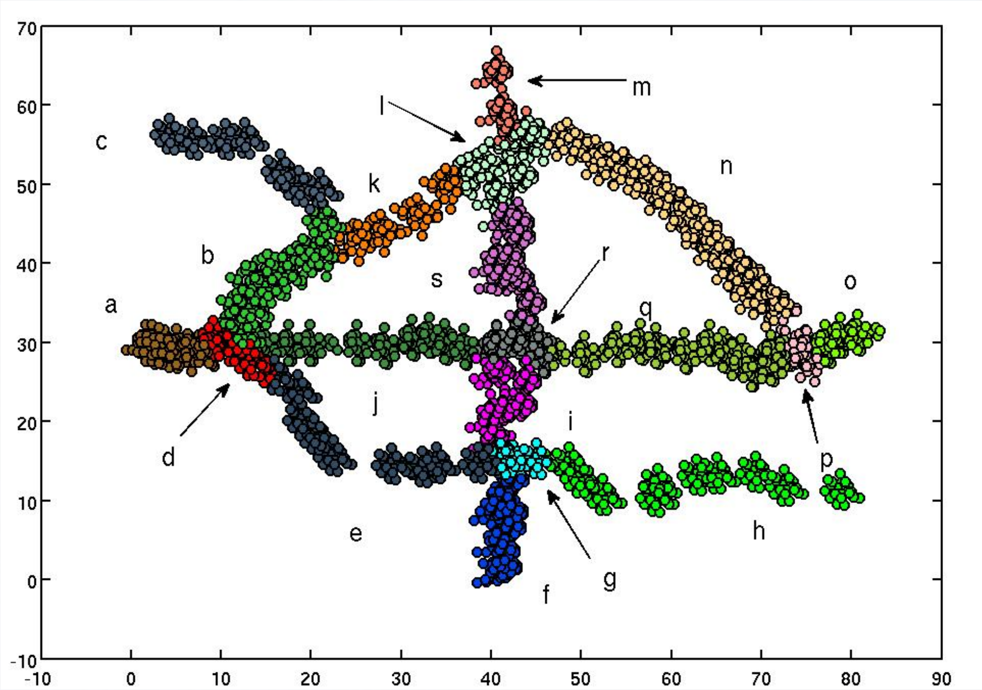


Sanmohan, V. Krüger, D. Kragic, and H. Kjellström. Automatic Primitive Segmentation and Action Recognition. *Advanced Robotics*, 25(6-7):871– 891, 2011.

V. Krueger, Sanmohan, D. Herzog, A. Ude, and D. Kragic. Learning Actions from Observations. *IEEE Robotics and Automation Magazine*, 17(2):30–43, 2010.

- What we have now is
  - **states:** clusters of trajectories that all have the same effect
  - detecting their **grammatical relationship** is trivial
- Next step builds a model for the observed actions within each cluster.

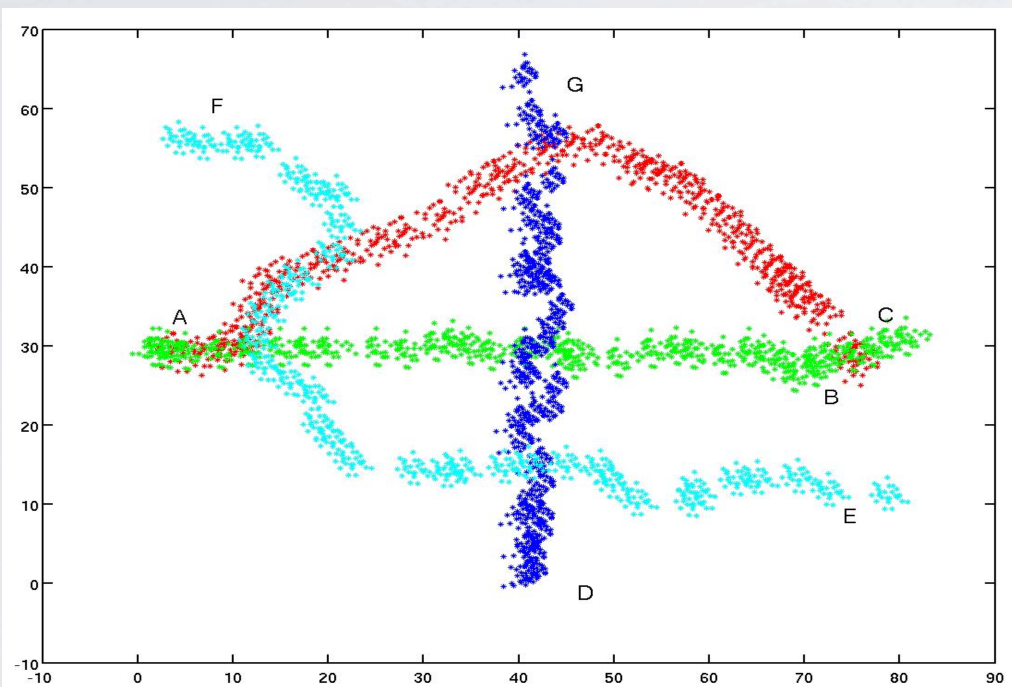


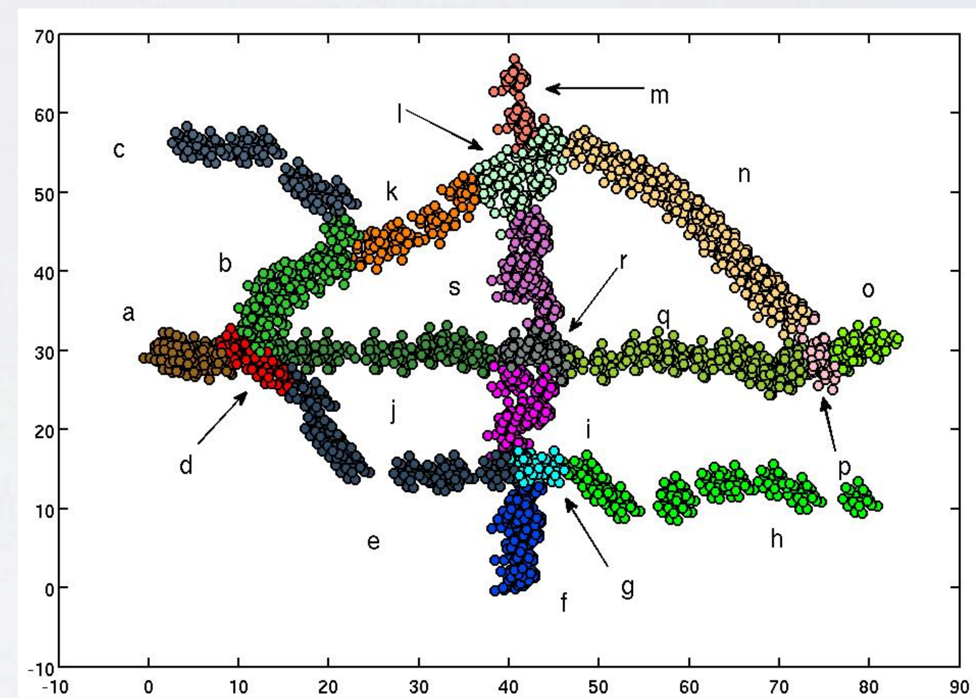


Sanmohan, V. Krüger, D. Kragic, and H. Kjellström. Automatic Primitive Segmentation and Action Recognition. *Advanced Robotics*, 25(6-7):871– 891, 2011.

V. Krueger, Sanmohan, D. Herzog, A. Ude, and D. Kragic. Learning Actions from Observations. *IEEE Robotics and Automation Magazine*, 17(2):30–43, 2010.

- What we have now is
  - **states:** clusters of trajectories that all have the same effect
  - detecting their **grammatical relationship** is trivial
- Next step builds a model for the observed actions within each cluster.
- Issues: right parameterization!! What matters?





Sanmohan, V. Krüger, D. Kragic, and H. Kjellström. Automatic Primitive Segmentation and Action Recognition. *Advanced Robotics*, 25(6-7):871– 891, 2011.

V. Krueger, Sanmohan, D. Herzog, A. Ude, and D. Kragic. Learning Actions from Observations. *IEEE Robotics and Automation Magazine*, 17(2):30–43, 2010.

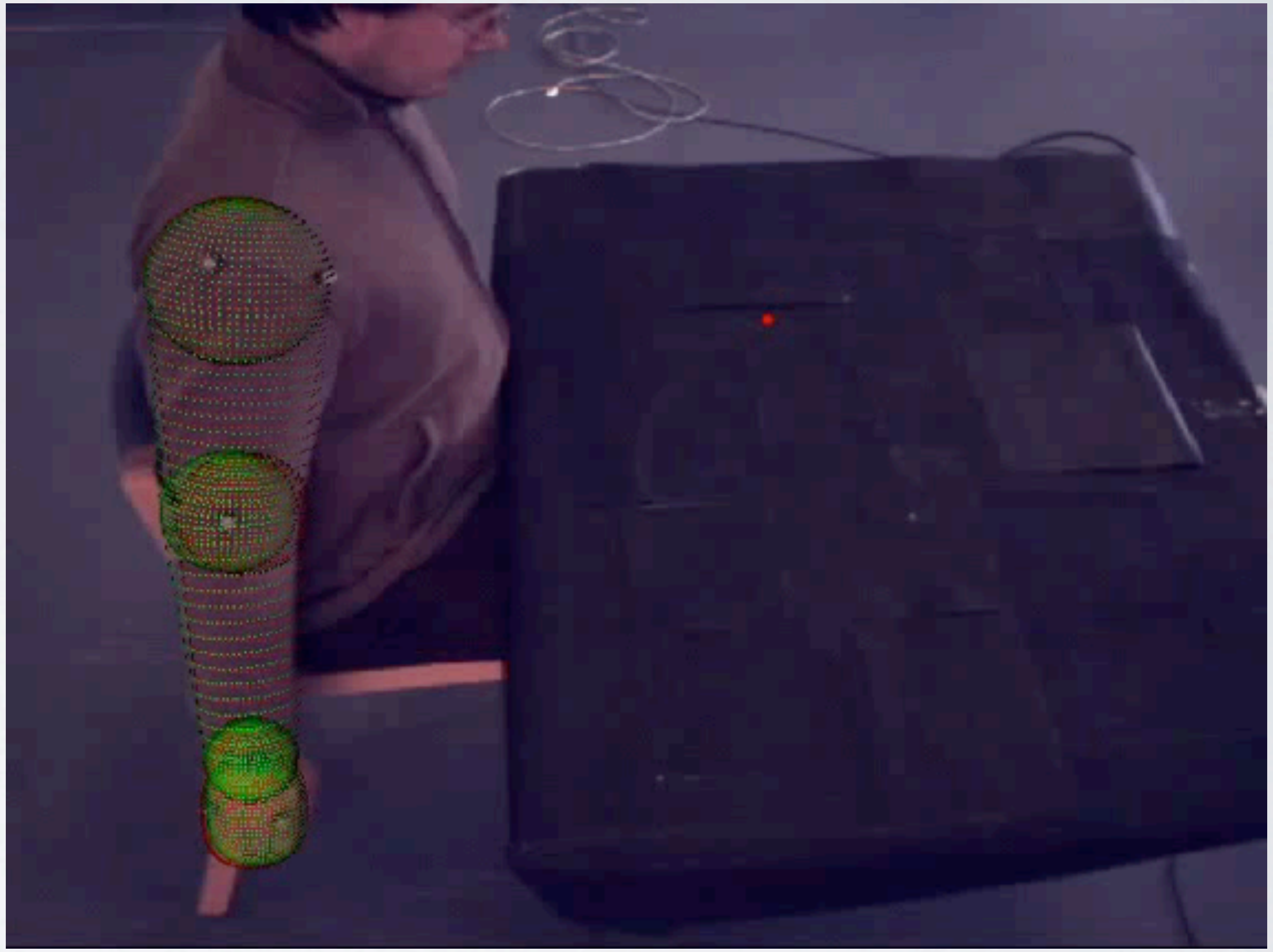
PARAMETRIC HIDDEN MARKOV MODELS

- Modeling clusters of trajectories
- **Parametric HMMs**: Hidden Markov Models, that allow for parametric means and covariances
- Parameters have meaning
  - given by the object and the effects.

Wilson&Bobick, PAMI 99

MODELING ACTIONS IN OBJECT ACTION SPACE

- ***Tell me the action and the object, and I know the movement*** *(up to some uncertainty)****!***
- Action and parameters infer joint settings and pose: **huge dimensionality reduction**
- Tracking is simplified, synthesis is trivial



On the fly demo, monocular(!!) data

TRACKING IN OBJECT ACTION SPACE

- Action parameters in case of a table top scenario $w = (k, x, y)$
  - k : PHMM state, associated with a human pose
  - x, y : object location on the table
  - i : action identifier

TRACKING IN OBJECT ACTION SPACE

- Action parameters in case of a table top scenario $w = (k, x, y)$
  - k : PHMM state, associated with a human pose
  - x, y : object location on the table
  - i : action identifier

to be estimated, can be constrained

TRACKING IN OBJECT ACTION SPACE

- Action parameters in case of a table top scenario $w = (k, x, y)$
  - k : PHMM state, associated with a human pose
  - x, y : object location on the table
  - i : action identifier

