Cognitive Primitives for Mobile Robots

Development with the Tekkotsu framework

Ethan Tira-Thompson
David S. Touretzky

Carnegie Mellon University
School of Computer Science

THE ROBOTICS INSTITUTE

What is Tekkotsu?

- Development framework for robotic applications
  - Handles common tasks to avoid reinvention
  - Provides libraries of utility code with integrated interfaces to speed new development
  - Increase communication and code reuse by providing a common platform between groups
  - Enable high level robotics education

What is Tekkotsu?

Behaviors request lock on MotionCommands
to make direct function calls on them

MainObj

EventRouter
`erouter`

Vision
`vision`

System sends state information (~32ms)

System sends camera frames (~30fps)

WorldState
`state`

MotionManager
`motman`

Requests joint positions

Returns positions based on current MotionCommands

MotoObj

System requests joint positions (~32ms)

Sends new joint positions to system

Requests sound buffer

SoundPlay

System requests sound buffer (~32ms)

Returns sound buffer by mixing current sounds

Sends 32 ms of sound to system

Behaviors can play sounds anytime

created by currently active Behaviors

Can access state anytime for reactive/open loop control

Added to Motion Manager

MotionCommands
(dynamically created)

SoundManager
`sndman`

Can play sounds at any time

Key

- Pre-emptive Process
- Shared Memory Region
- Unshared Global Variable

TinyFTPD

From Sony's sample code, allows you to FTP files during run time.

Self contained

What is Tekkotsu?

- Written in C++
- Event-based architecture
- Extensive use of templates and inheritance
- High performance, real time
- Open Source - LGPL
- See Tekkotsu.org for code and documentation

Why AIBO?

- AIBO is currently a unique platform
  - Highly articulated (18 joints)
  - Significant processing power
    - 576 MHz MIPS, 64MB RAM

ERS-7

  - Lots of sensors (Color camera, 3 IR range-finders, 3-axis accelerometer, stereo microphones, array of buttons)
  - 802.11b wireless ethernet
  - Small, light - “desktop” development
  - Affordable! $1799 before 6% academic discount

Sample Camera Image

Interactions Between Robotics and Cognitive Science

- Cognitive Scientists can explore practical applications of their ideas
- Roboticists can draw new inspiration from cognitive theories

Interactions Between Robotics and Cognitive Science

- Cognitive Scientists can explore practical applications of their ideas
- Roboticists can draw new inspiration from cognitive theories

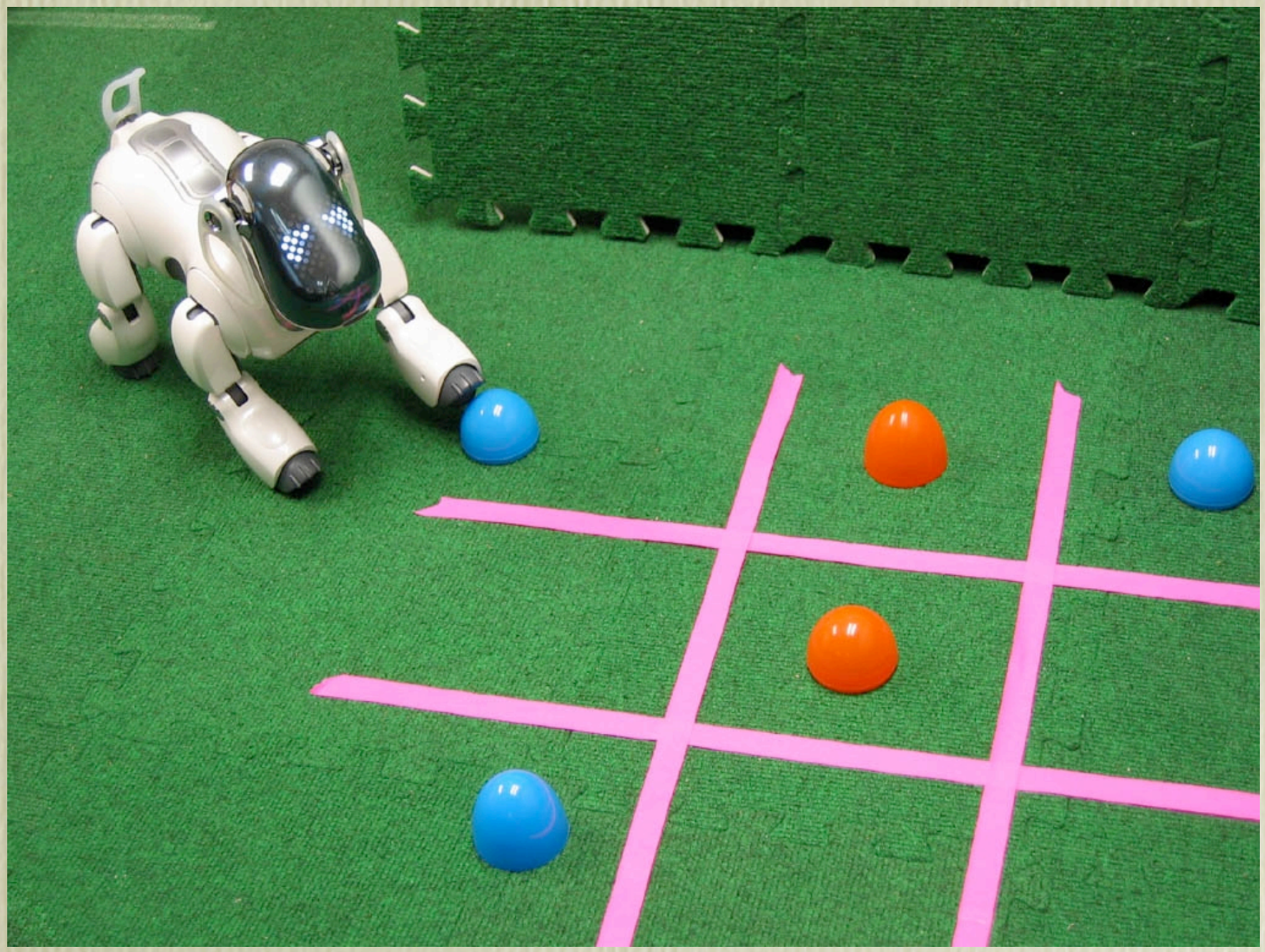
Primitives for Cognitive Robotics

- Perception (Vision)
- Mapping
- Manipulation
- Control (state machines, subsumption, etc.)
- Attention
- Learning
- Human-Robot Interaction

