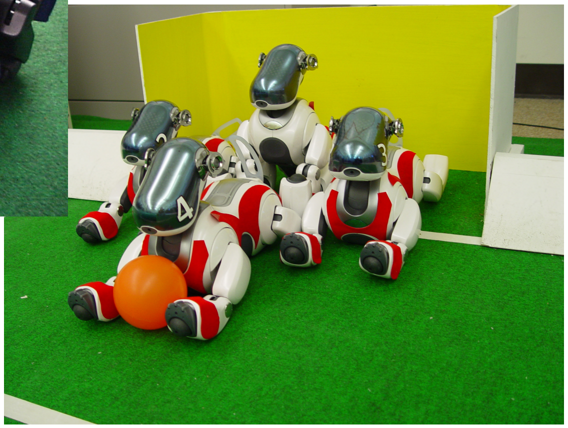
CMRoboBits: Creating an Intelligent AIBO Robot

Paul E. Rybski

Thanks to: Prof. Manuela M. Veloso,
Scott Lenser,Douglas Vail, Sonia Chernova

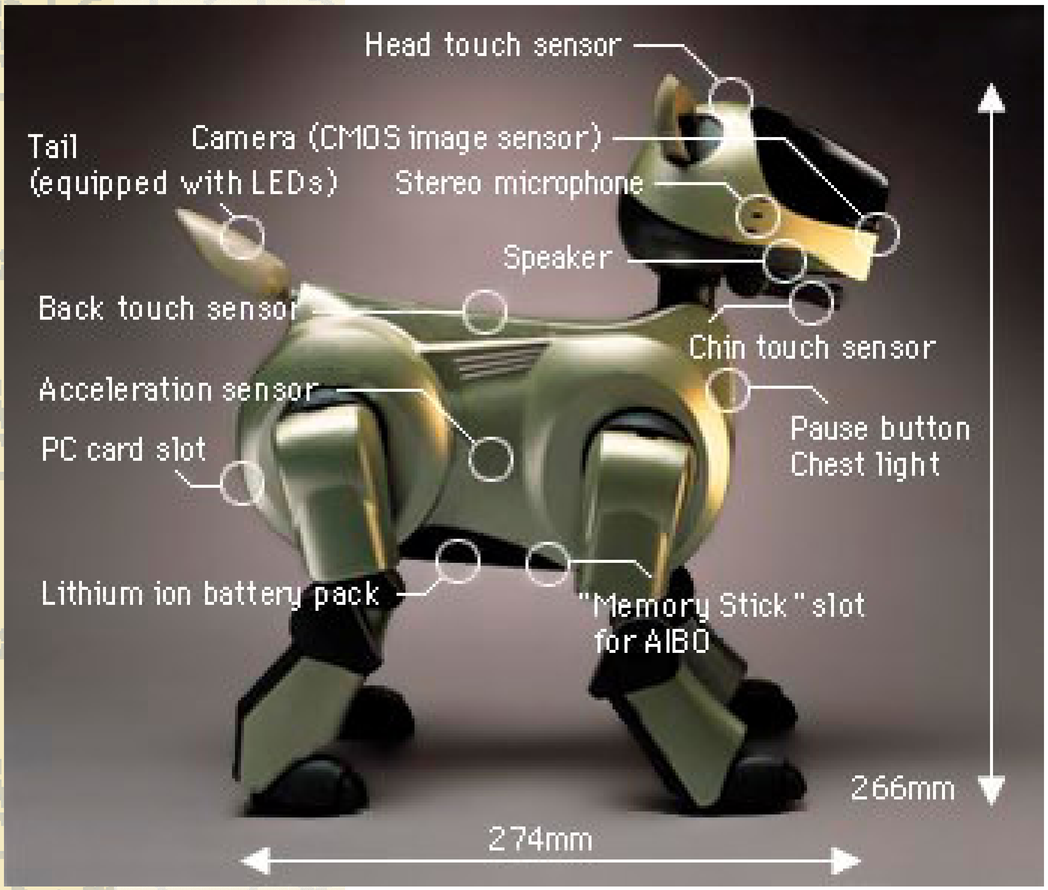


CORAL Research Group
School of Computer Science
Carnegie Mellon University

AIBO-Based Robotics/AI Course

- Why AIBOs?
  - *Relatively* inexpensive at ~$2,000
  - Tremendous hardware/sensing in tiny package
- Practicality / take-home message
  - Provide students with a holistic view of mobile robots
  - Demystify robot programming
  - Show connections to computer science
    - Illustrate *challenges* of introducing explicit sensing and actuation in the real world!
  - Overview of computer vision, embedded systems programming, behavioral control, kinematics, spatial reasoning, multi-robot cooperation
- Robots are *integrated intelligence*:
  - Sensing & perception
  - Motion & kinematics
  - Behavior & cognition
  - Multi-robot cooperation & coordination
- AIBO – a remarkable *complete* robotic platform… everything onboard
  - How do you teach computer science on an *embedded* platform such as the AIBO?
  - "Hacking the real world"

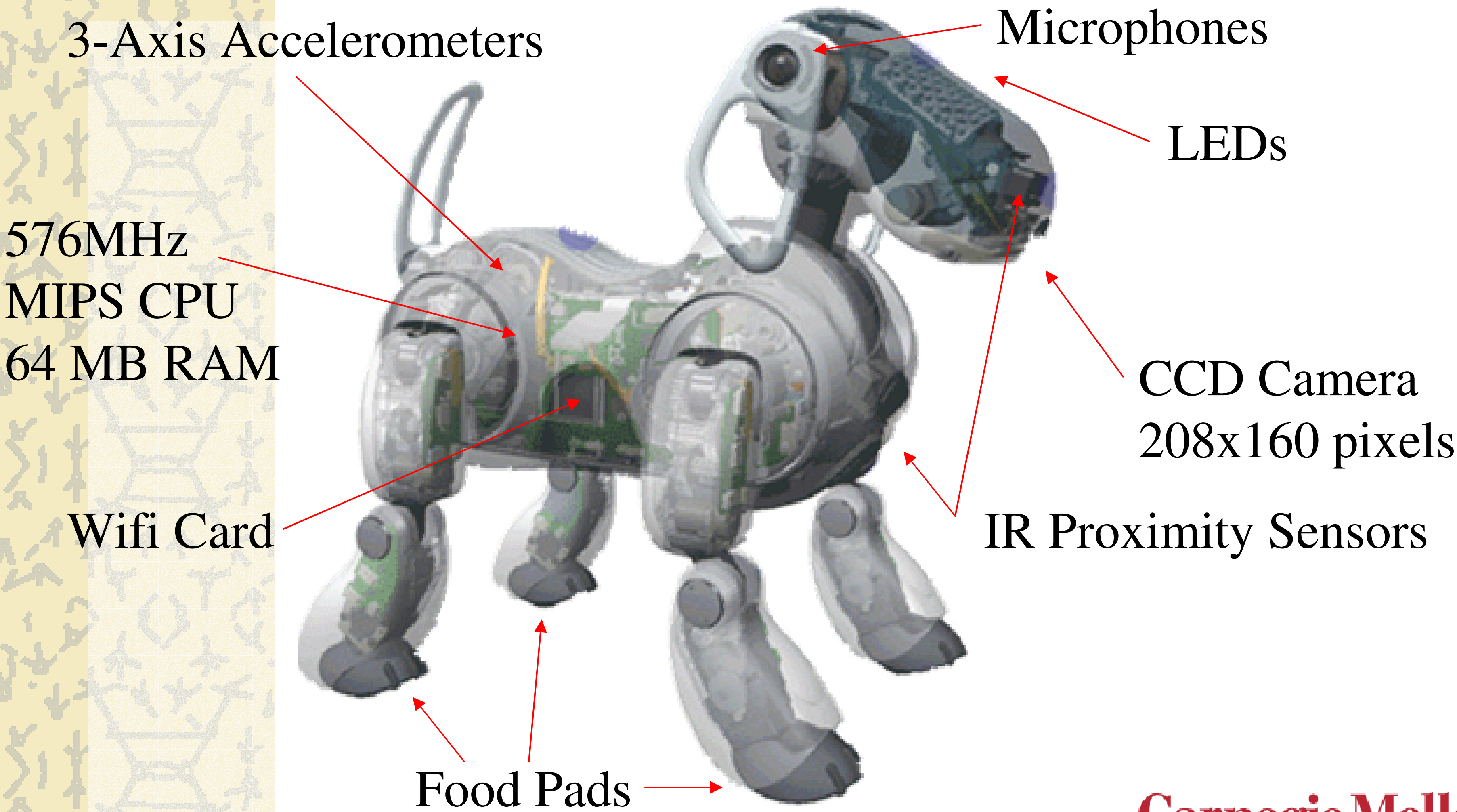
Sony AIBO Robot (ERS-210)