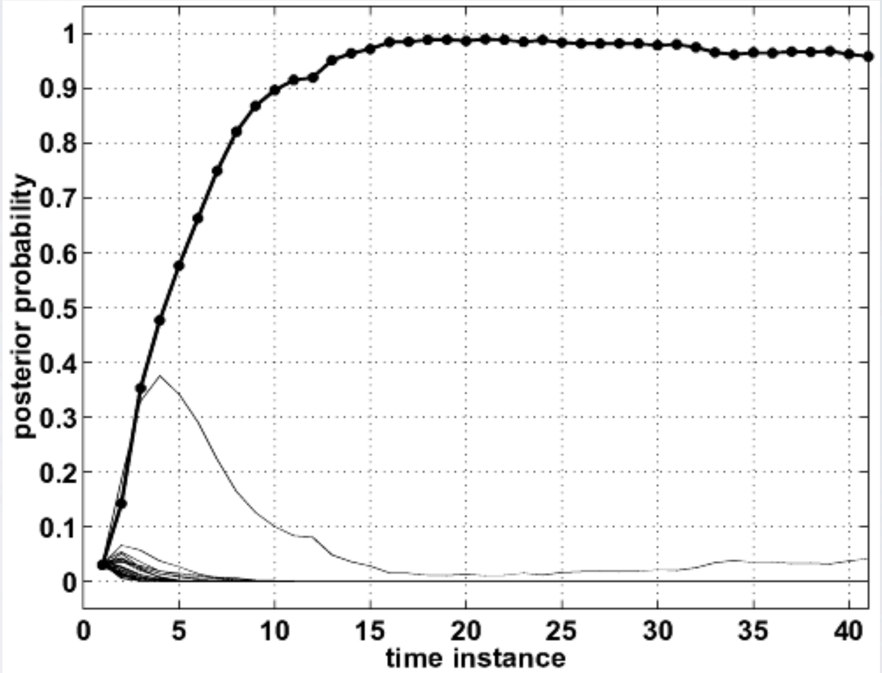
to be estimated, can be constrained

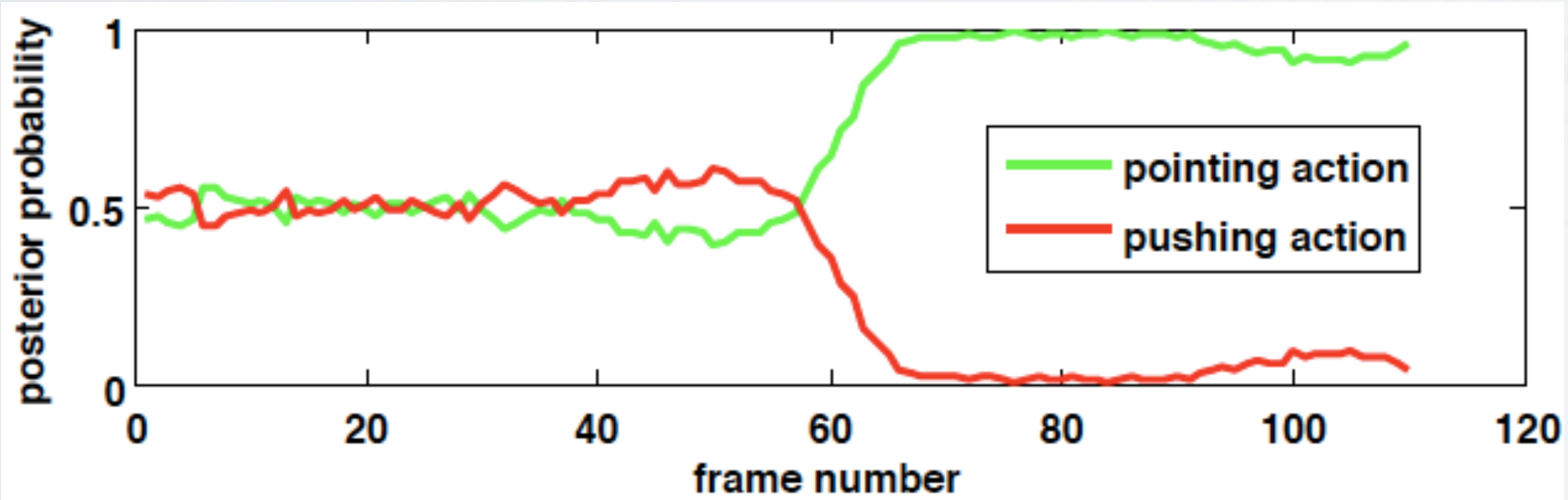
- Classical Bayesian Propagation over time

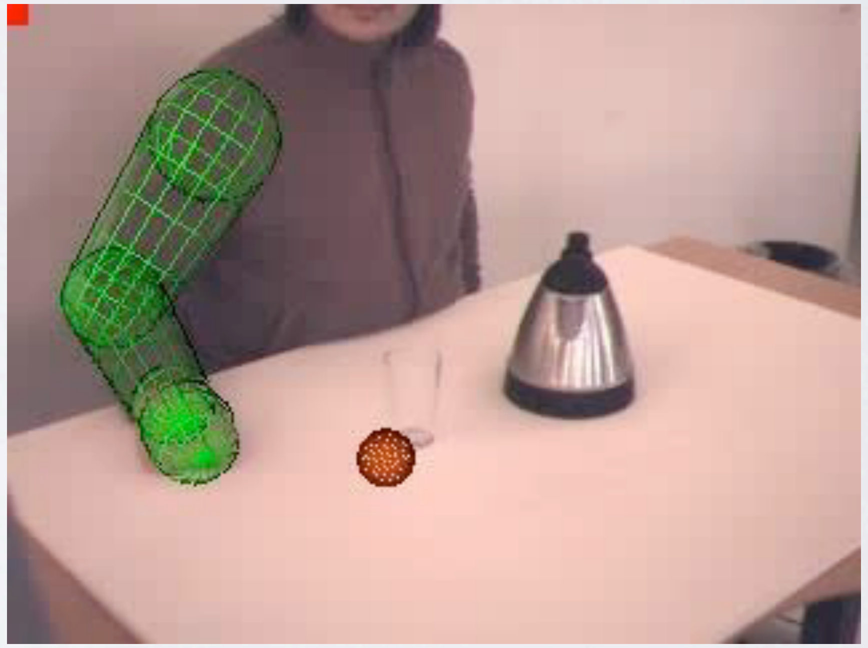
$$P(\omega_t, i_t | Z_1 \ldots Z_t)) \equiv p_t(\omega_t, i_t)$$

$$= \sum_{i_{t-1}} \int_{\omega_{t-1}} p_t(Z_t|\omega_t, i_t) p(\omega_t, i_i | \omega_{t-1}, i_{t-1}) p_{t-1}(\omega_{t-1}, i_{t-1}) d\omega_{t-1}$$

RECOGNIZING ACTIONS

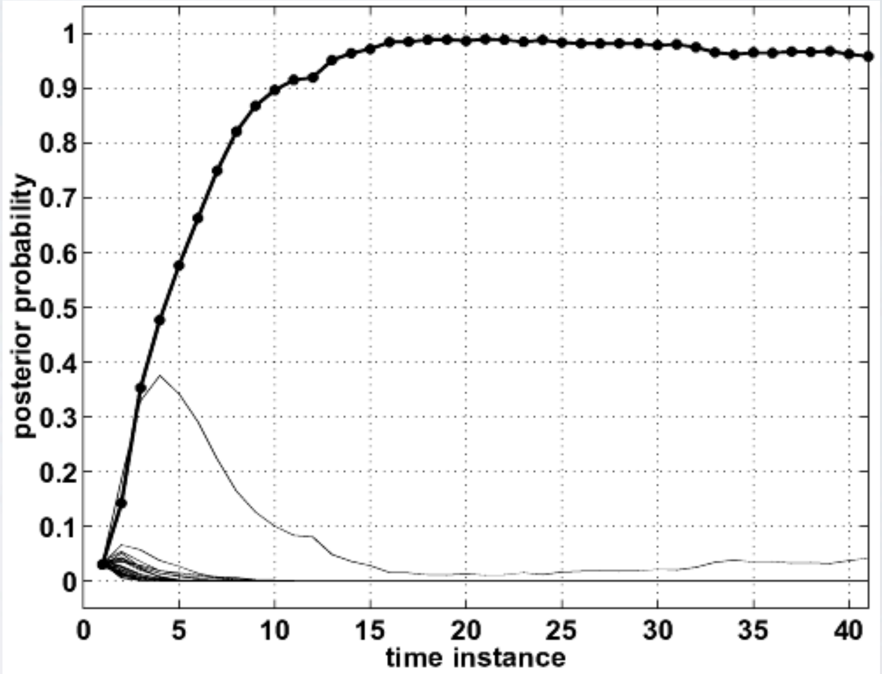


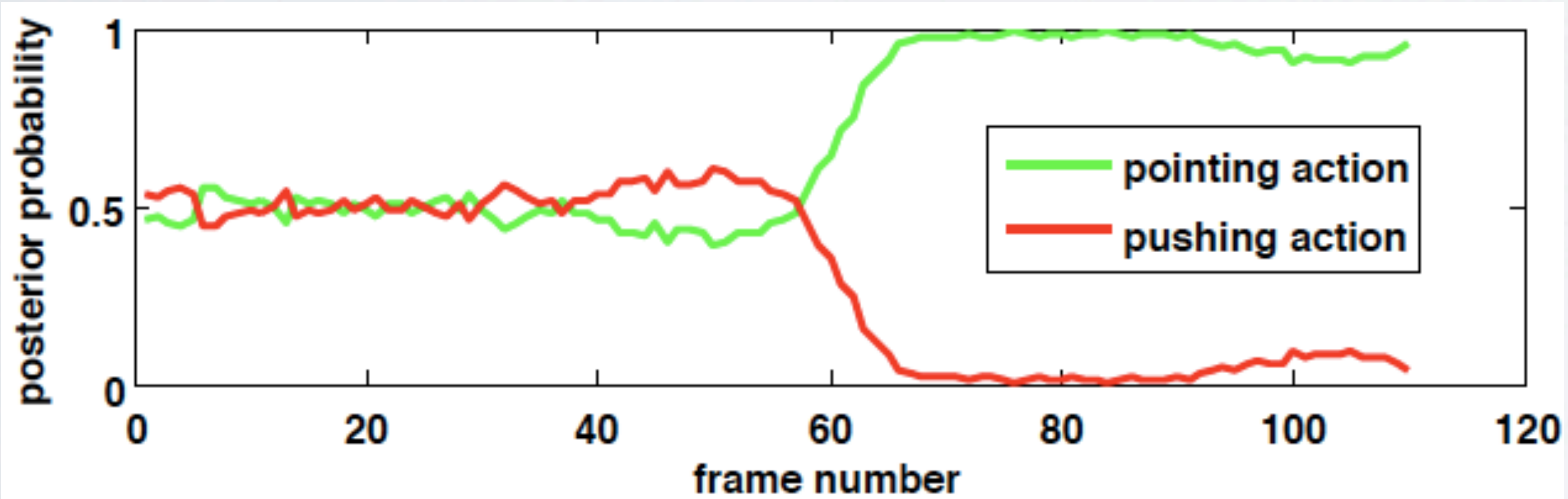


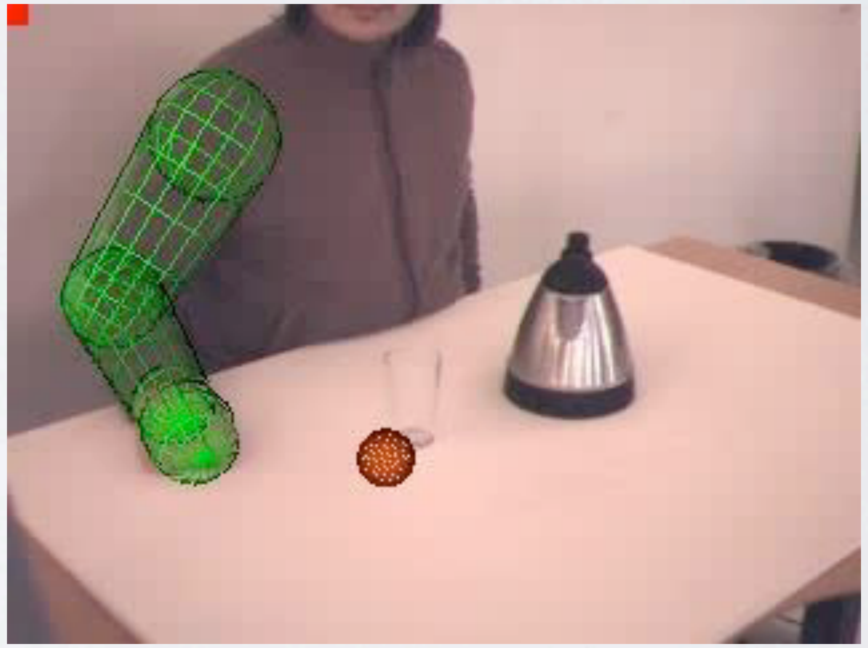


RECOGNIZING ACTIONS

- Parametric action recognition
- pointing, reaching, pushing and filling actions.
- parameters of the action are marginalized out