Primitives for Cognitive Robotics

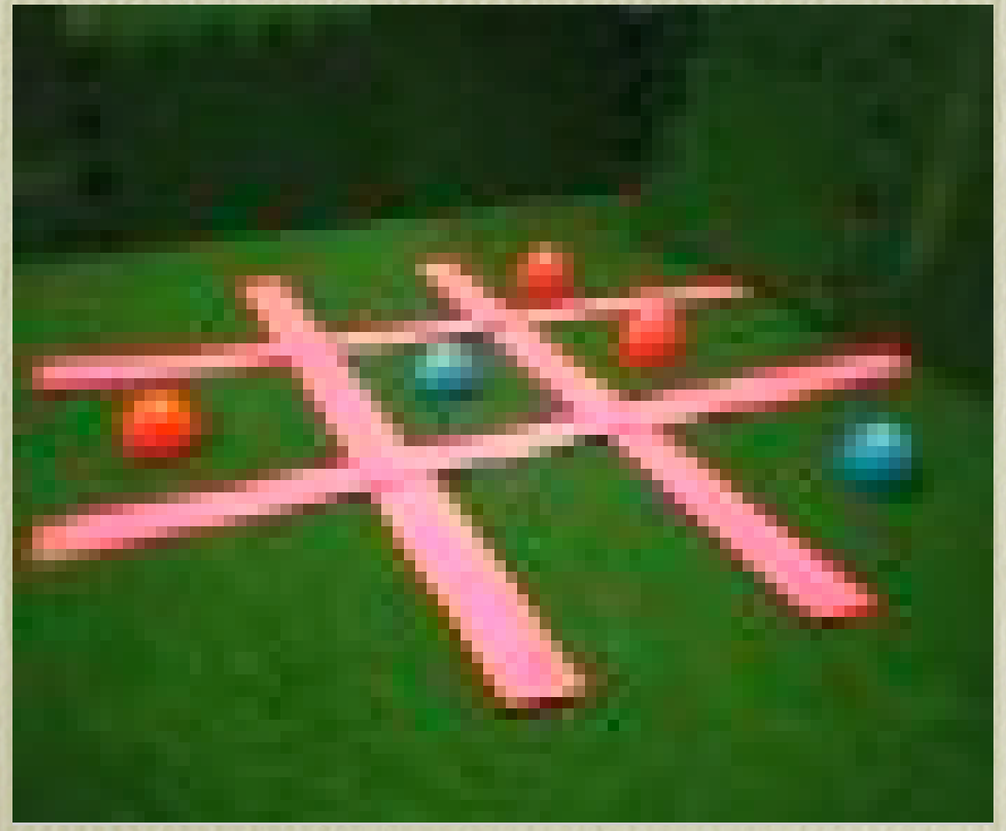
- Perception (Vision)
- Mapping
- Manipulation
- Control (state machines, subsumption, etc.)
- Attention
- Learning
- Human-Robot Interaction