- 384 MHz MIPS processor
- 32MB RAM
- 176x144 pixel camera
- 3-axis accelerometers
- Stereo microphones
- Speaker
- 18 degrees of freedom
  - 3 DOF in each leg
  - 3 DOF in the head
  - 2 DOF in the tail
  - 1 DOF in the jaw
- 802.11b ethernet in PC slot
- Touch sensors

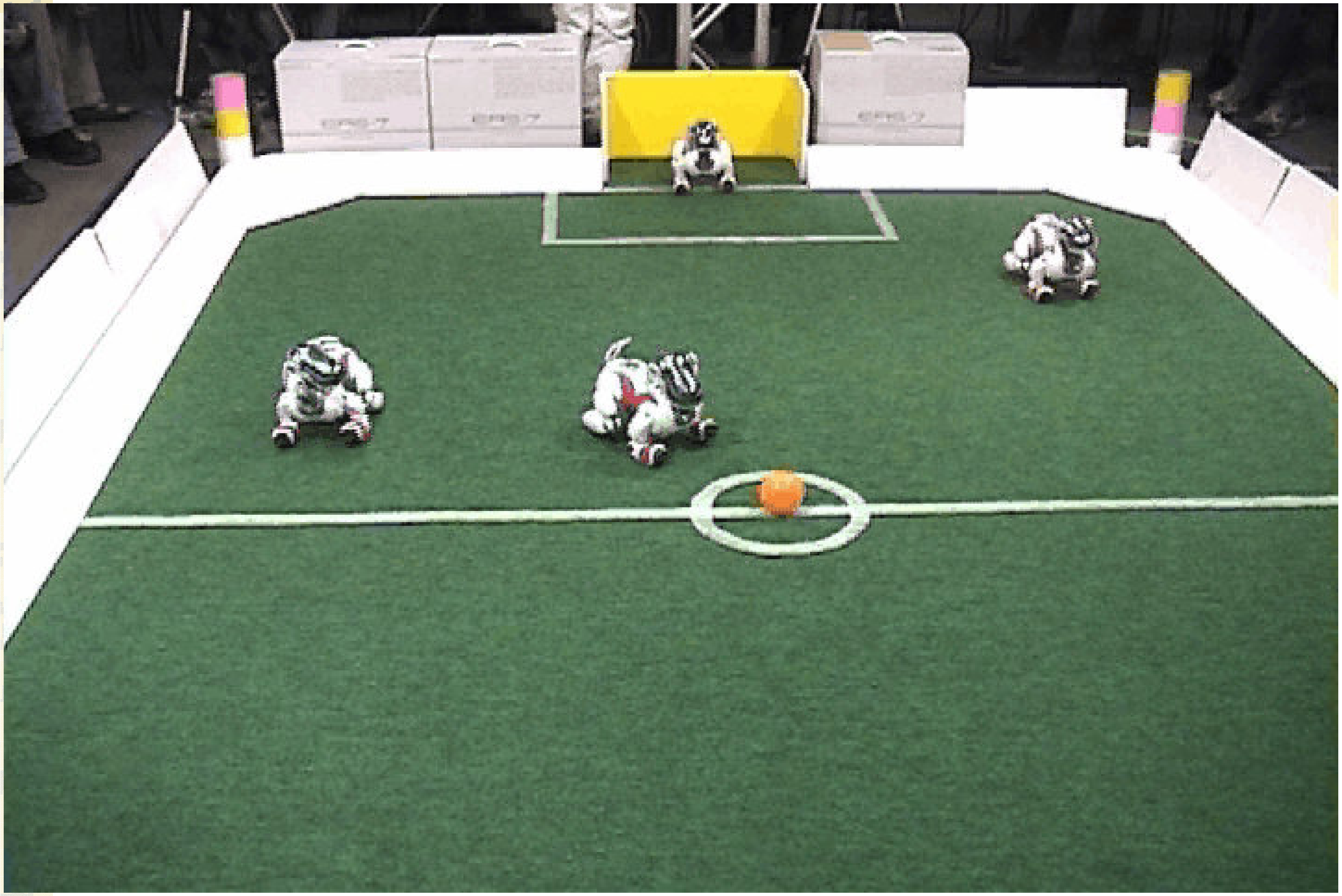
Carnegie Mellon

AIBO ERS-7



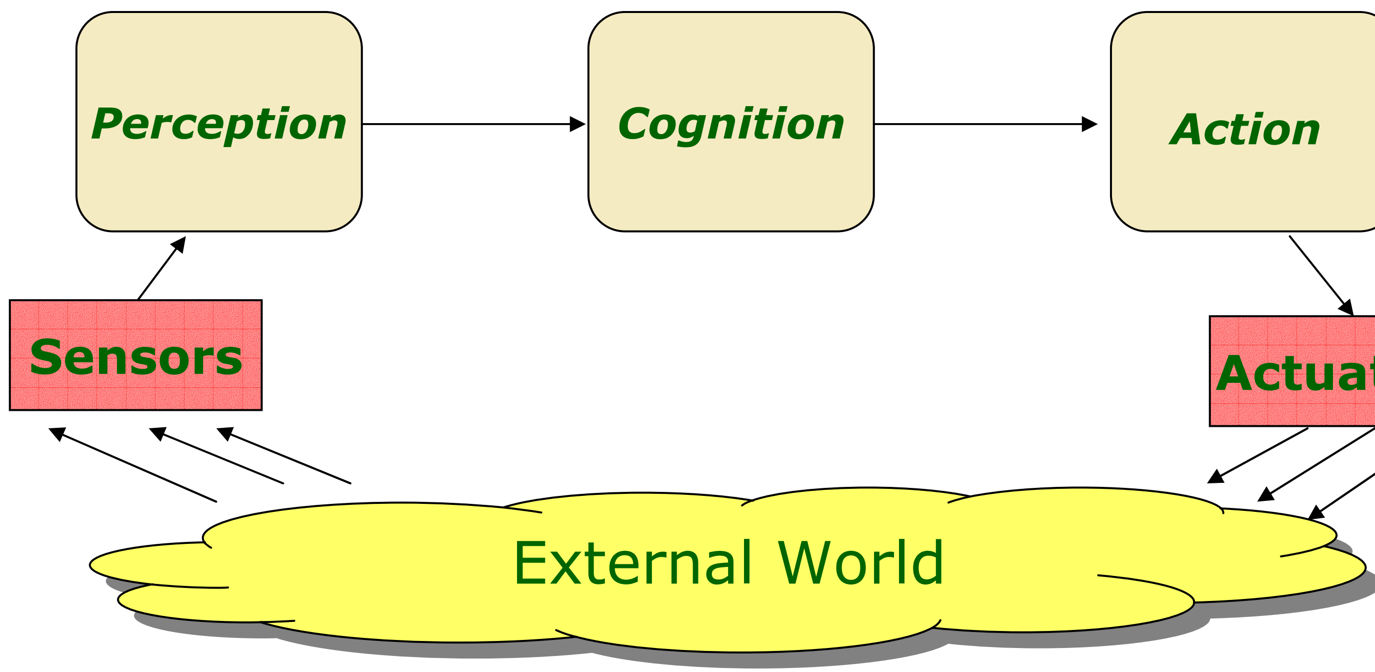
Carnegie Mellon

The “Hook”