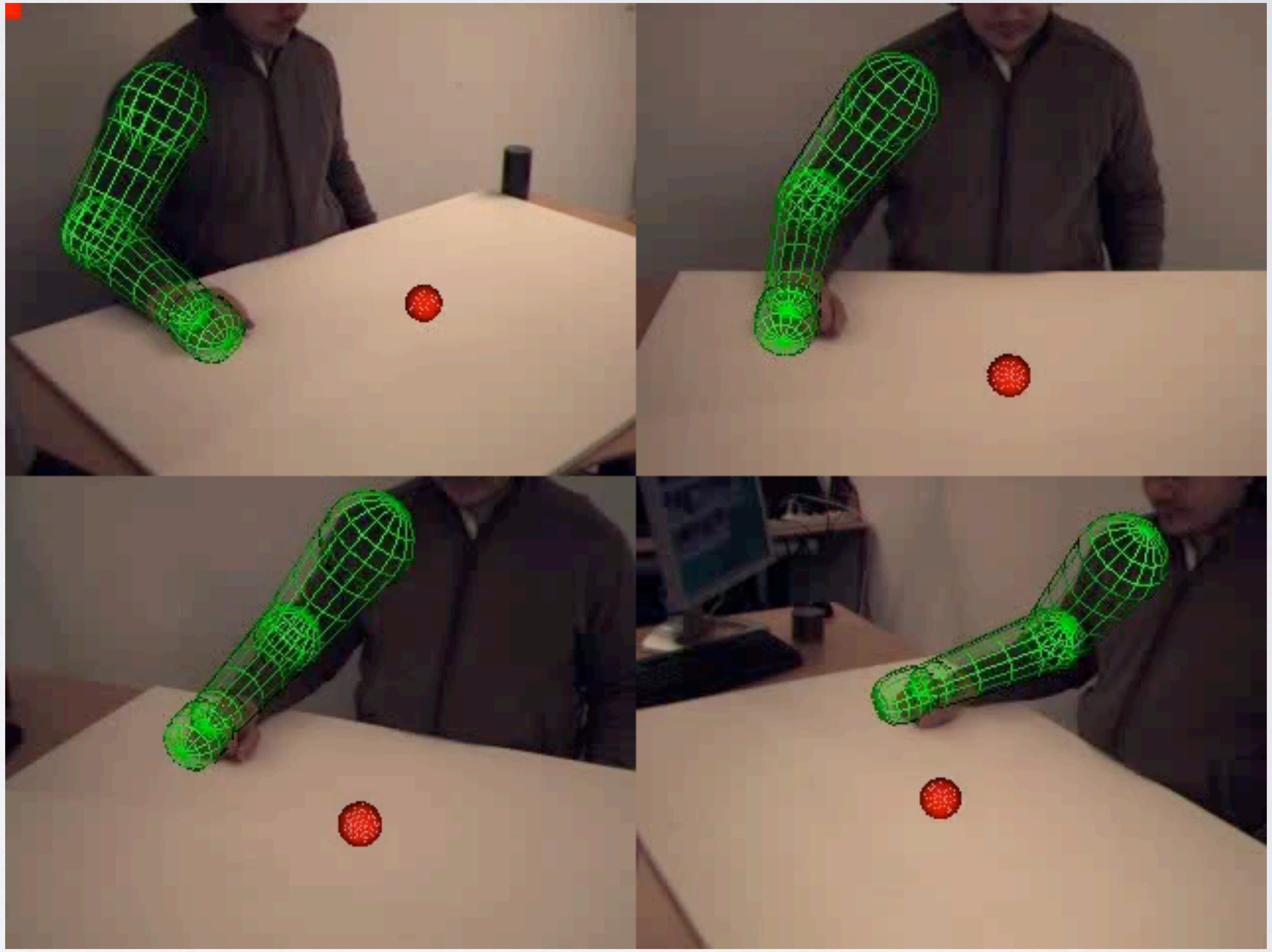




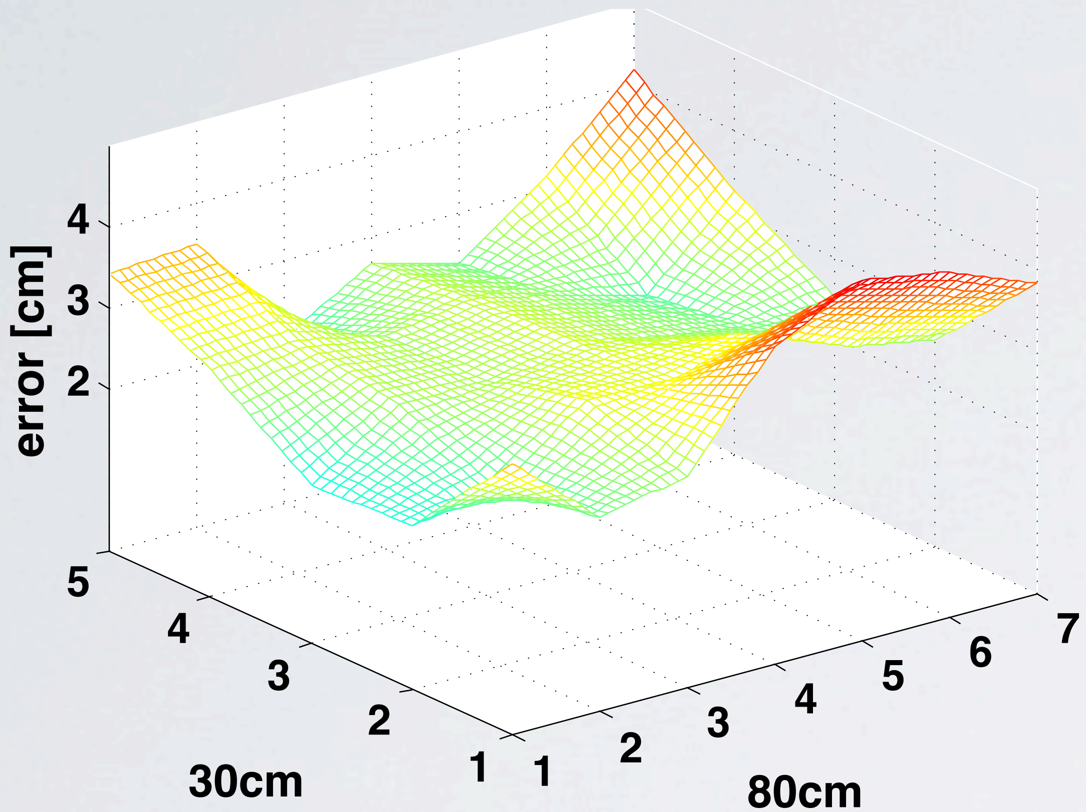
TRACKING IN OBJECT ACTION SPACE

TRACKING IN OBJECT ACTION SPACE

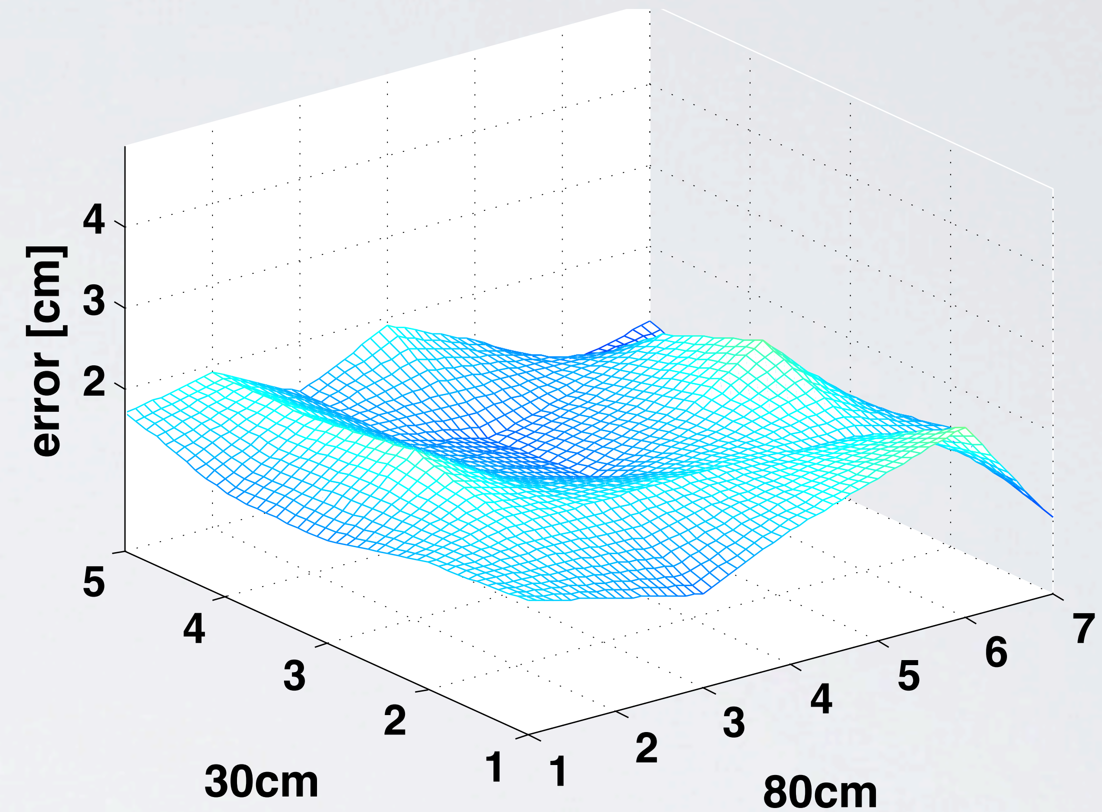
- Monocular and multi-view tracking
- red dot marks the active camera
- color of the ball is given by the parameter uncertainty



RECOVERY OF PARAMETERS



Deviation Trajectory



Error recovered params

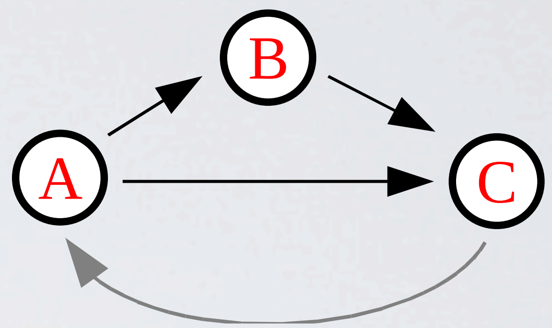
Same error as for human ground-truth trajectories(!)

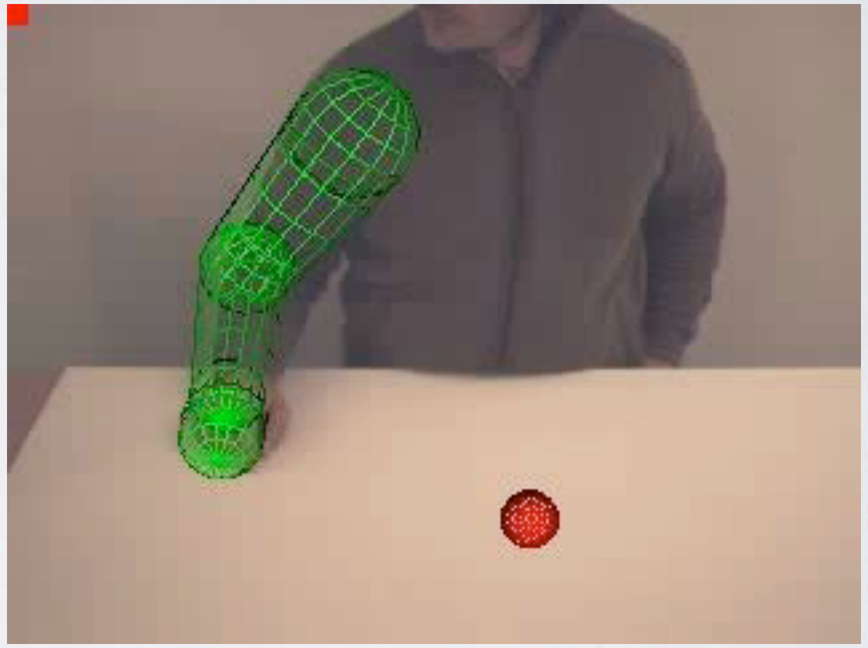
- 3x2 grid with 5 repetitions each.
- Integrated error along the trajectory

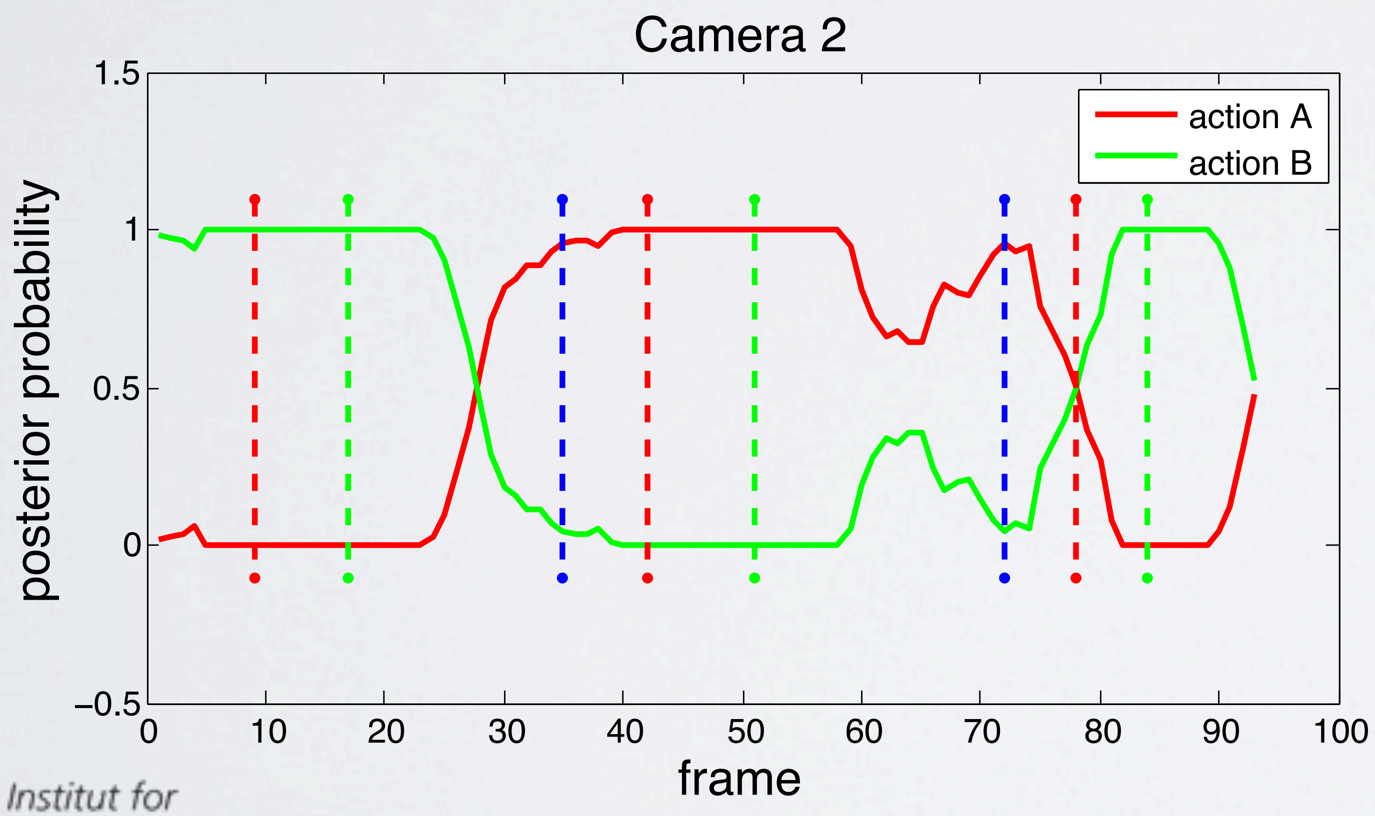
$$\epsilon = \sqrt{\int \sum_{i=1}^{7} \frac{(f_i(\alpha(t)) - \bar{f}_i(\bar{\alpha}(t)))^2}{7} dt \Big/ \int \alpha(d)dt}$$

TRACKING IN OBJECT ACTION SPACE

- Using more complex grammars
- Pick and Place actions
- Tracker switches between different action primitives