Touchstone Task: Tic-Tac-Toe



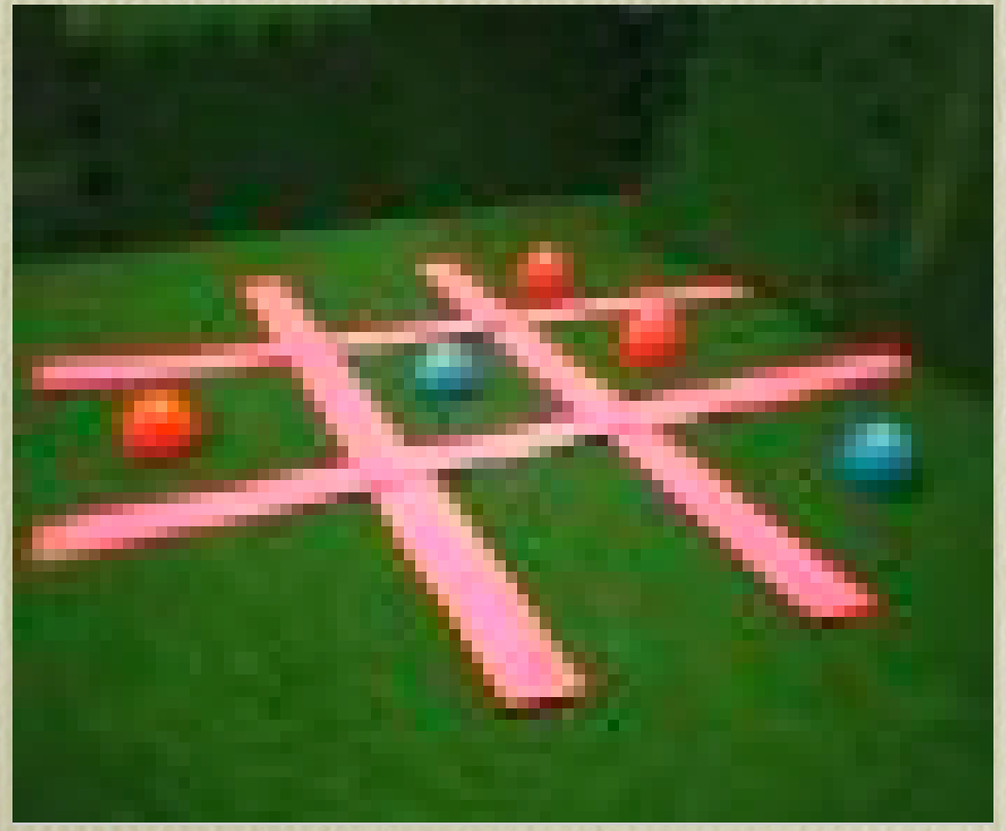
Primitives for Cognitive Robotics

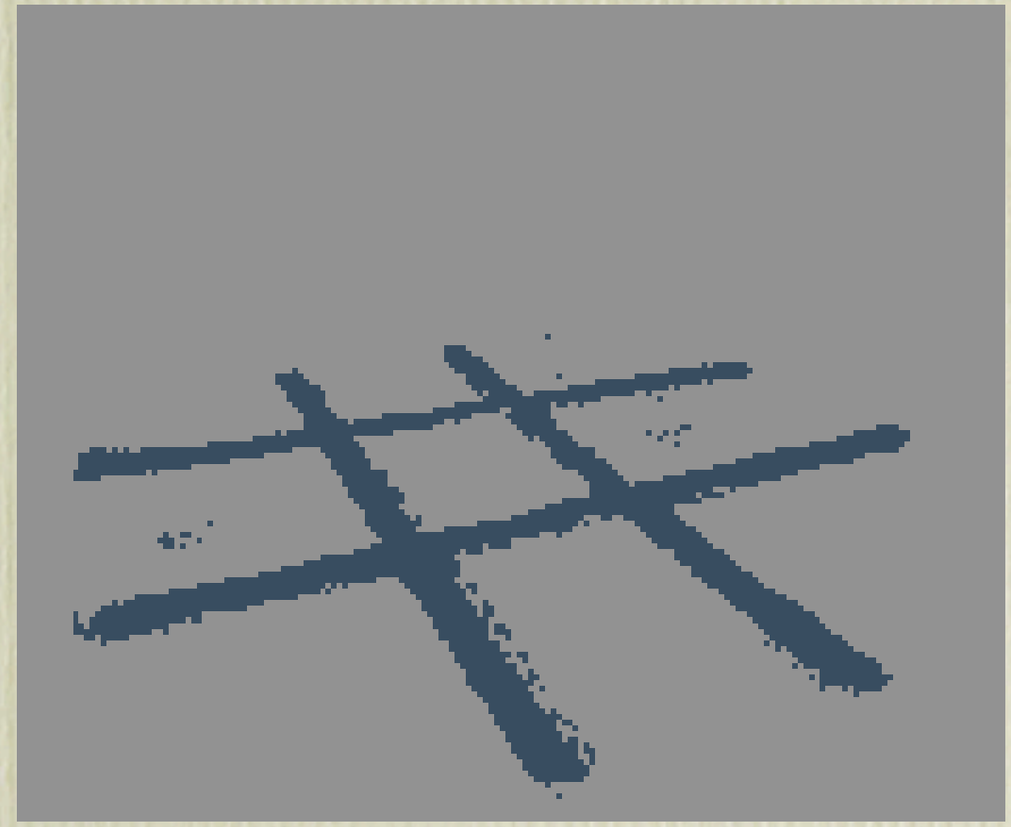
- Perception
  - Visual Routines [S. Ullman, Cognition, 1984]



Primitives for Cognitive Robotics

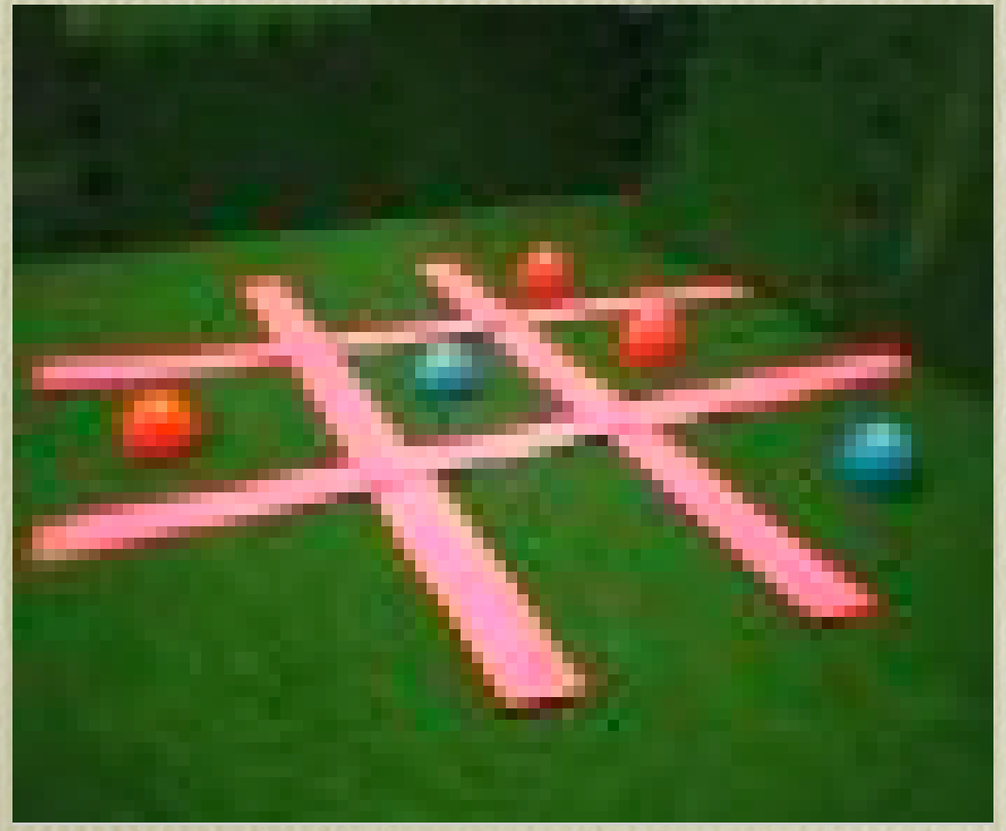
- Perception
  - Visual Routines [S. Ullman, Cognition, 1984]