Carnegie Mellon

Intelligent Complete Robot



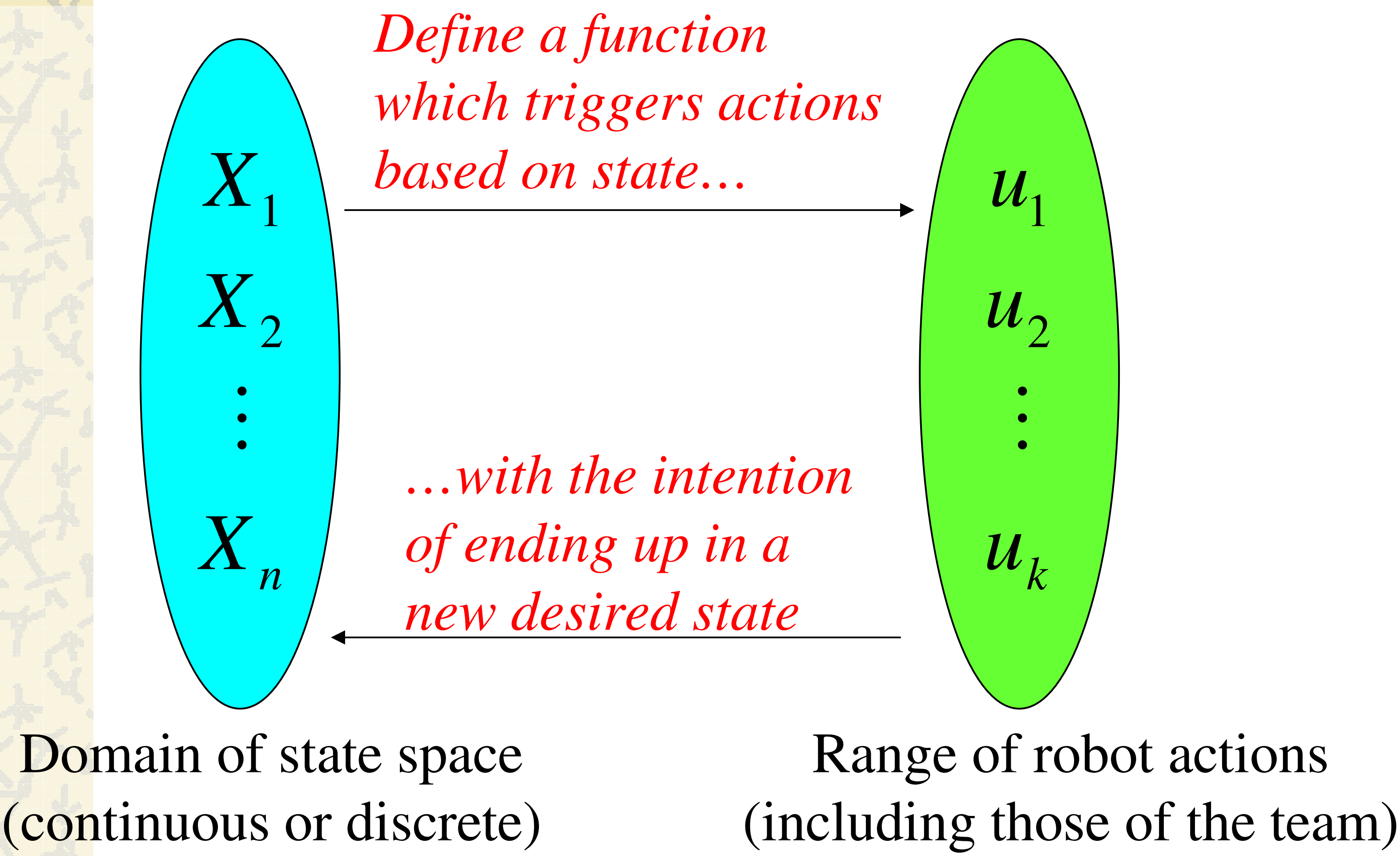
Carnegie Mellon

General Course Overview

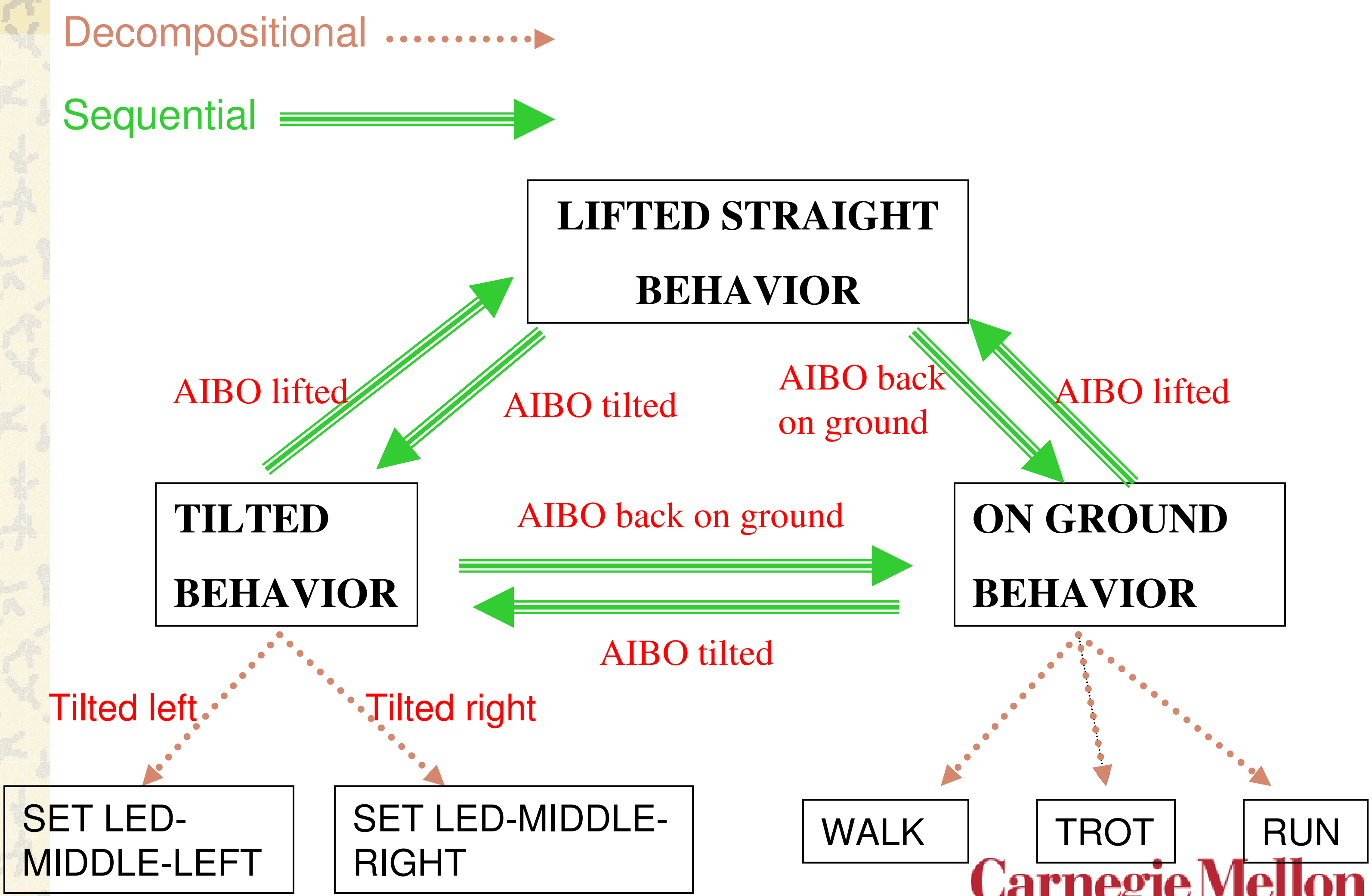
- Behaviors
- Sensors and actuators
- Motion
- Vision
- Localization
- Multi-Robot Cooperation
- Learning

Carnegie Mellon

Behaviors as Functions



Decompositional and Sequential Lnks in a Behavior FSM



Carnegie Mellon

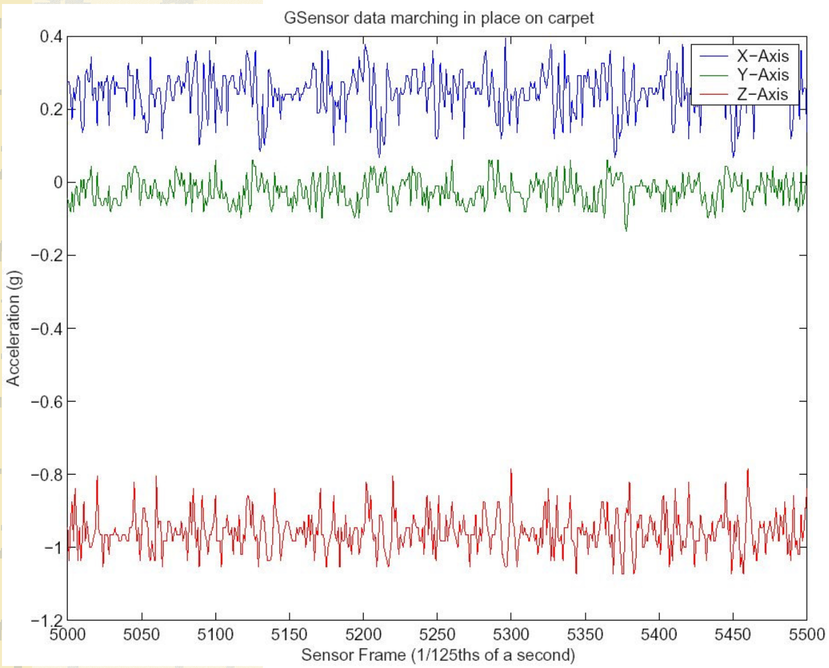
Python Behaviors

```
class Stop(object):
    def __call__(self, features, command):
        command.motion_cmd = Motion.MOTION_STAND_NEUTRAL
        command.head_cmd = Motion.HEAD_SCAN_MARKERS
        command.vx = 0
        command.vy = 0
        command.va = 0