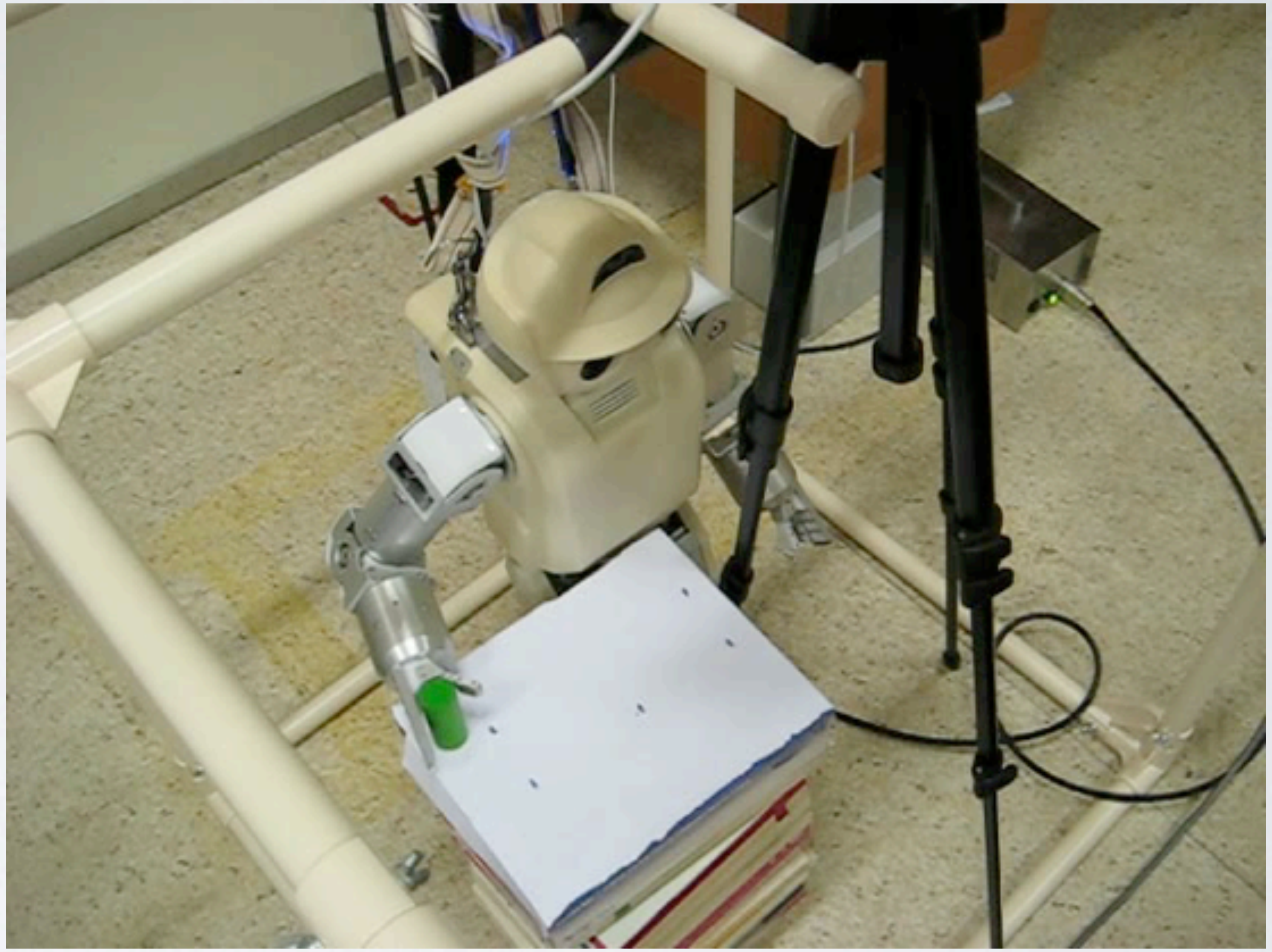




- Dennis Herzog and Volker Krueger. Tracking in Action Space. *In Human Motion: Understanding, Modeling, Capture and Animation, Workshop at ECCV 2010*, 2010. Springer.
- Dennis Herzog and Volker Krueger. Tracking in Action Space. *Int. Journal Computer Vision and Image Understanding (CVIU)*. submitted

OBJECT-ACTION SPACE FOR ROBOT CONTROL

- HOAP3 robot
- arm movements are defined by PHMMs
- robot picks and places the objects

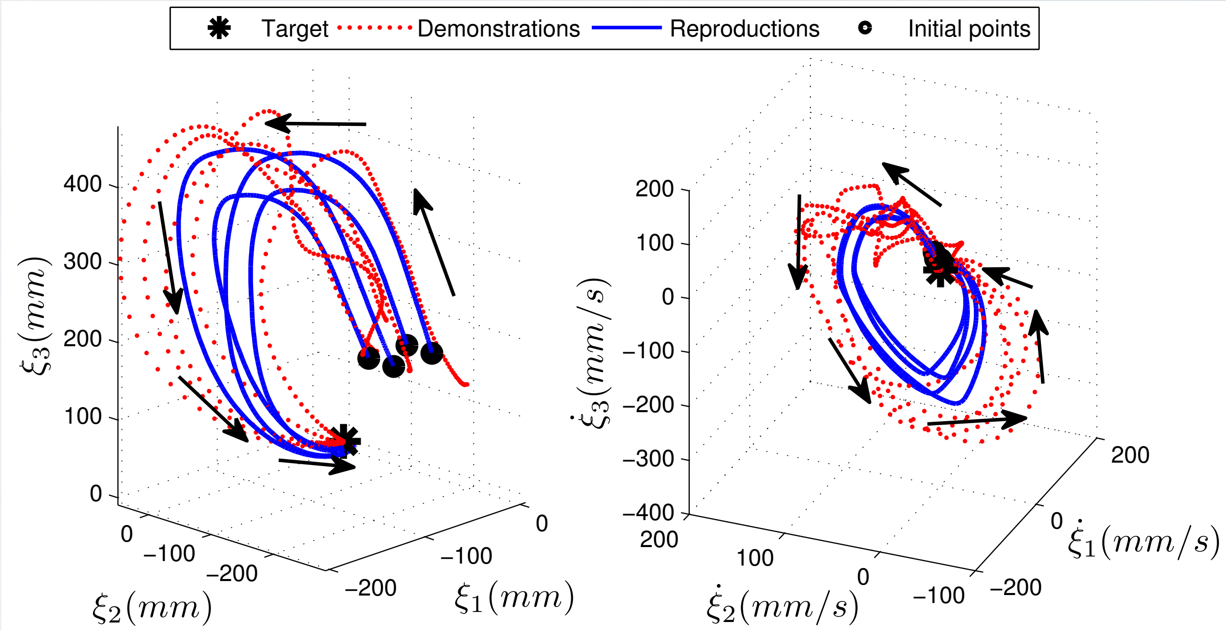


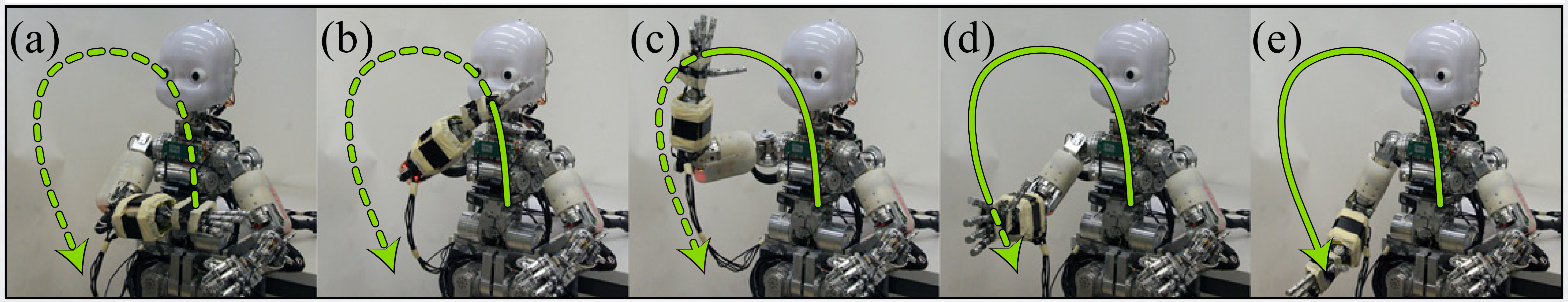
V. Krueger, Sanmohan, D. Herzog, A. Ude, and D. Kragic. Learning Actions from Observations. *IEEE Robotics and Automation Magazine,* 17(2):30–43, 2010.

D. Herzog, A. Ude, and V. Krueger. Motion Imitation and Recognition using Parametric Hidden Markov Models. In Humanoids, IEEE-RAS International Conference on Humanoid Robots, Daejeon, Korea, South, December 1-3, 2008

LEARNING OF ACTION PRIMITIVES

- Action primitives for motor control
- Starting point: Khansari-Zadeh and Billard, Imitation Learning of Globally Stable non-linear Point-to-Point Robot Motions using Non-linear Programming, IROS2010
- SEDS-approach (see Billard's and Calinon's presentation)





NON-LINEAR DYNAMIC MODEL SETTING UP THE PROBLEM

NON-LINEAR DYNAMIC MODEL
SETTING UP THE PROBLEM

- We model the movements using a dynamic model: $\dot{\xi} = f(\xi;\theta) + \epsilon$
  - θ = Model parameters
  - ξ = state vector

NON-LINEAR DYNAMIC MODEL
SETTING UP THE PROBLEM

- We model the movements using a dynamic model: $\dot{\xi} = f(\xi;\theta) + \epsilon$
  - θ = Model parameters
  - ξ = state vector
- To model f, a Gaussian mixture model is used:

$$\mathcal{P}(\xi^{t,n}, \dot{\xi}^{t,n}) = \sum_{k=1}^{K} \pi^k \mathcal{N}(\xi^{t,n}, \dot{\xi}^{t,n}; \theta^k)$$

NON-LINEAR DYNAMIC MODEL
SETTING UP THE PROBLEM

- We model the movements using a dynamic model: $\dot{\xi} = f(\xi;\theta) + \epsilon$
  - $\theta =$ Model parameters
  - $\xi =$ state vector
- To model f, a Gaussian mixture model is used:

$$\mathcal{P}(\xi^{t,n},\dot{\xi}^{t,n}) = \sum_{k=1}^{K} \pi^k \mathcal{N}(\xi^{t,n},\dot{\xi}^{t,n};\theta^k)$$

- where $\theta^k = \{\pi^k, \mu^k, \Sigma^k\}$ $\quad \mu^k = \begin{pmatrix} \mu^k_{\xi} \\ \mu^k_{\dot{\xi}} \end{pmatrix}$, $\Sigma^k = \begin{pmatrix} \Sigma^k_{\xi} & \Sigma^k_{\xi\dot{\xi}} \\ \Sigma^k_{\dot{\xi}\xi} & \Sigma^k_{\dot{\xi}} \end{pmatrix}$

NON-LINEAR DYNAMIC MODEL SETTING UP THE PROBLEM

- We model the movements using a dynamic model: $\dot{\xi} = f(\xi;\theta) + \epsilon$
  - $\theta =$ Model parameters
  - $\xi =$ state vector
- To model f, a Gaussian mixture model is used:

$$\mathcal{P}(\xi^{t,n}, \dot{\xi}^{t,n}) = \sum_{k=1}^{K} \pi^k \mathcal{N}(\xi^{t,n}, \dot{\xi}^{t,n}; \theta^k)$$

- where $\theta^k = \{\pi^k, \mu^k, \Sigma^k\}$ $\quad \mu^k = \begin{pmatrix} \mu^k_{\xi} \\ \mu^k_{\dot{\xi}} \end{pmatrix}$, $\Sigma^k = \begin{pmatrix} \Sigma^k_{\xi} & \Sigma^k_{\xi\dot{\xi}} \\ \Sigma^k_{\dot{\xi}\xi} & \Sigma^k_{\dot{\xi}} \end{pmatrix}$
- This can then be rewritten as

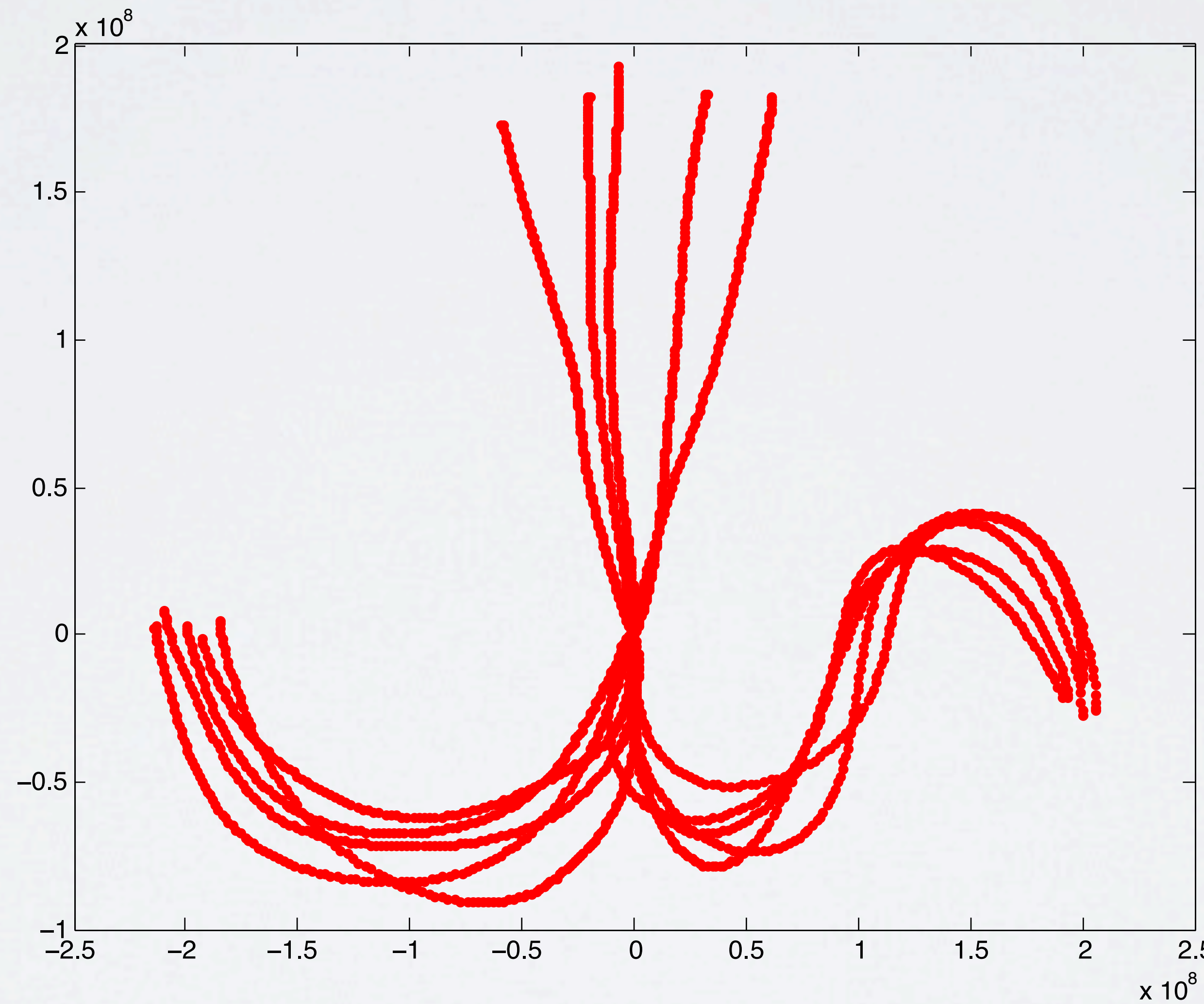
$$\hat{\dot{\xi}} = \sum_{k=1}^{K} \frac{\pi^k \mathcal{N}(\xi;\theta^k)}{\sum_{i=1}^{K} \pi^i \mathcal{N}(\xi;\theta^i)} (\mu^k_{\dot{\xi}} + \Sigma^k_{\dot{\xi}\xi}(\Sigma^k_{\xi})^{-1}(\xi - \mu^k_{\xi}))$$