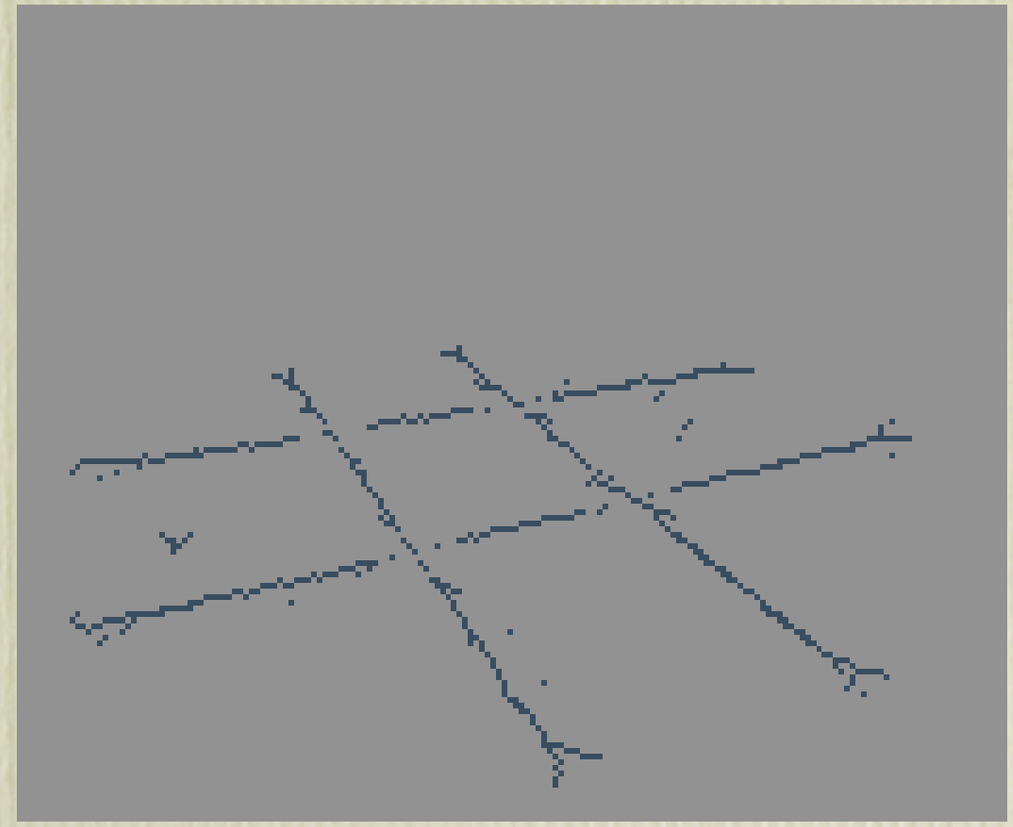




Primitives for Cognitive Robotics

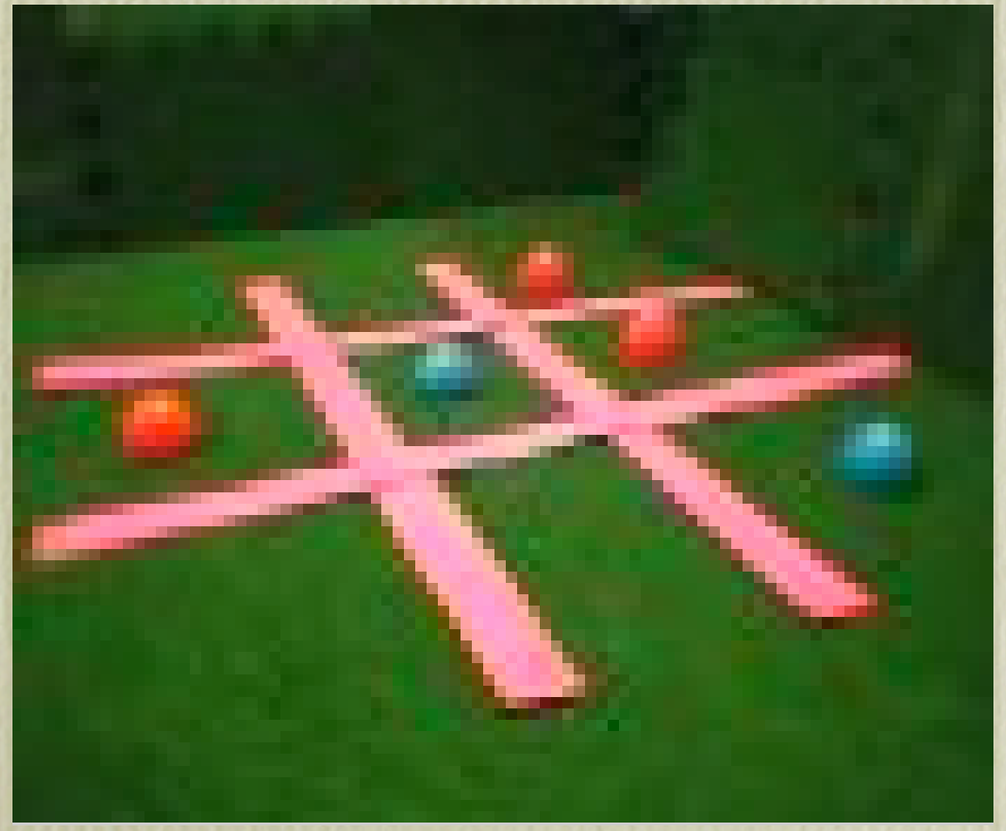
- Perception
  - Visual Routines [S. Ullman, Cognition, 1984]