        return (1.0, command)
```

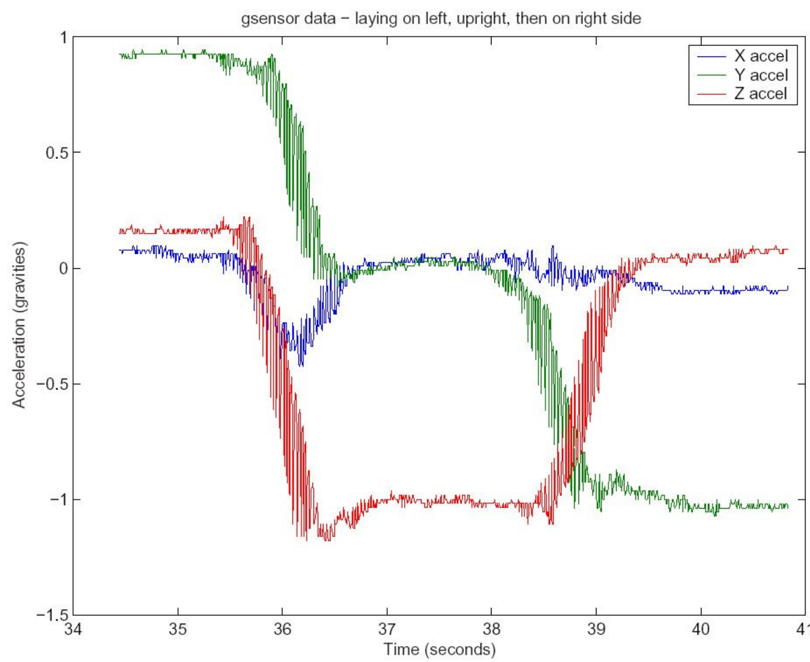
```
fsm = FSM.FSM('state1', ['state1', 'state2', 'state3'])
try:
    fsm.startLoop()
    while fsm.running:
        if fsm.state == 'state1':
            if some_reason:
                fsm.trans('state2', 'some reason')
                continue
            else:
                fsm.endLoop()
        elif fsm.state == 'state2':
            if another_reason:
                fsm.trans('state3', ‘another reason')
                continue
            else:
                fsm.endLoop()
        elif fsm.state == 'state3':
            fsm.endLoop()
except FSMError, error:
    print 'FSM Error:', str(error)
    fsm.endLoop()
```