Institut for
Mekanik og Produktion

AALBORG UNIVERSITET

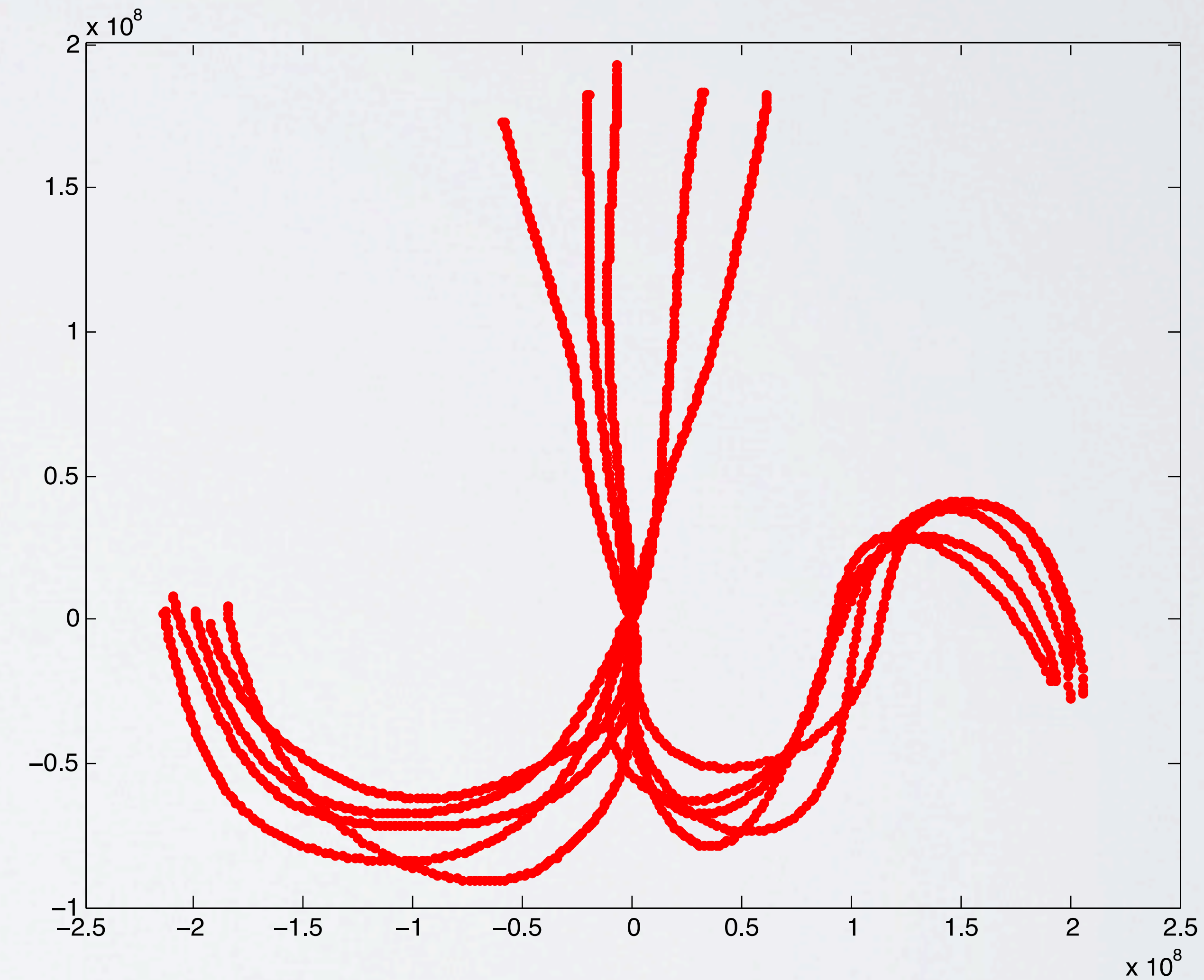
BUT WHAT ABOUT K?

- Someone needs to decide!



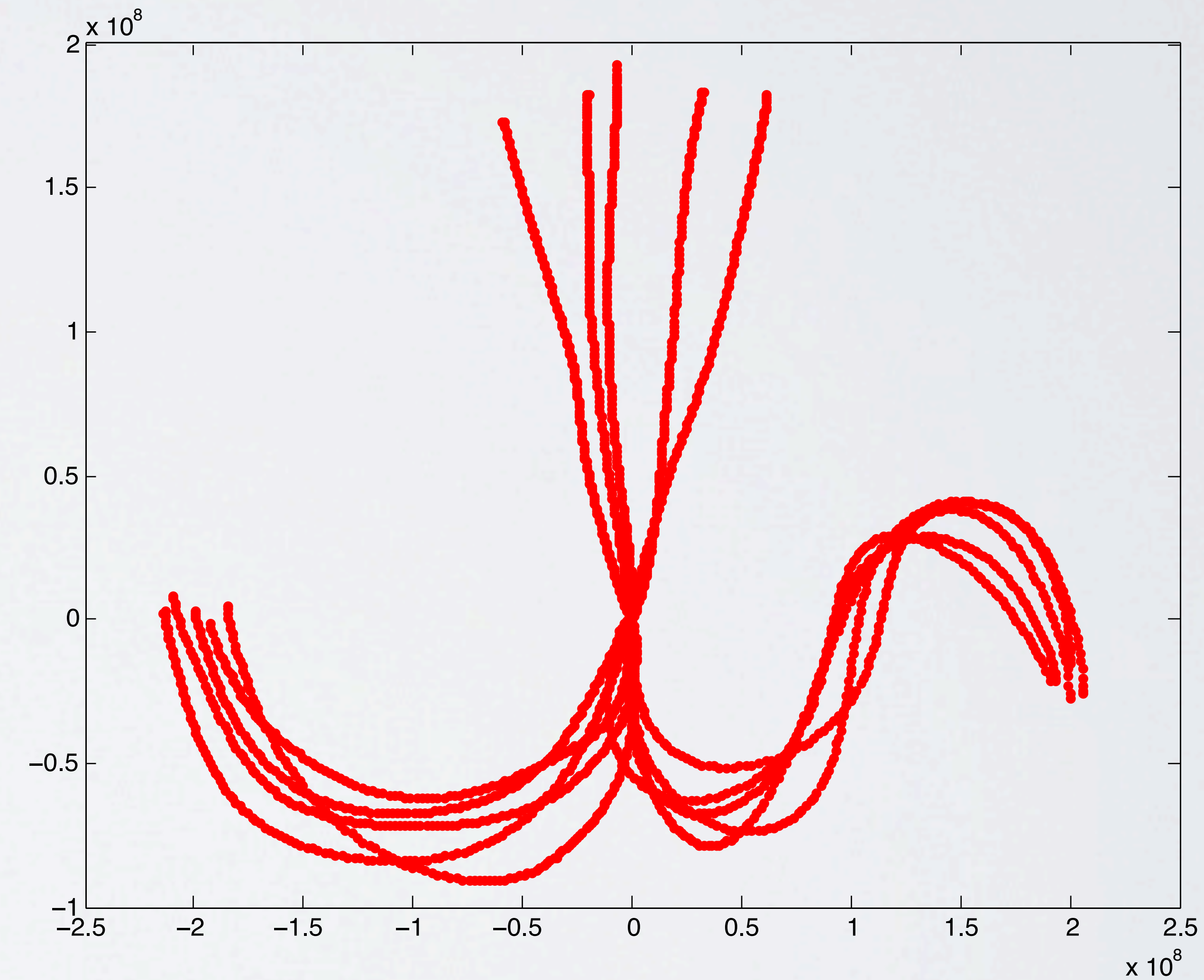
BUT WHAT ABOUT K?

- Someone needs to decide!
- We know that finding K is a principle problem!
  - so that is fine.



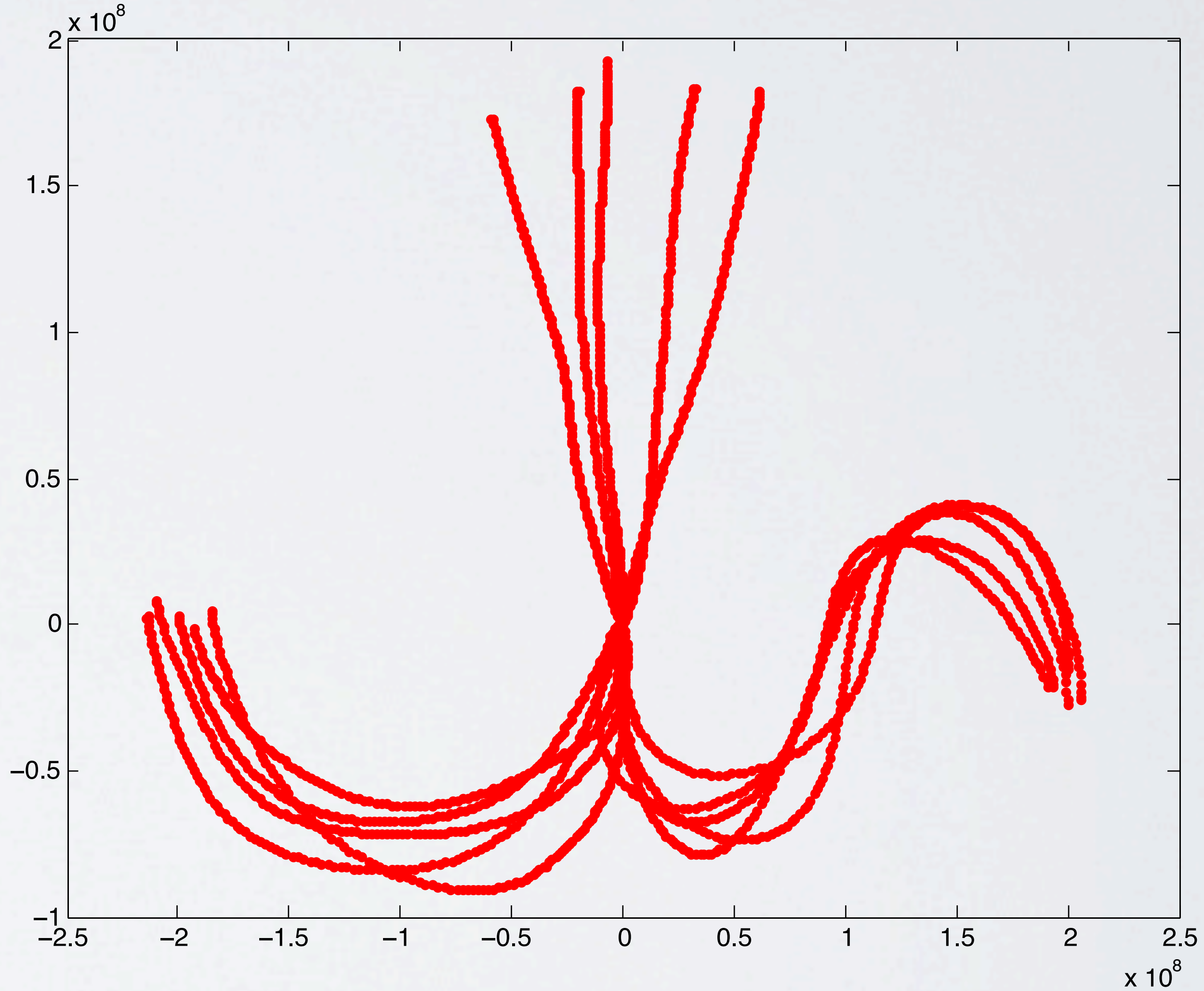
BUT WHAT ABOUT K?

- Someone needs to decide!
- We know that finding K is a principle problem!
    - so that is fine.
- But the big problem is this:
    - the actions to be learned must be known in advance!! *That* is obviously a problem!



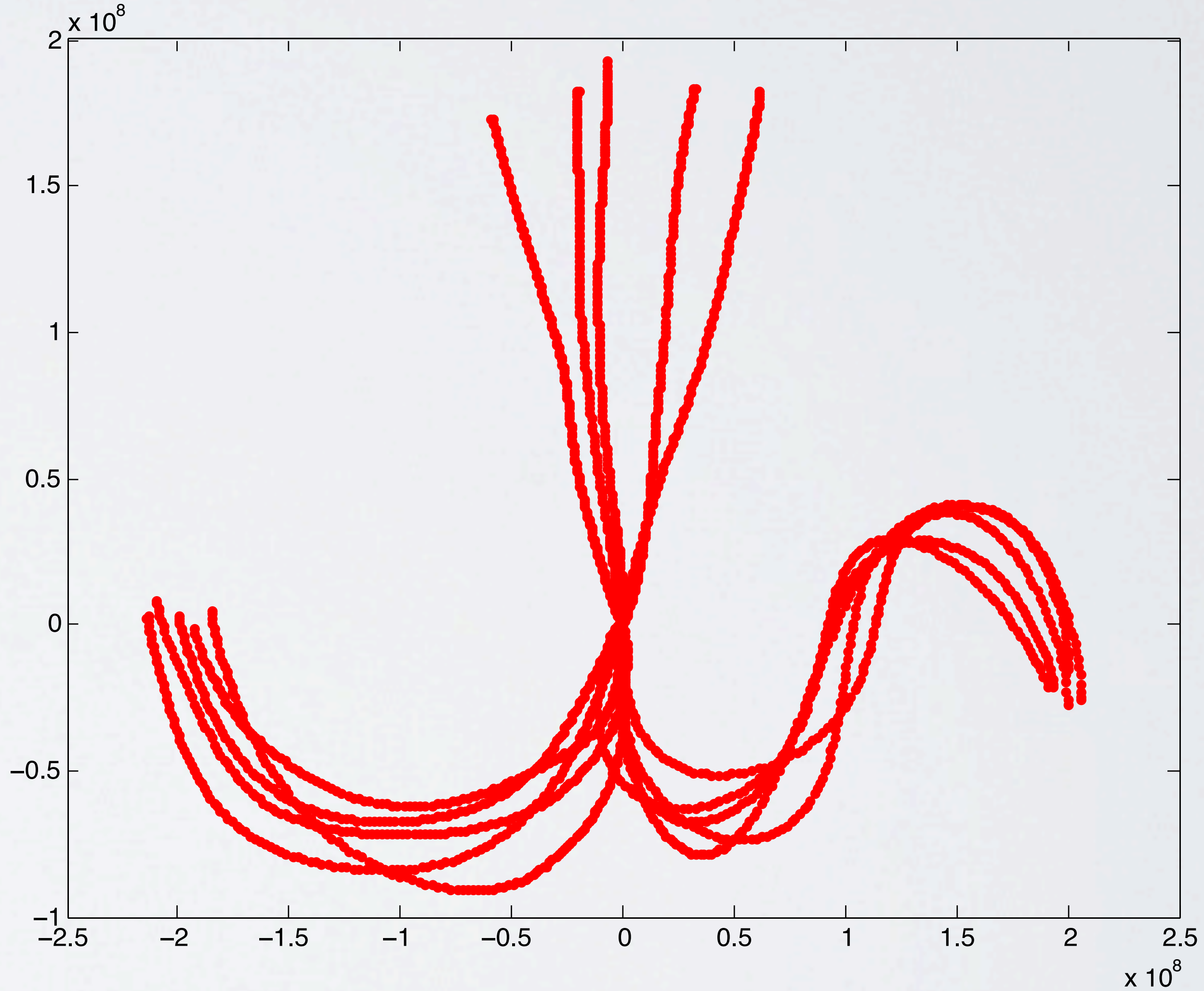
BUT WHAT ABOUT K?