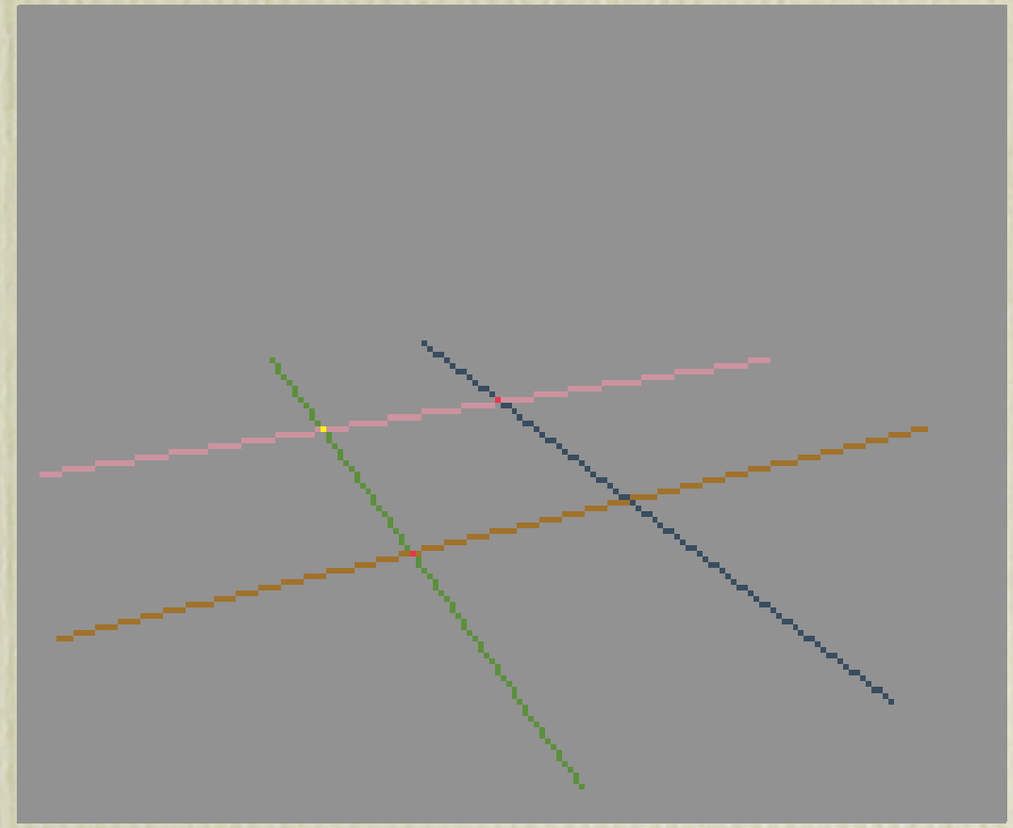




Primitives for Cognitive Robotics

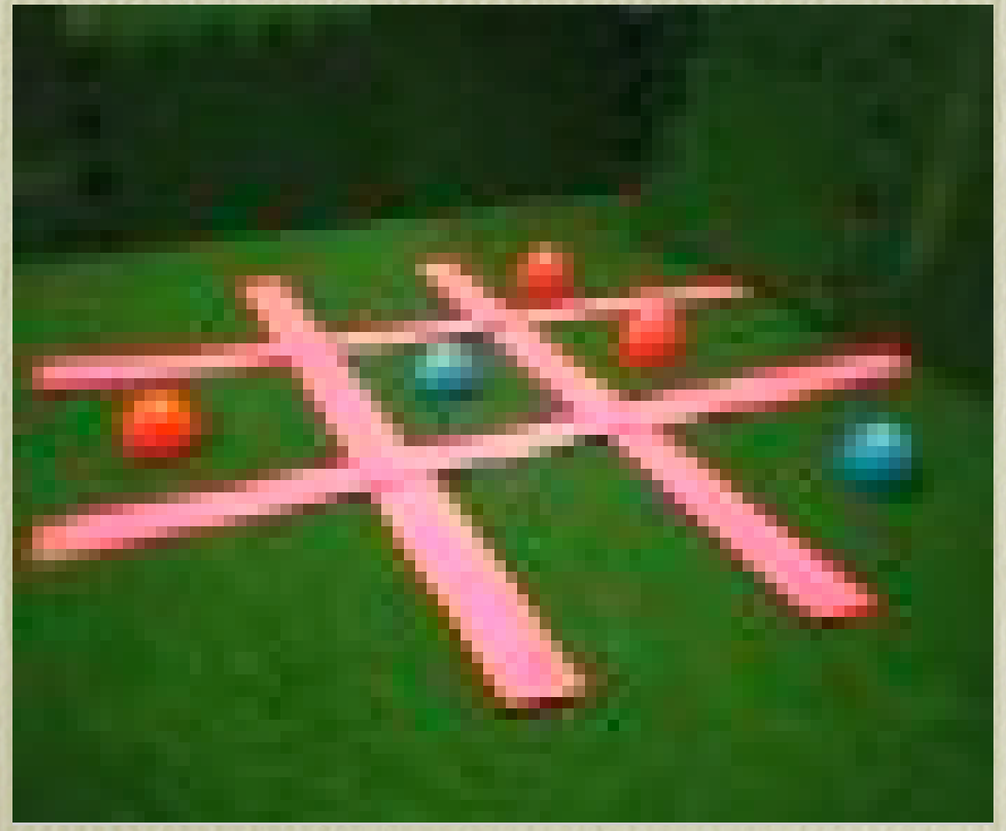
- Perception
  - Dual Coding Theory [Paivio, Mental Representations, 1986]