Sensors : Accelerometer

Robot standing



Robot held on left side, then rotated to upright, then rotated to right side down



Carnegie Mellon

The Motion Interface

Dynamic Walking Motion

Walk Parameters

Walk Engine

Static Frame-Based Motion

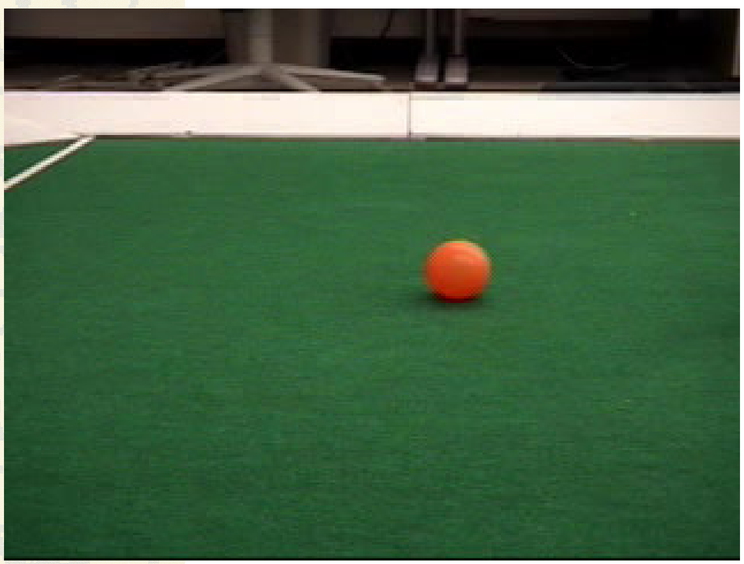
Motion Frames

Frame Interpolator

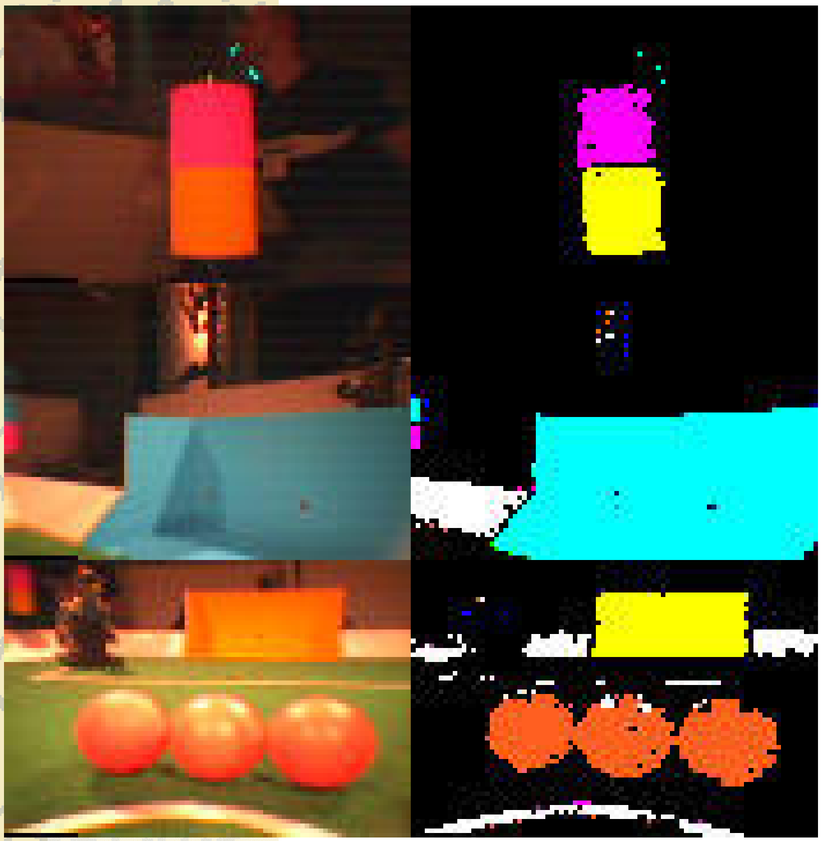
Carnegie Mellon

Motion

- Four-legged actuation for mobility
- Head motion for sensing and object manipulation
- Manipulation of objects (kicking) with all limbs



Vision

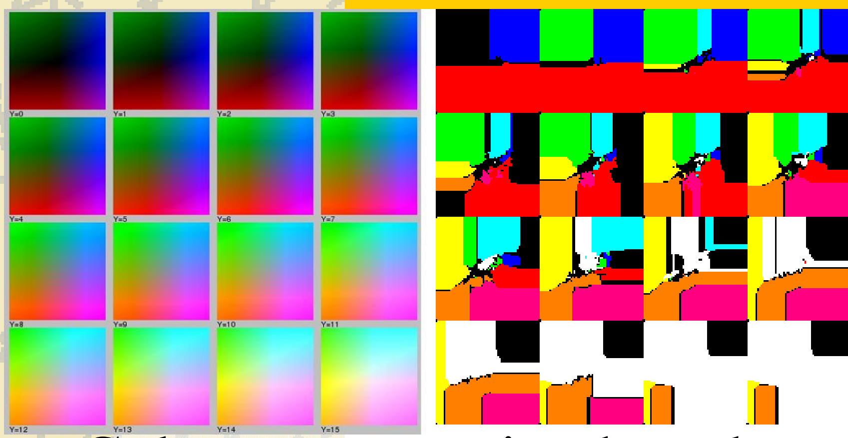


Raw RGB Images

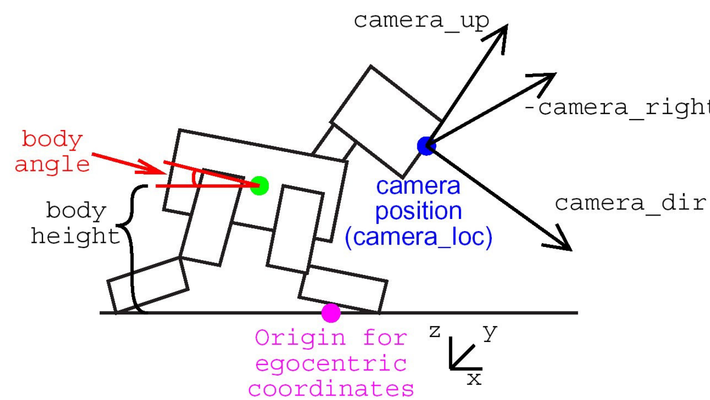
Segmented Images

- Single monocular camera
- Color-based image segmentation
  - All things of interest are classified by their colors
- Object detection and classification
  - Object distance and pose calculated from AIBO joint angles (flat world assumption)
- CMVision2