- Someone needs to decide!
- We know that finding K is a principle problem!
    - so that is fine.
- But the big problem is this:
    - the actions to be learned must be known in advance!! *That* is obviously a problem!
- We use Dirichlet process to find K.

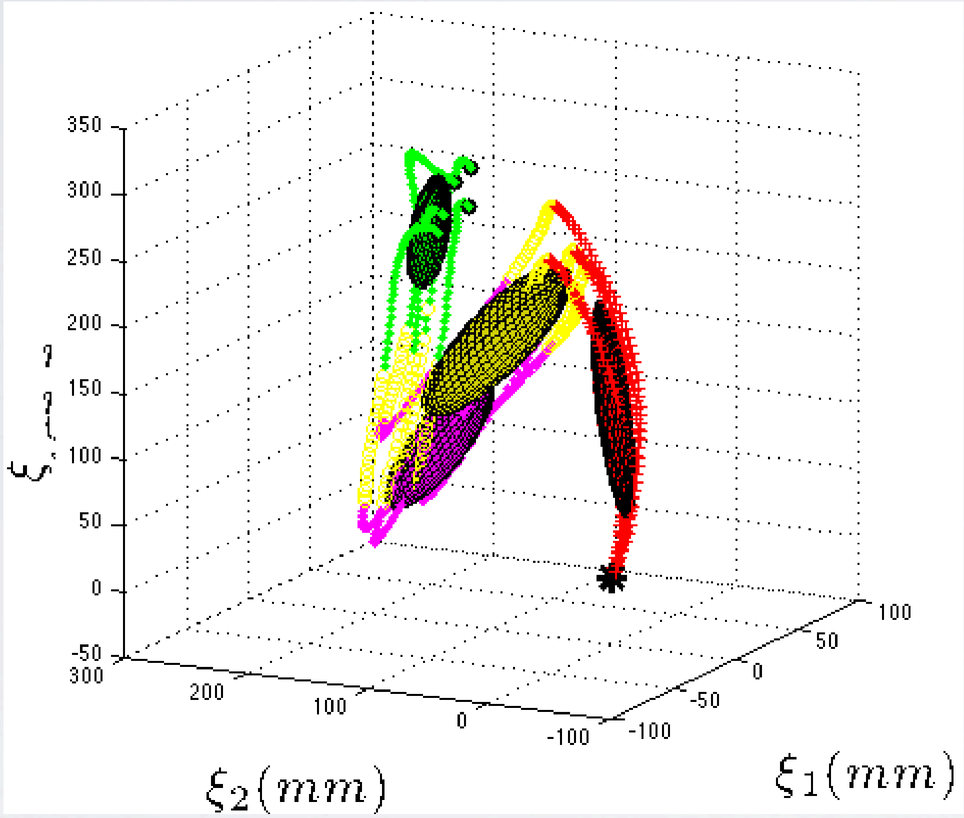


BUT WHAT ABOUT K?

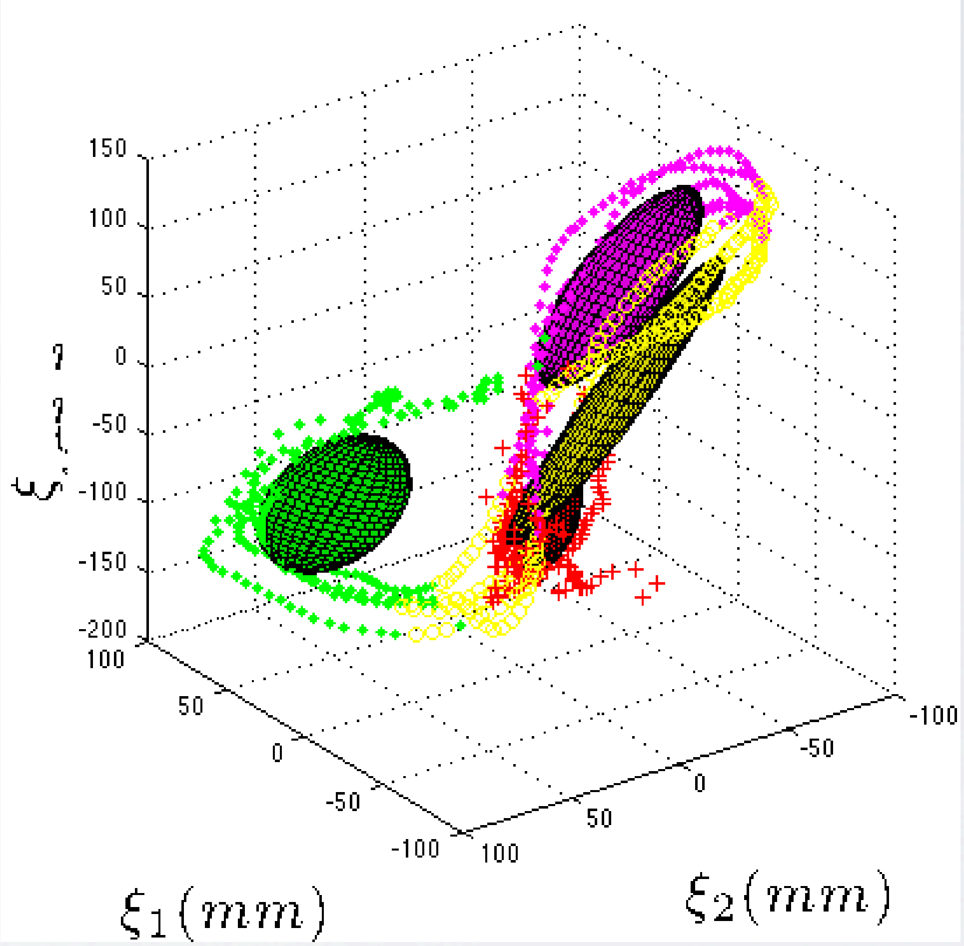
- Someone needs to decide!
- We know that finding K is a principle problem!
  - so that is fine.
- But the big problem is this:
  - the actions to be learned must be known in advance!! *That* is obviously a problem!
- We use Dirichlet process to find K.
  - Application of DPs here is non-trivial.



RESULTS



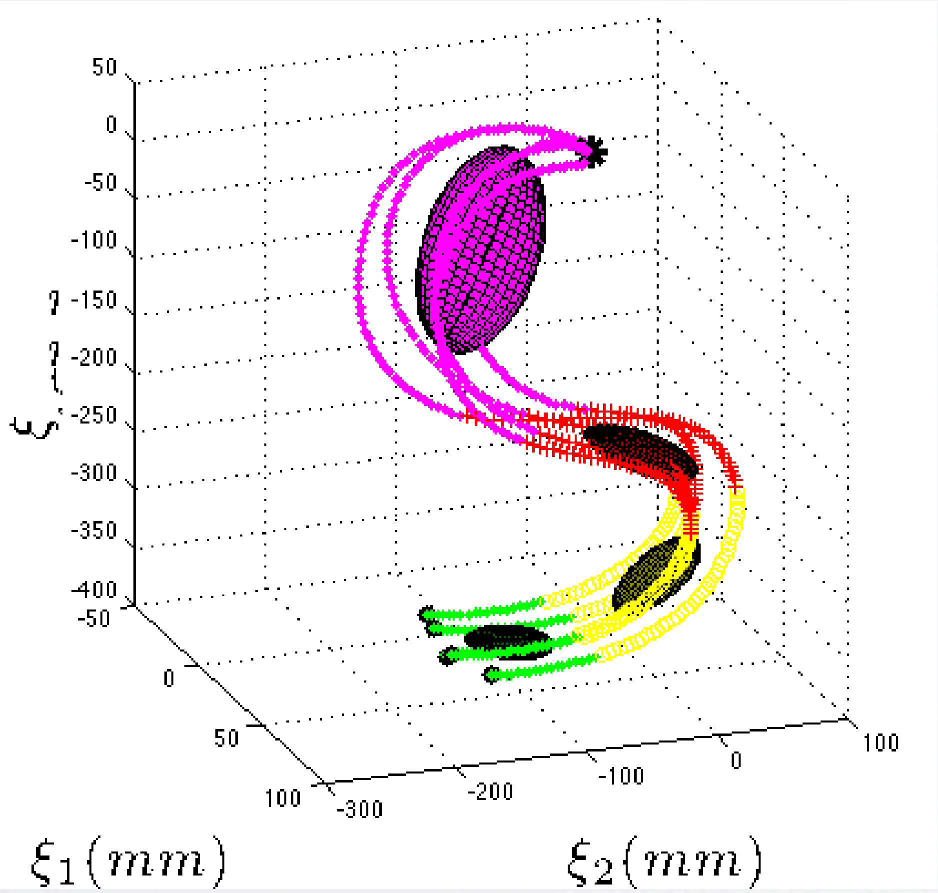
Location



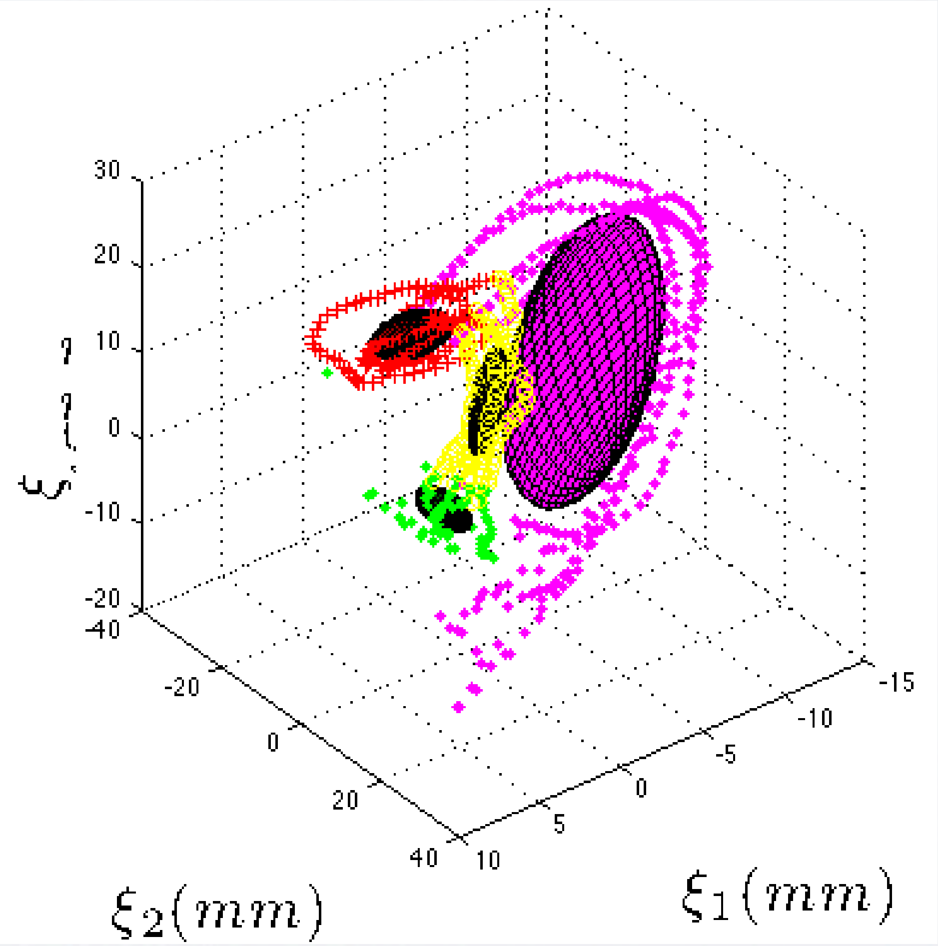
Velocity

- Results for 3D movement "Letter N" captured with iCub: 4 Gaussians

WORKS GREAT!