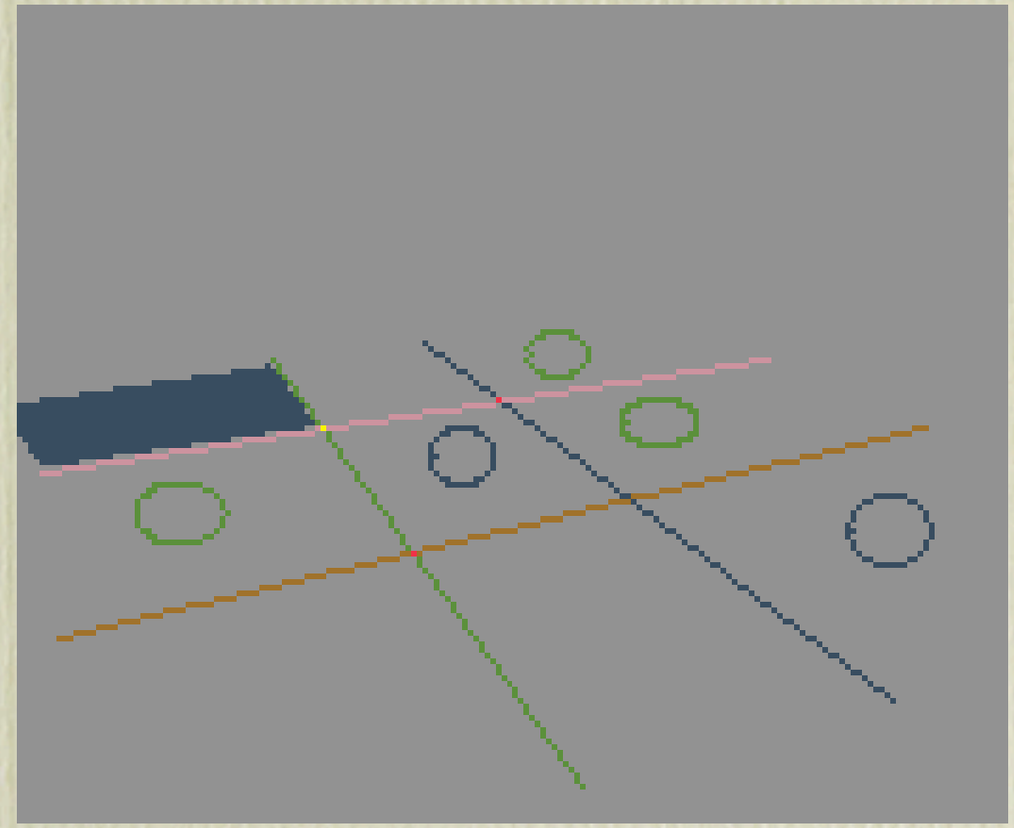




Primitives for Cognitive Robotics

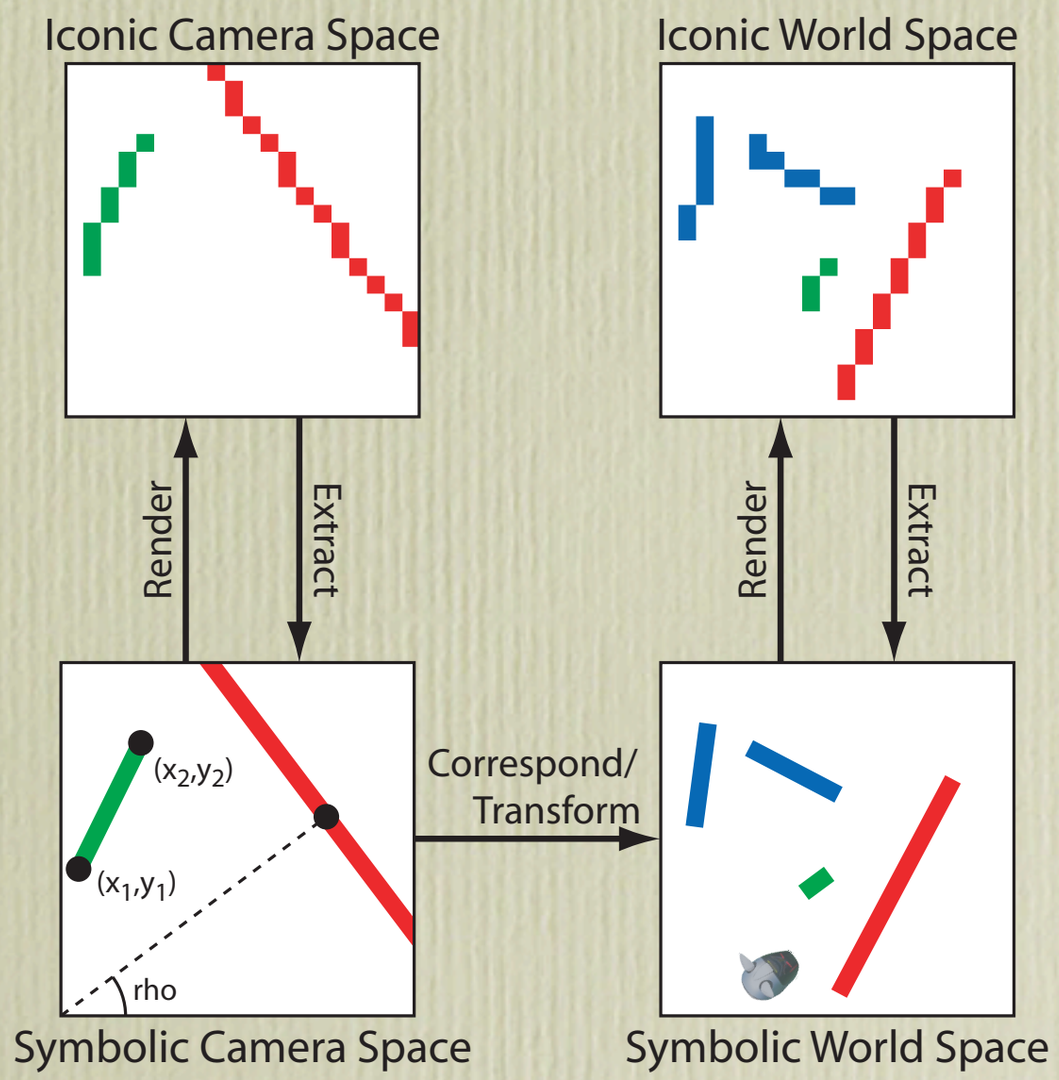
- Perception
  - Dual Coding Theory [Paivio, Mental Representations, 1986]