Vision



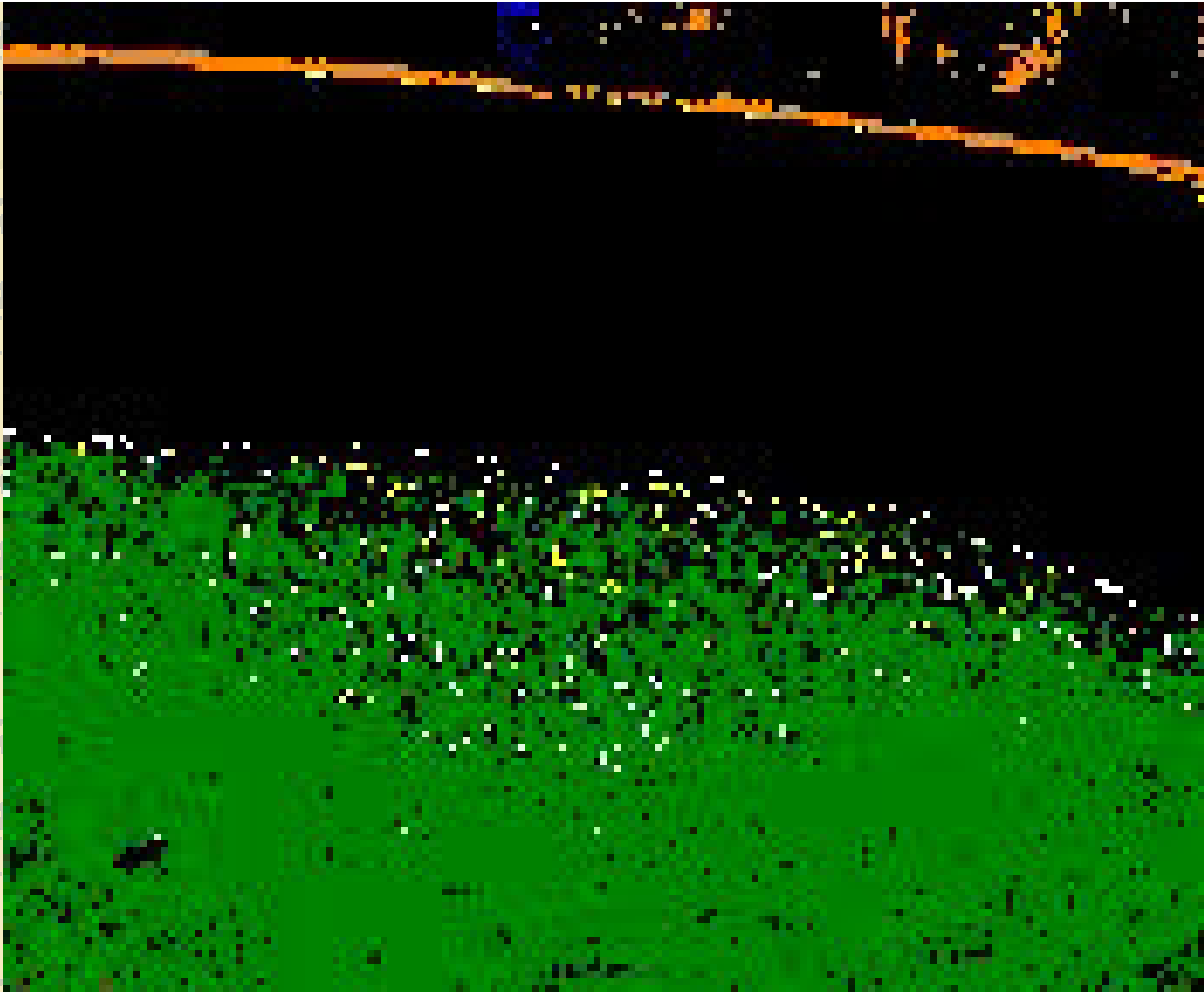
Color segmentation based on YUV color space



Object models based on flat world assumption

Accurate vision in fixed environmental conditions

Carnegie Mellon



on

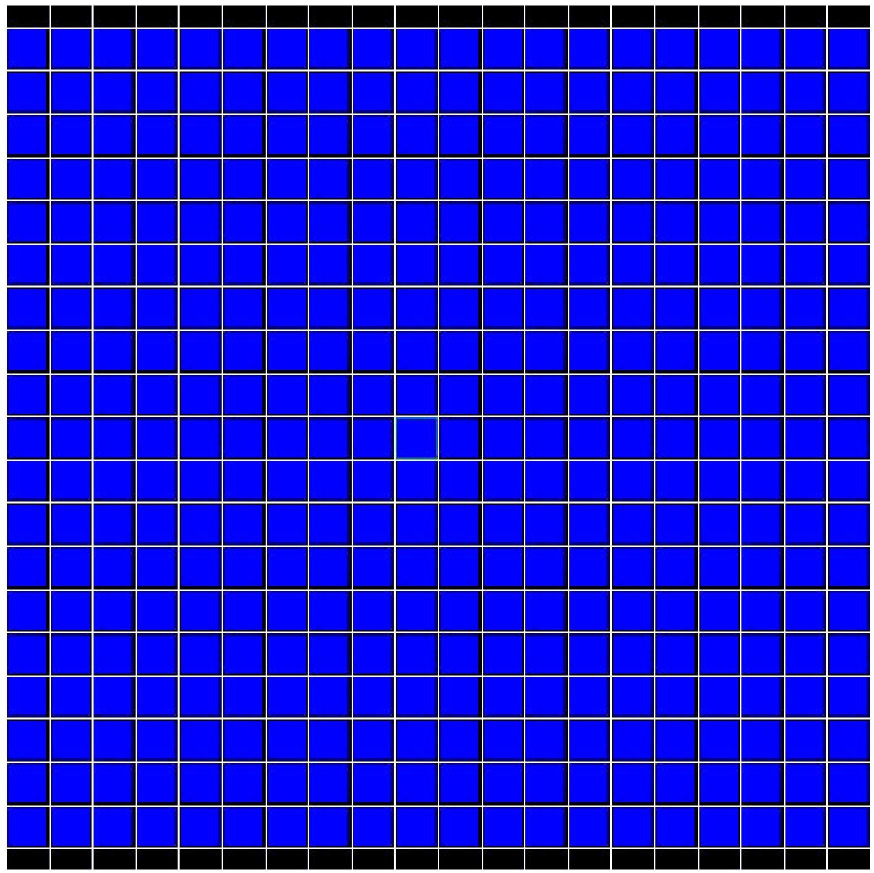
Playing Mastermind with AIBOs

Carnegie Mellon

Vision Local Perceptual Model

Occupancy Grid Key

- Unknown
- Occupied
- Free
- Maybe Occupied



Projected Points Key

- Clear field
- Field wall
- Field line
- Yellow Goal
- Blue Goal
- Opponent
- Teammate
- Obstacle

Freespace and obstacles are stored as points. After 2 seconds, points are forgotten.