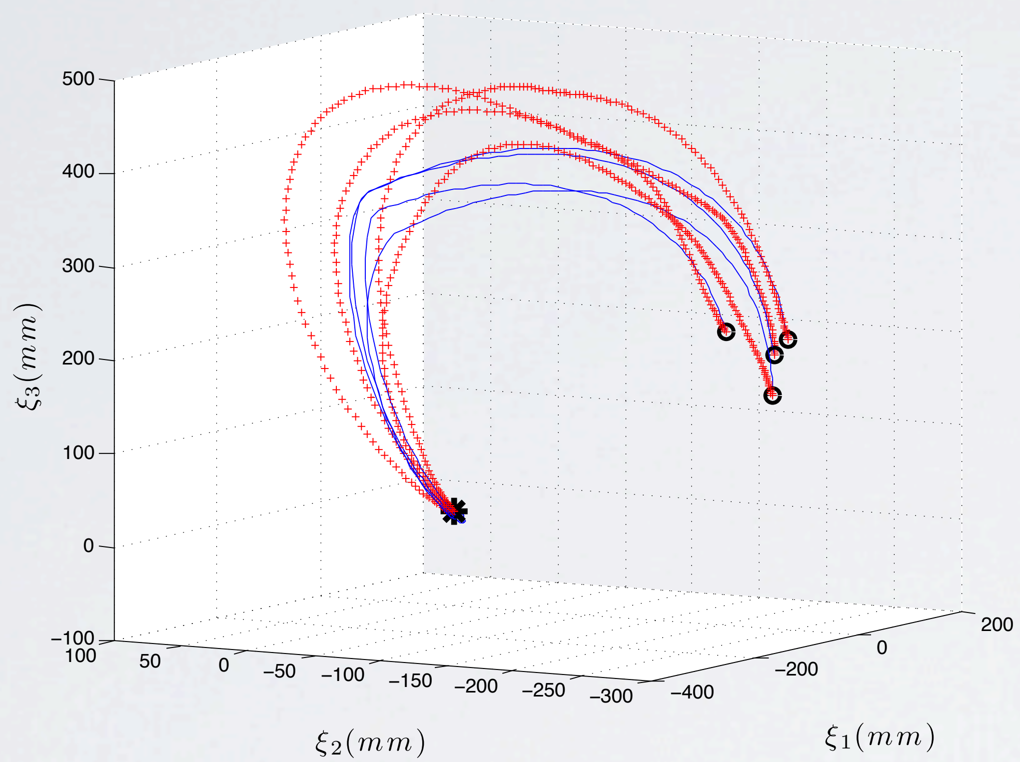
Location



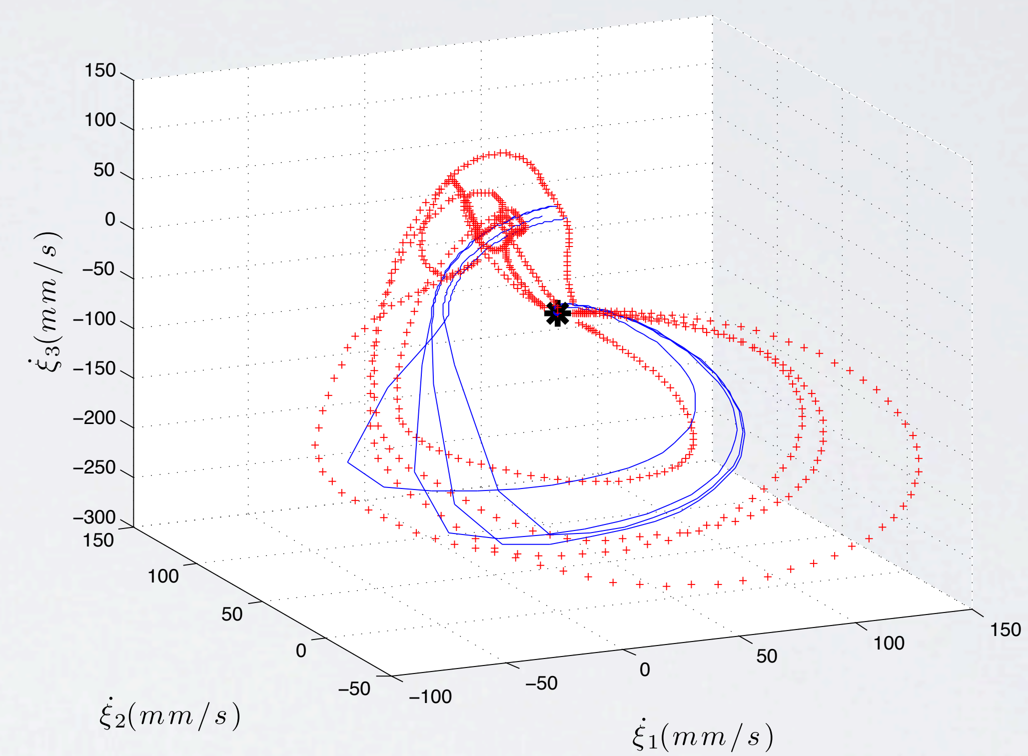
Velocity

- Results for 3D movement "Letter S" captured with iCub: 4 Gaussians

WORKS GREAT!



Location



Velocity

- Results for 3D movement "Letter C": Comparison Training vs Simulation

Krüger *et al.* Imitation Learning of Non-Linear Point-to-Point Robot Motions using Dirichlet Processes. ICRA 2012

SPIRIT OF OACS FOR INDUSTRIAL ROBOTS

SKills are OACS for Industrial Applications

Institut for
Mekanik og Produktion

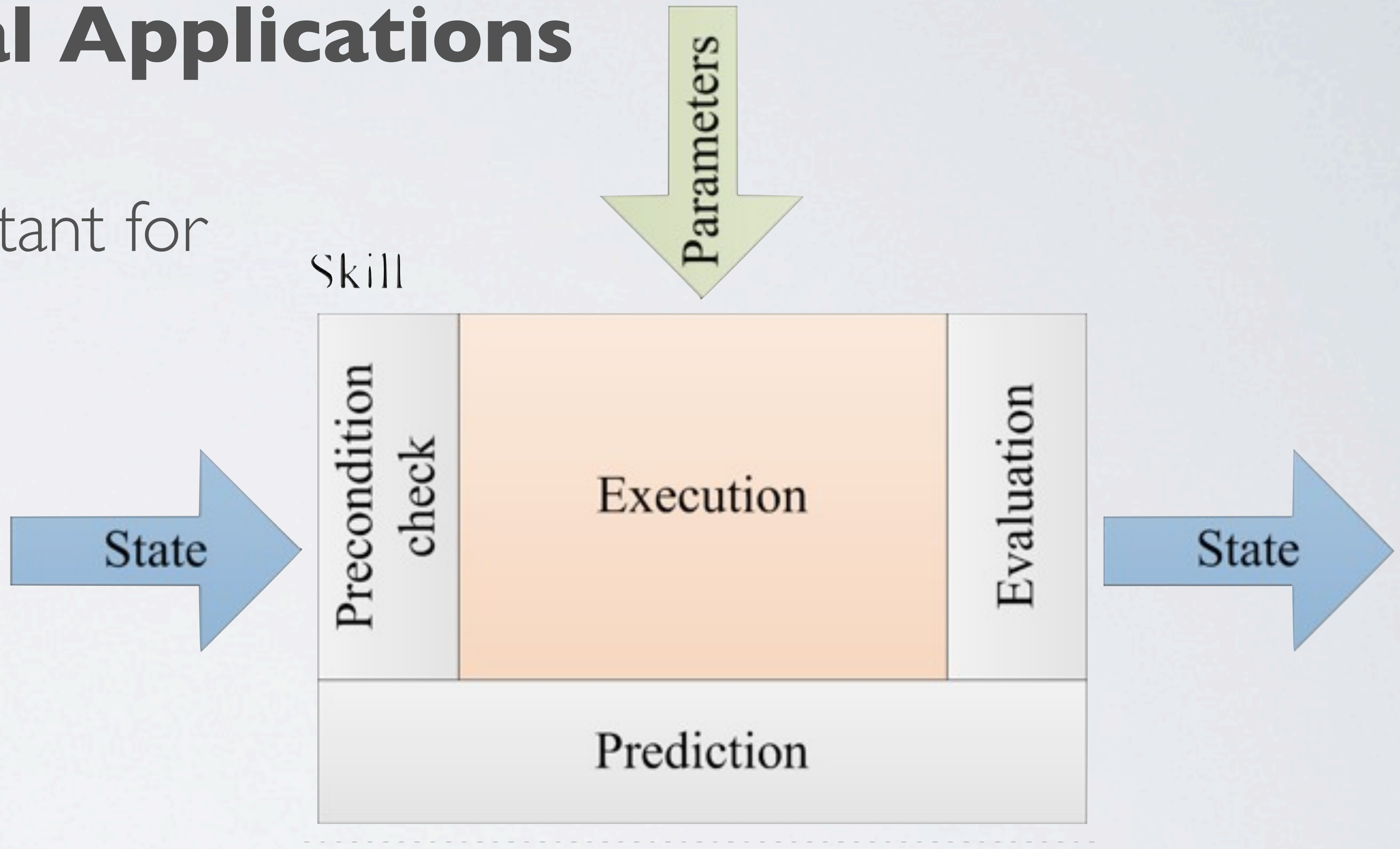
TAPAS

ECHORD

AALBORG UNIVERSITET

SKills are OACS for Industrial Applications

- **pre- and post-conditions**: Important for robustness and planning
  - STRIPS-like planner
  - Markov Decision process



Institut for Mekanik og Produktion

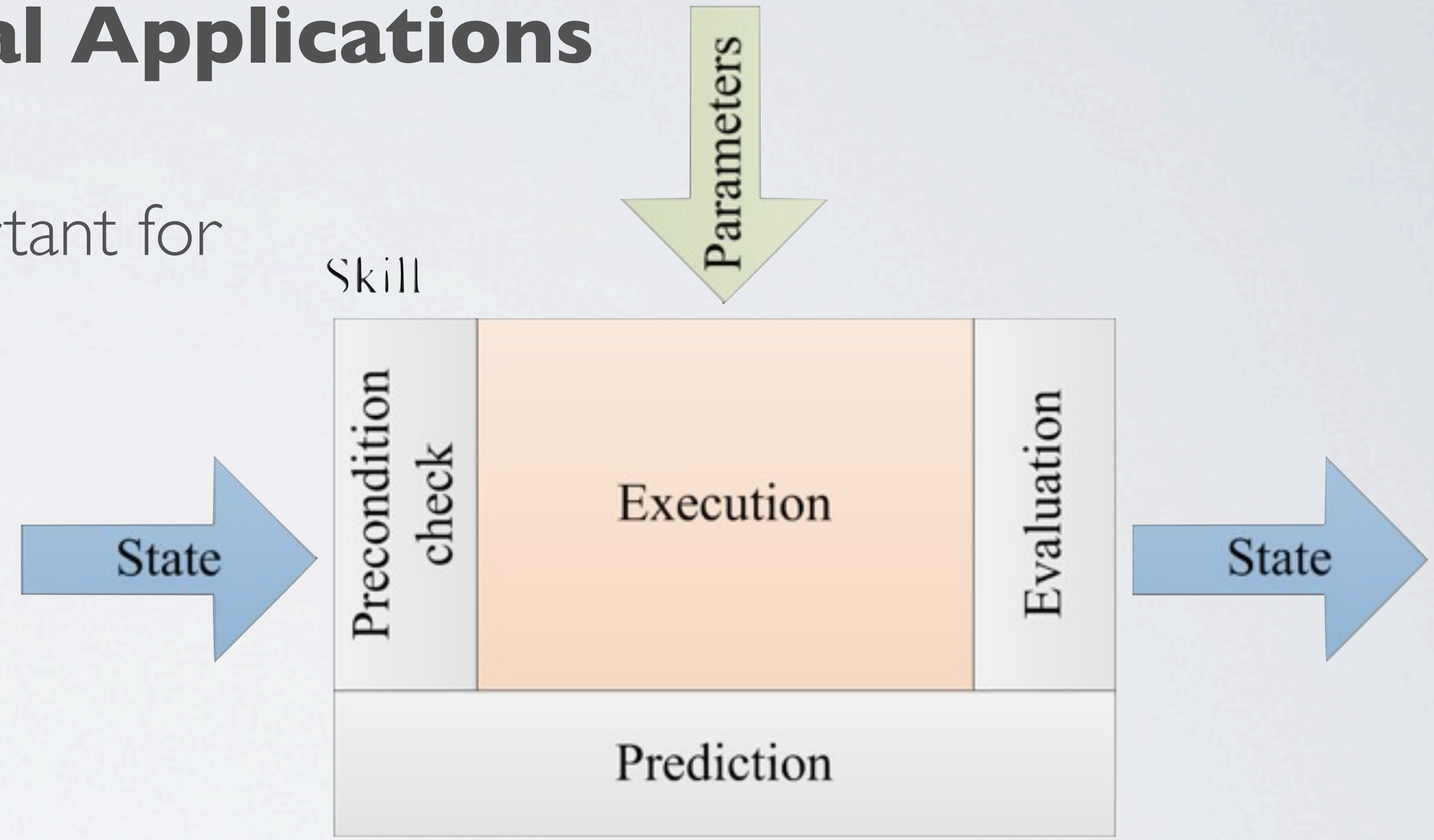
TAPAS

ECHORD

AALBORG UNIVERSITET

SKills are OACS for Industrial Applications

- **pre- and post-conditions**: Important for robustness and planning
  - STRIPS-like planner
  - Markov Decision process



Problem: Finding the right set of skills

VOCABULARY OF TASKS

- analyzed 566 tasks at Grundfos
  - task implementations
  - standard operation procedures (SOPs)

Logistic	Assistive	Service
Transportation	Machine Tending	Maintenance, Repair and Overhaul
Multiple Part Feeding	Assembly	Cleaning
Single Part Feeding	Inspection	
	Process Execution	

VOCABULARY OF SKILLS

6 Transportation Skills	10 Assistive Skills
• Move to <location> • Locate <object> • Pick up <object> • Place <object, coord> • Unload <container, coord> • Shovel <container, coordinate>	• Pick up <object> • Place <object, coordinate> • Locate <object> • Press <object> • Check <object> • Align <object, object> • Open <object> • Close <object> • Release <object> • Turn <object>

Skill	**Description**
Move to	To go from one location (station) to another in the factory
Locate	To determine or specify the position of an object by searching or examining
Pick up	To take hold of and lift up
Place	To arrange something in a certain spot or position
Unload	Unload a container: to remove, discharge or empty the contents from a container
Shovel	To take up and move objects with a shovel
Check	Quality control: the act or process of testing, verifying or examining
Align	To make an object come into precise adjustment or correct relative position to another object
Open	To move (as a door) from a closed position and make available for entry, passage or accessible
Close	To move (as a door) from an open position
Press	To press against with force in order to drive or impel.
Release	To let go or set free from restraint e.g. release a button
Turn	To turn a knob or handle