Primitives for Cognitive Robotics

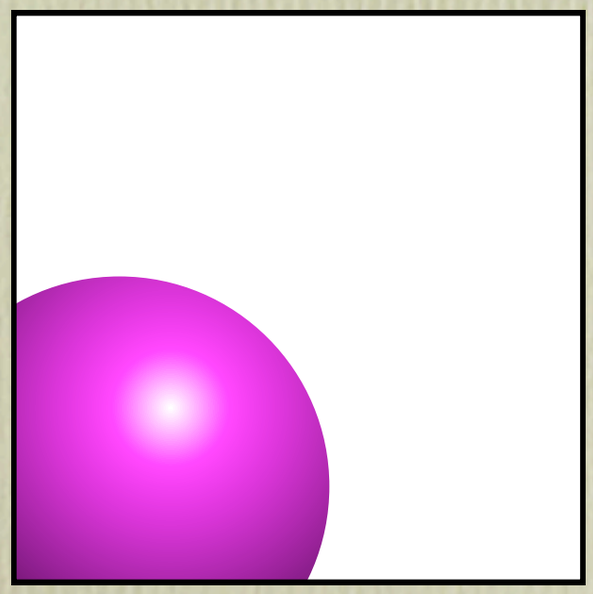
- Perception
  - Mapping and Navigation



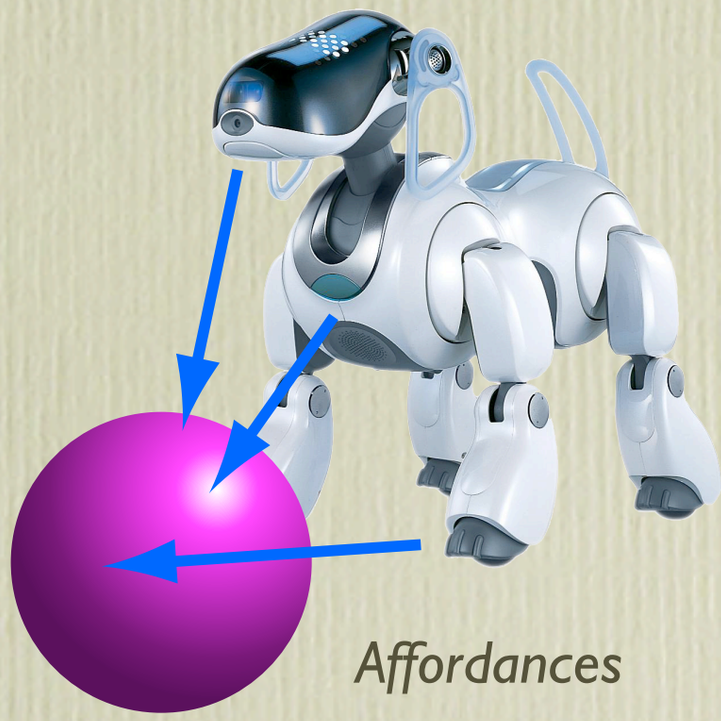
Primitives for Cognitive Robotics

- Perceiving objects in terms of the actions that can be performed upon them (door knob/grab twist)

- Affordances [J.J. Gibson, The Theory of Affordances, 1977]



Camera Image



Affordances

Current Development

- Manipulation Engine
  - Point of contact
  - Type of contact (ballistic impact, steady pressure, rolling swipe, etc.)
  - Path to follow
  - Type of visual monitoring
  - Motion constraints (e.g. areas or objects to avoid)

Future Primitives

- Planning: a higher level approach to control structure
  - Attentional control: where should the AIBO be looking now?
  - Learning: we would like to support experimentation with different learning architectures, such as SOAR or ACT-R
- Modeling human POV (Schultz)
  - Stay for the next talk! :)