Occupancy grid is created by sampling points in grid cells for every frame of video.

More AIBO Tricks

Visual Sonar

Scott Lenser and Manuela Veloso
Computer Science Department
Carnegie Mellon University

October 21, 2003

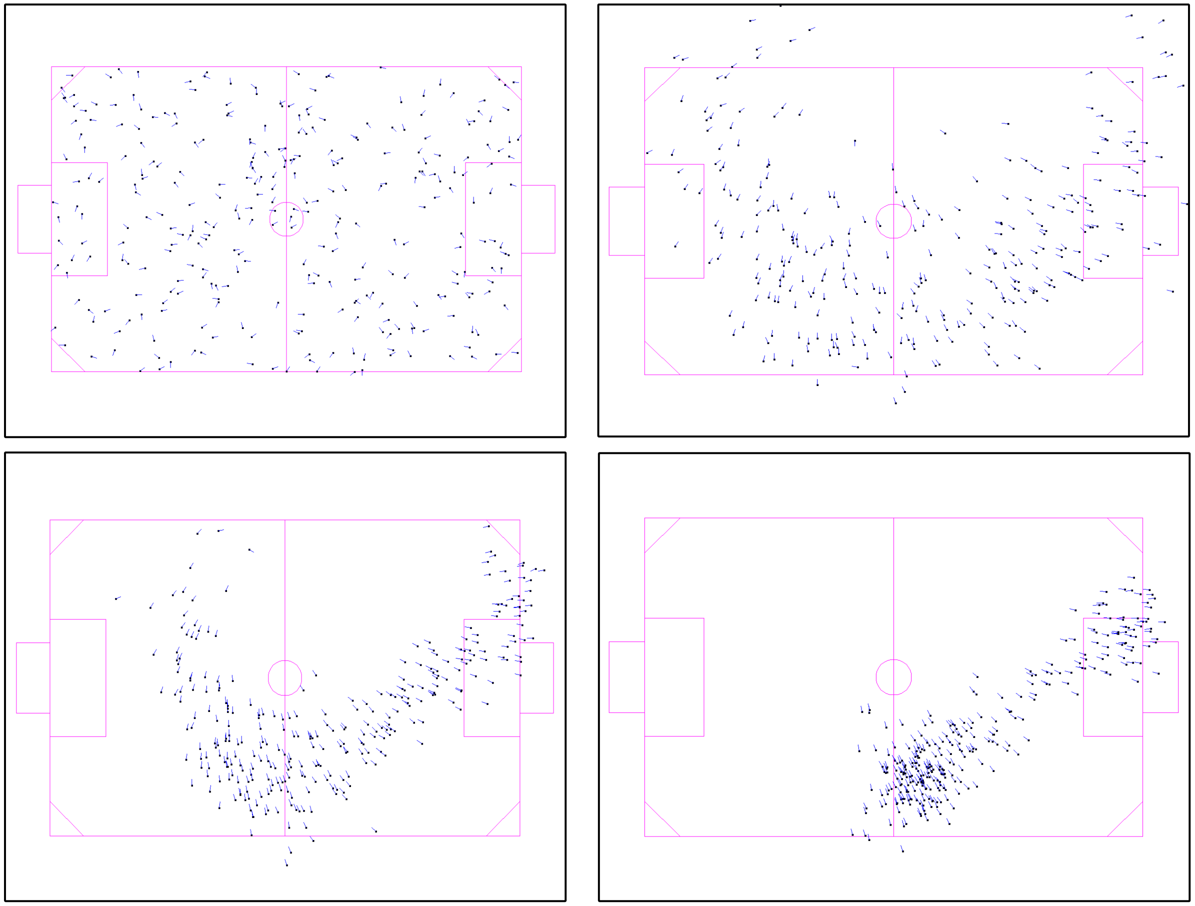
Carnegie Mellon

Localization

- Particle-filter based localization
- Unique landmarks placed around the field
- Quality of localization is based on number and quality of visual readings
- Teach basic probabilistic reasoning, modeling, and Bayesian filtering techniques

Carnegie Mellon

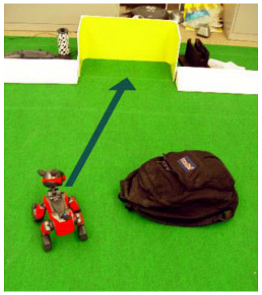
Localization



Carnegie Mellon

Navigating to a Goal

Combines goal pursuit with obstacle contour following to navigate around obstacles

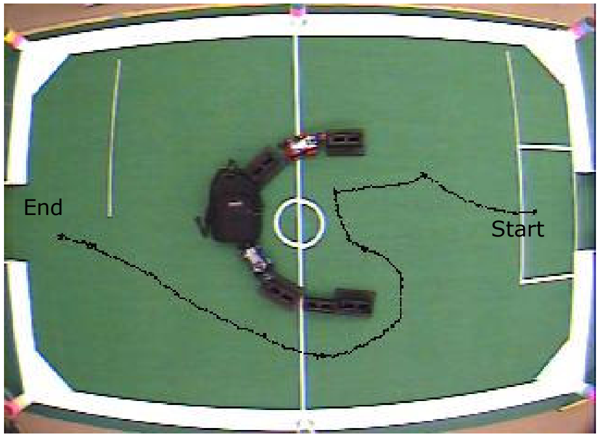


Move directly towards goal if no obstacles block path.

If path is blocked, follow obstacle contour.

Stop obstacle following when straight-line vector to path becomes available.