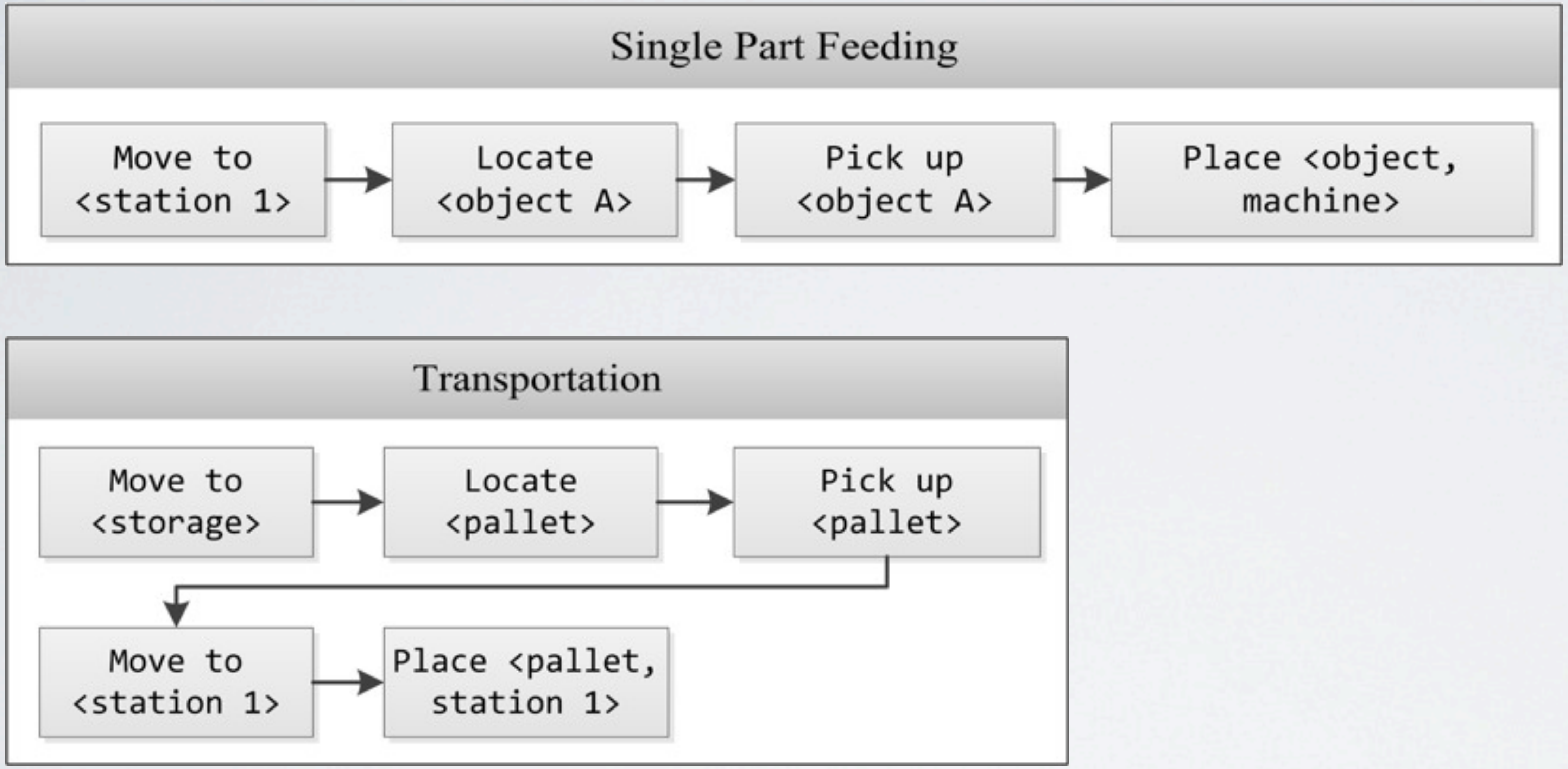
Skill-complete with 13 skills

TAPAS ECHORD

Institut for Mekanik og Produktion

AALBORG UNIVERSITET

PROGRAMING USING SKILLS



Institut for
Mekanik og Produktion

TAPAS

ECHORD

AALBORG UNIVERSITET

SENSING IS THE KEY

SENSING IS THE KEY

Grasping Skill(object)

prior state

- object is in field of view --> provides 3D coord
- object is graspable (use 3D coord)
- distance to object
- grasping trajectory exists

execute grasping trajectory. Use force torque to already after the actual grasp verify for success

posteriori:

- object is in the gripper
- location within the robot body space

Institut for Mekanik og Produktion

TAPAS

ECHORD

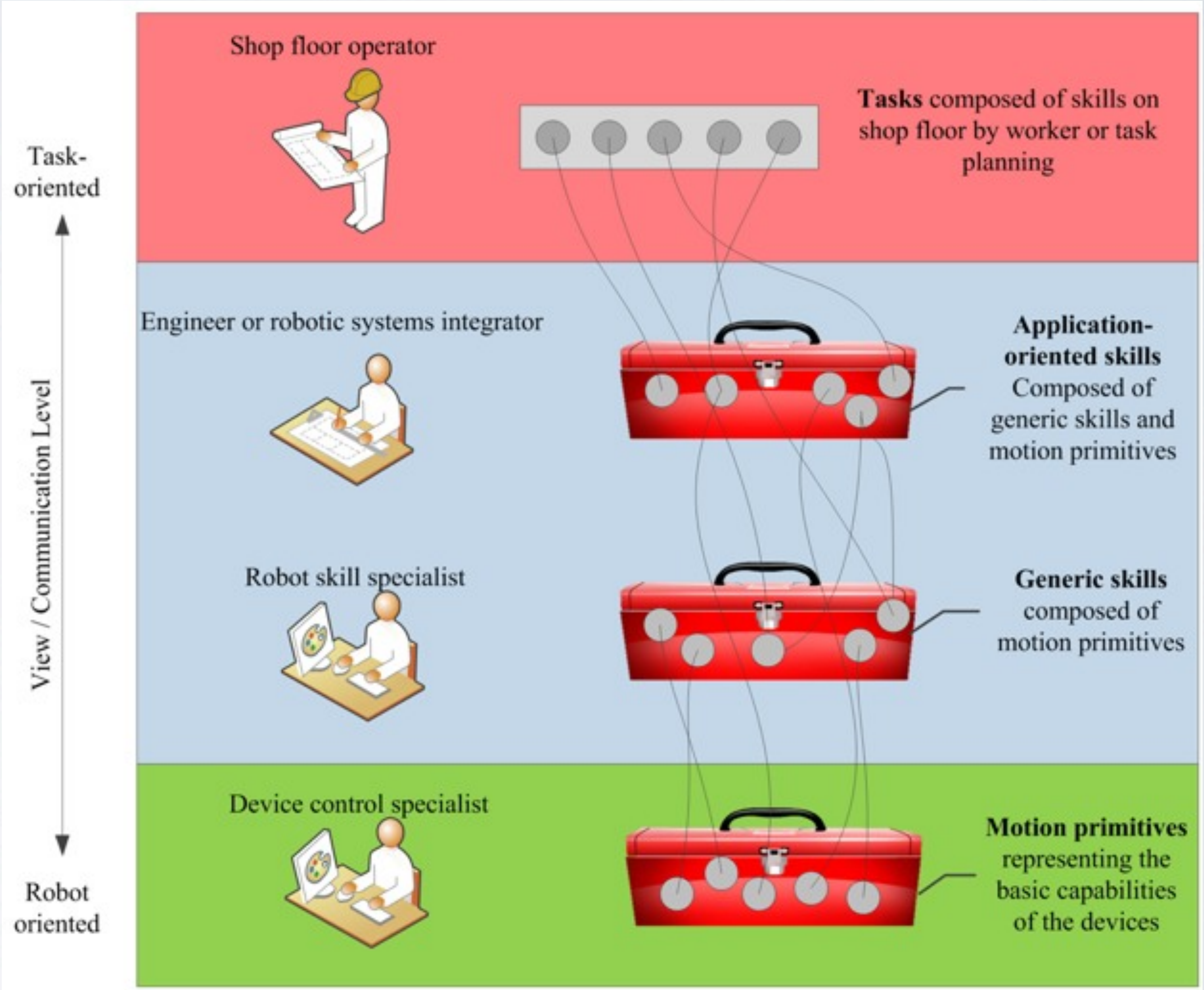
AALBORG UNIVERSITET

PLACE-SKILL(LOCATION)

- **prior**
  - location in {table, shelf, magazin} (location given as bar code)
  - empty <location> is available and reachable
  - is the gripper free?
- **execute** place skill + simultaneous verification using force torque
- **poterior**
  - empty gripper
  - location not empty anymore
  - arm back in robot body space (note: breaching the robot body space: moving skills may not lead to a breach, but manipulation skills may)

IMPLEMENTATION LAYERS

- Generalized Plans on different levels



PROGRAMING USING SKILLS

- programs can be generated automatically using a planner
  - probabilistic using Markov decision process
  - binary using STRIPS planner

See the demo here:
[feeding Demo.mov - YouTube]

PROGRAMING USING SKILLS

- programs can be generated automatically using a planner
  - probabilistic using Markov decision process
  - binary using STRIPS planner

See the demo here:
feeding Demo.mov - YouTube

- gripper: full/empty
- magazin: filling level
- robot location: discrete:
  - home
  - warehouse
  - feeder 1,2,3...
- SLC full/empty
- feeder: empty, apparently full, full

SUMMARY + CONCLUSIONS

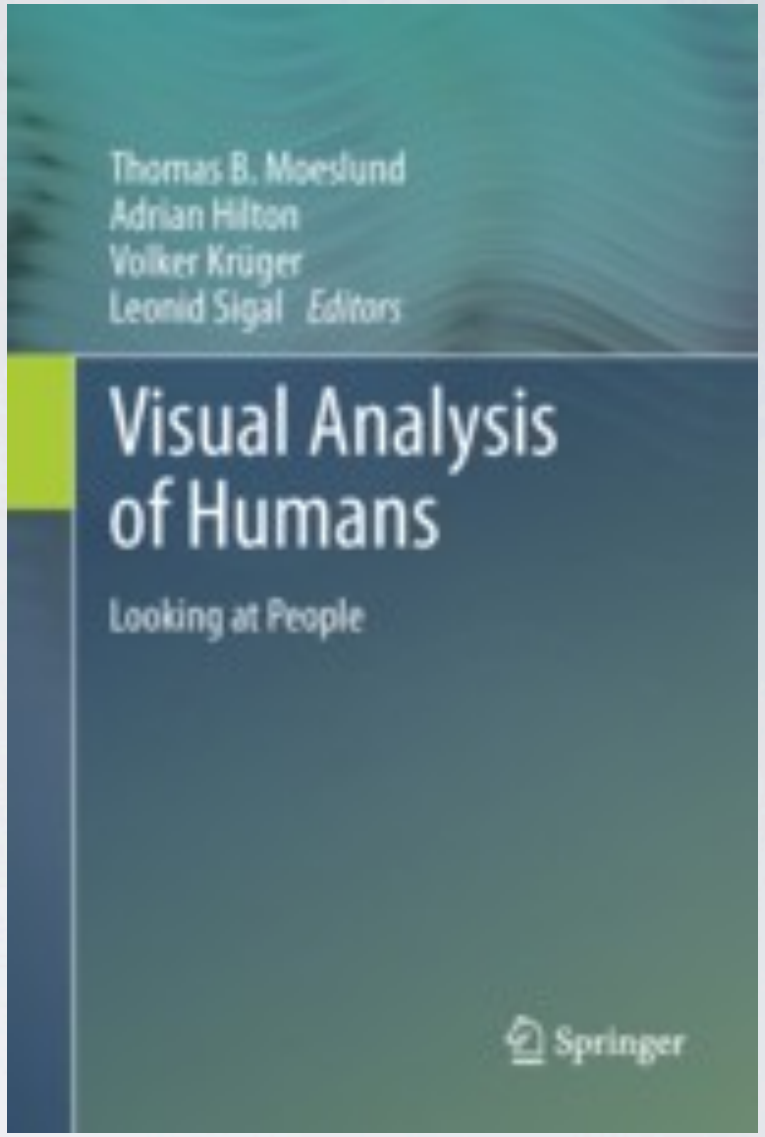
- Object-Action Complexes
  - Unsupervised learning of action grammar based on effects of the observed actions
  - Modeling of human actions using SEDS-DMPs and PHMMs
  - **only tested on simple scenarios, not clear how well it will scale**
- hand-generated "OCAs" / Skills for industrial scenario
- Are OACs are good choice for industrial applications?
  - What about assembly tasks?
  - How should the skills be for collaboration? event-driven rather than effect-driven?

REFERENCES

- The meaning of action
  - V. Krueger, D. Kragic, A. Ude, and C. Geib. Meaning of Action. Int. Journal on Advanced Robotics, Special issue on Imitative Robotics, T. Inamura and G. Metta (eds.), 2007.
  - A. Bobick and V. Krüger. Visual Analysis of Humans, T. Moeslund ,A. Hilton ,V. Krüger and L.Segal (Eds.), chapter On Human Action, pages 279–288. Springer, 2011.
- Action Primitives
  - Danica Kragic Dana Kulic and Volker Krüger. Visual Analysis of Humans, In T. Moeslund, A. Hilton ,V. Krüger and L.Segal (Eds.), chapter Learning Action Primitives, pages 333–353. Springer, 2011.
  - Sanmohan, V. Krüger, D. Kragic, and H. Kjellström. Automatic Primitive Segmentation and Action Recognition. Advanced Robotics, 25(6-7):871– 891, 2011.
  - V. Krueger, Sanmohan, D. Herzog, A. Ude, and D. Kragic. Learning Actions from Observations. IEEE Robotics and Automation Magazine, 17(2):30–43, 2010.
  - Simon Bøgh, Oluf Nielsen, Mikkel Pedersen, Volker Krüger: Does your Robot have Skills?. 43rd International Symposium on Robotics (ISR)
- Modeling, recognition and generation of Actions
  - Dennis Herzog and Volker Krueger. Tracking in Actionspace. In Human Motion: Understanding, Modeling, Capture and Animation, Workshop at ECCV 2010, 2010. Springer.
  - Dennis Herzog and Volker Krueger. Tracking in Object-Action Space. Int. Journal Computer Vision and Image Understanding (CVIU). submitted
  - D. Herzog, A. Ude, and V. Krueger. Motion Imitation and Recognition using Parametric Hidden Markov Models. In Humanoids, IEEE-RAS In- ternational Conference on Humanoid Robots, Daejeon, Korea, South, December 1-3, 2008.
  - Volker Krüger, Vadim Tikhanoff, Lorenzo Natale and Giulio Sandini: Imitation Learning of Non-Linear Point-to-Point Robot Motions using Dirichlet Processes IEEE 2012 Conf. of Robotics and Automation, ICRA 2012
  - Carsten Høilund, Volker Krüger and Thomas Moeslund. Evaluation of Human Body Tracking System for Gesture-based Programming of Industrial Robots ICIEA2012
- Systems
  - Carsten Høilund, Mikkel Pedersen and Volker Krüger. Using Human Gestures to Program Generic Skills for a Mobile Robot Arm in a Feeder Filling Scenario. ICRA ECHORD-Workshop, 2012.

THANKS

Questions? Comments?



www.m-tech.aau.dk/Research+Groups/Robotics+and+Automation/

Institut for
Mekanik og Produktion

AALBORG UNIVERSITET