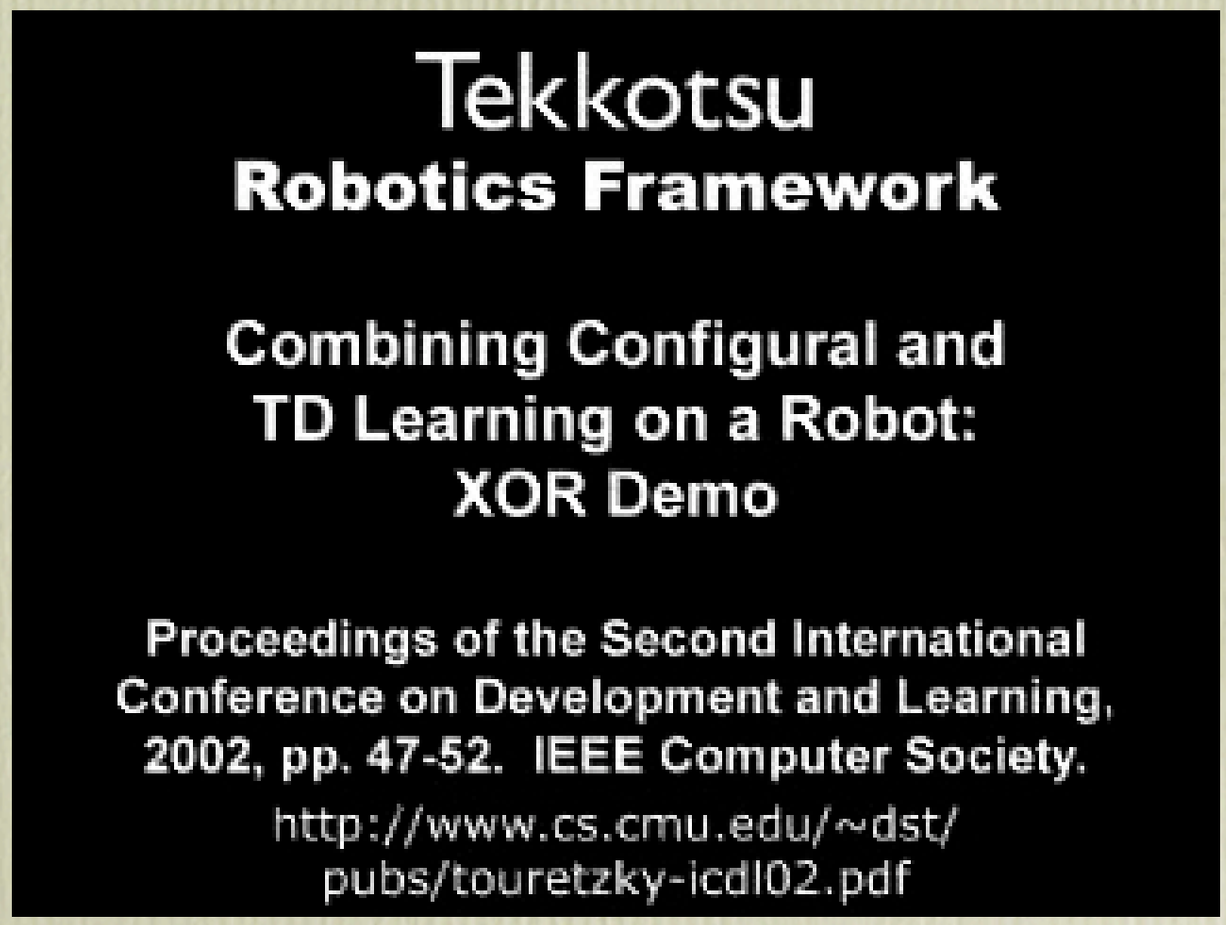
Teaching Robot Programming

- We're creating a cognitive robotics course to be taught in January 2006

- Focusing on teaching robotic programming at a high level using the cognitive primitives described in this talk

AIBO Home Movies

- Training XOR using a variant of Temporal Difference Learning

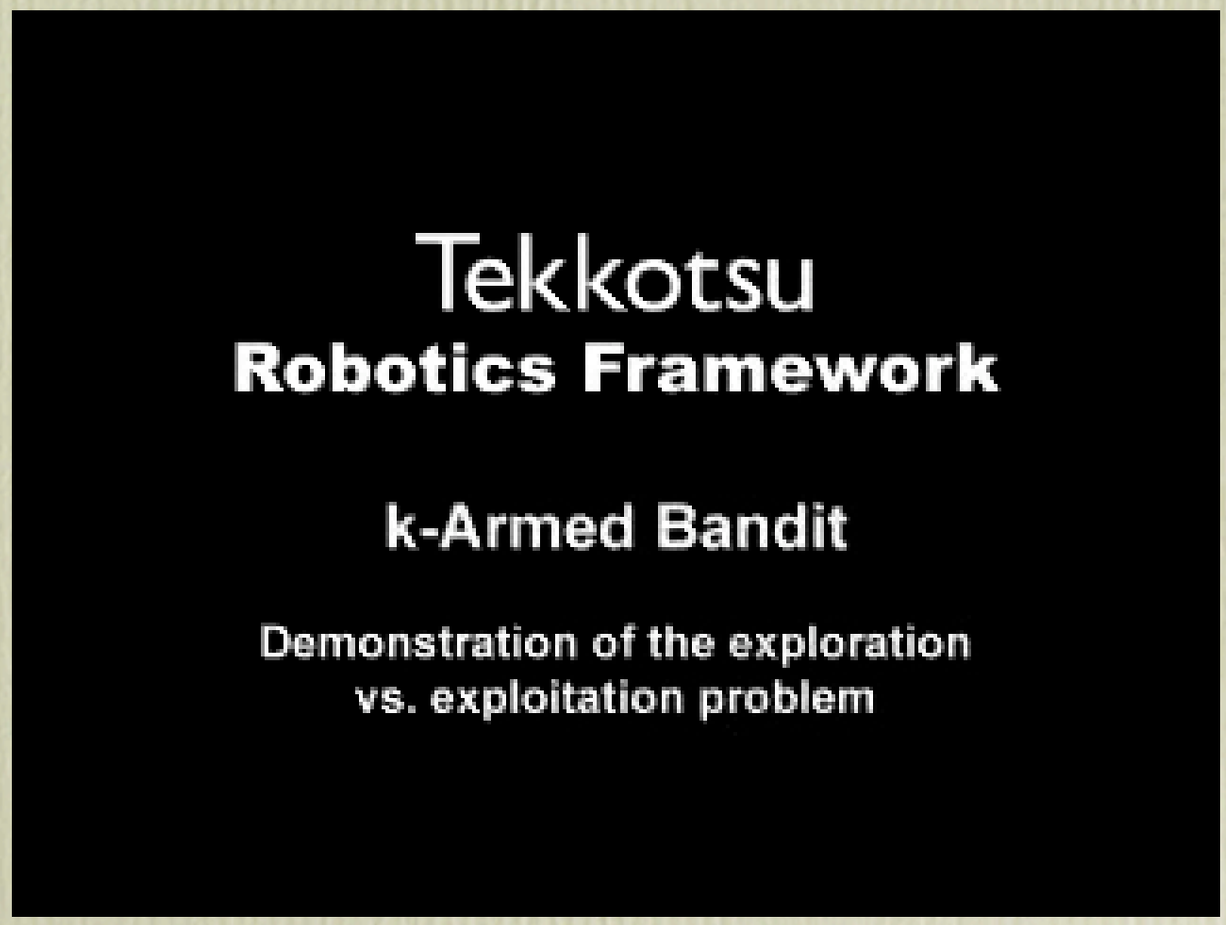


Movie Clip available from:

`http://www-2.cs.cmu.edu/~tekkotsu/Samples.html#TD-learning_XOR`

AIBO Home Movies

- k-Armed Bandit (exploration vs. exploitation)
  - Just a homework assignment ;-)



Movie Clip available from:
`http://www-2.cs.cmu.edu/~tekkotsu/Samples.html#k-Armed_Bandit`

Thanks!

Questions?