Avoiding obstacles

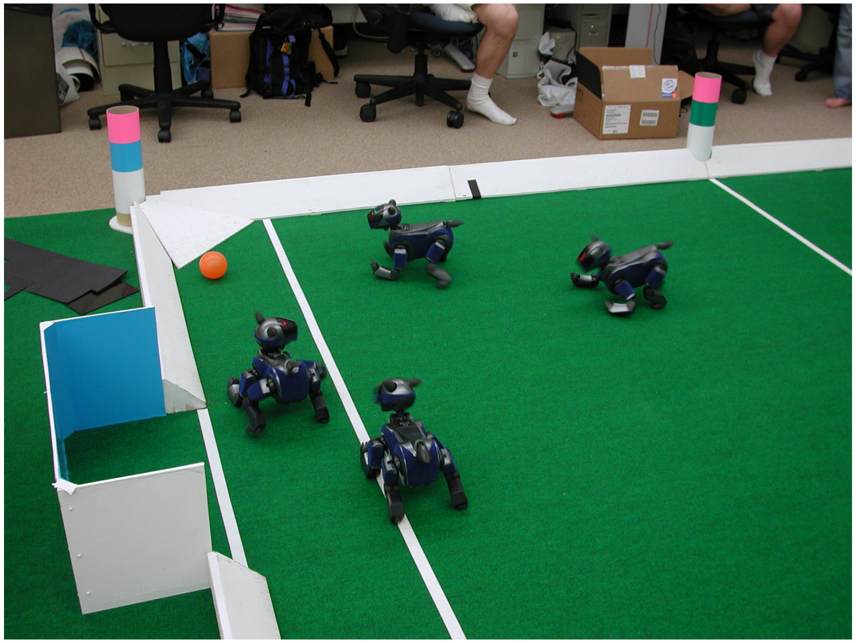


Sample trajectory tracked with overhead camera

Carnegie Mellon

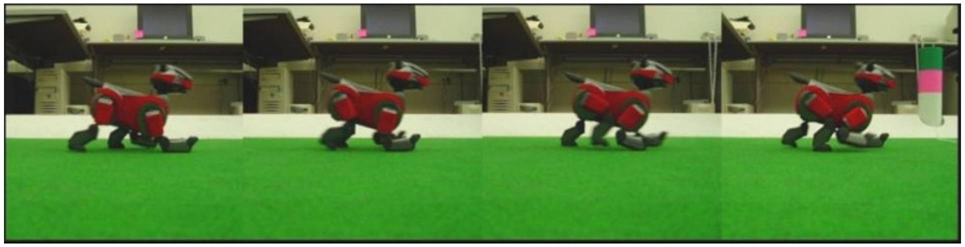
Multi-Robot Cooperation

- Communication
  - Using 802.11b
  - Must be robust to errors
- Cooperative motions
  - Synchronization through state machines
  - Global localization is critical for some tasks



Learning

- Taught mainly as an academic exercise
  - Methods for finding good values for complex parameter sets
- Examples of learning systems are discussed in lectures
- Final project allows some students to explore learning concepts



Walk learning movie

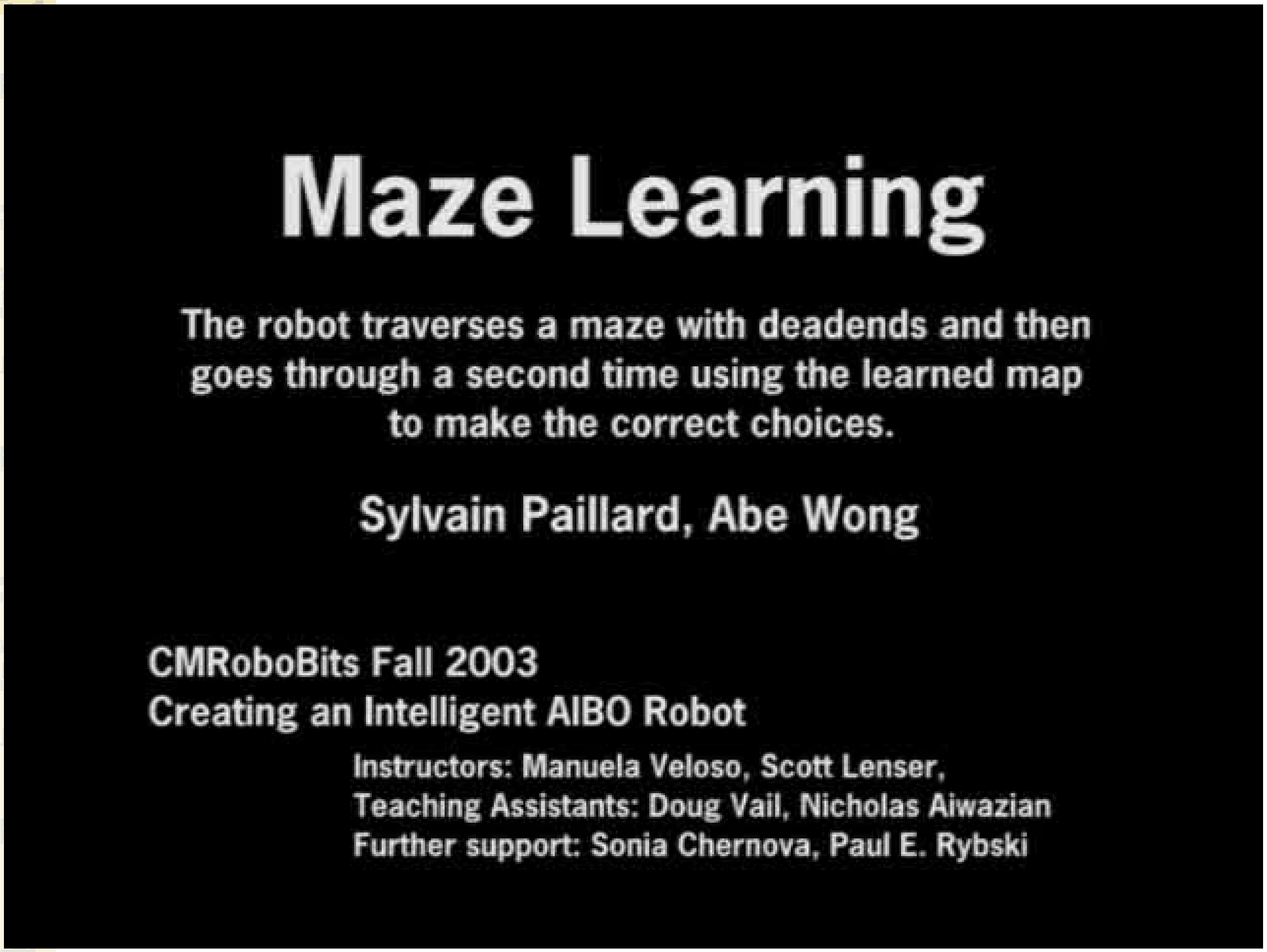
Insights

- Students took to the AIBOs very well
  - Actively installed the compilation tools on their own machines
- Software still requires an expert to maintain it
- Students discovered interesting ways to break software
  - Using Open-R to interact with hardware in strange ways
- C++ as a systems language has lots of gotchas
  - Pointers? What are these pointer things?
- Python is an excellent embedded behavior language
- Never could train students to properly unmount their memory sticks…
- Real-world issues brought out the best (worst?) in student creativity

Multi-Robot Cooperation

Student Project

Student Project



Student Project

Student Project

Further information…

- Paul Rybski
  - http://www.cs.cmu.edu/~prybski
  - prybski@cs.cmu.edu
- CMRoboBits Course Page
  - All course materials are available for download
    - More updates coming!!!
  - http://www.cs.cmu.edu/~robosoccer/cmrobobits
- CMVision2
  - http://www-2.cs.cmu.edu/~jbruce/cmvision/
- CORAL Research Page
  - http://www.cs.cmu.edu/~coral
- OpenR SDK
  - http://openr.aibo.com
- RoboCup
  - http://www.robocup.org

Carnegie Mellon