Team Project

Evaluation Criteria

- **Robot**: A machine that senses, thinks, and acts
- **Platform**: Pleo (hardware), Amigobots, or Player/Stage/Gazebo (software)
- **Team**: 3-4 Students (must get approval for more or less students in a group)
- Project Evaluation Criteria
  - SENSING: 25%
  - THINKING: 35%
  - ACTING: 30%
  - CREATIVITY: 10%

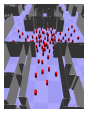
Deliverables

- **September 21st** - One Page Abstract due
  - Team Members
  - Description of Application, including
    - Sensor Suite
    - Robot Platform
    - Environment
- **October 26th** – Progress Report due
  - 2 page update
- **December 5th/7th** – Mon/Wed
  - Team Presentations
  - Provide hard and soft copy of slides
- **December 12th** – Finals Week
  - Final Project Report Due

Simulation Platform

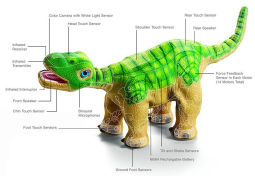
- **Gazebo – 3D multiple robot simulator with dynamics**
  - http://playerstage.sourceforge.net/index.php?src=gazebo
  - Multi-robot simulator for outdoor environments that is capable of simulating a population of robots, sensors and objects in a three-dimensional world. Generates both realistic sensor feedback and physically plausible interactions between objects.

  - **Sensors:**
    - Camera (Stereo pan,tilt and mono), Laser range finder, GPS, Sonars
  - **Robot Platforms:**
    - Pioneer, Peoplebot, Segway, Avatar Helicopter, etc.
  - ***Supported OS: Linux, Mac OSX***



Hardware Platform

- **Pleo Robotic Dinosaur**



Hardware Platform

- **MobileRobots Amigobots**
  - Mobile robot with an onboard microcontroller and sensors that enable the robot to sense what's around it and to safely drive in and around its environment.

(Robot Behavior Learning Through Audio/Visual Cues)

Dance Pleo! Dance!

Aaron Curtis, Adith Srinivasan,
Jaeeun Shim and Eugene Gargas

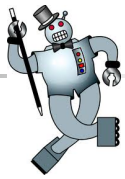
Outline

- Motivation
- Goal
- Platform
- Sensing
- Thinking
- Acting
- Implementation
- Demonstration
- Challenges
- Conclusion

Motivation

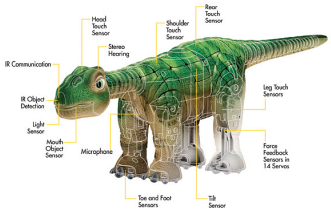
- Personal Robotics Ubiquity
  - Functionality
  - Human-Robot Interaction
    - Programming vs. Teaching
      - Programming = Technical
      - Teaching = Natural
  - Cost
    - More Capabilities = More Cost
    - Less Cost = Greater Acceptance
      - Tradeoffs
      - Do More with Less

Goal



- Teach a robot new dance moves
  - Visual Cues
    - Colored object motions
    - Different colors equal different body parts
  - Audio Cues
    - Beat detection
    - Motion/Beat sensory synchronization
  - Motion
    - Actuate motors synchronizing motion with beat
- Time permitting
  - Adjust moves to different tempos
  - Correct learned behaviors
  - Robots teaching robots

Platform



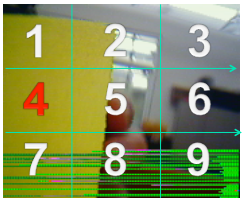
- Pleo
- 14 Joints
(Built for Dancing)
- Life OS & Pawn
  - Behavior Based
    Scripting (Competing Drives determine behavior)
- SD Card Storage (Burn & Learn)
- USB Port (Debugging Lifesaver)

Sensing

- Touch
  - Notification when a body part is tapped
  - Notification when a body part is held

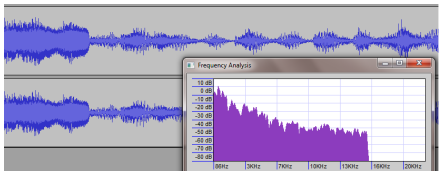
Sensing

- Camera
  - Notification if a trackable object is detected and where.



Sensing

- Microphone x 2
  - Notification of 'loud' noise.
  - Notification of directional change.

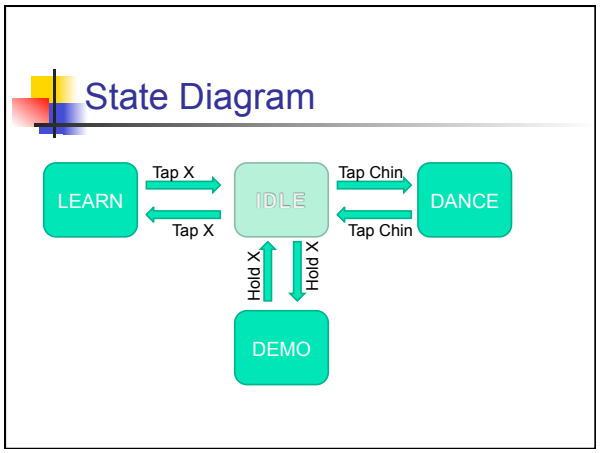


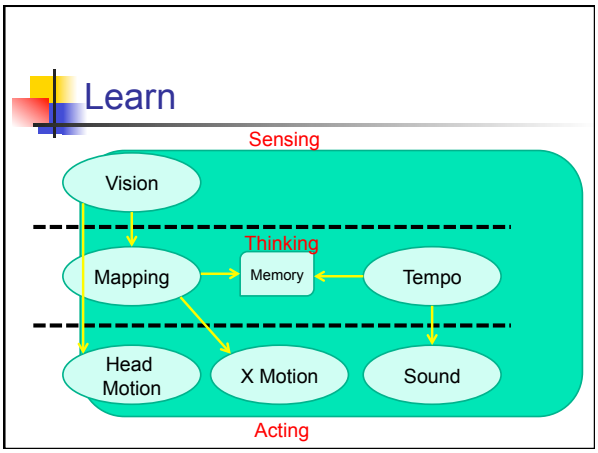
Thinking

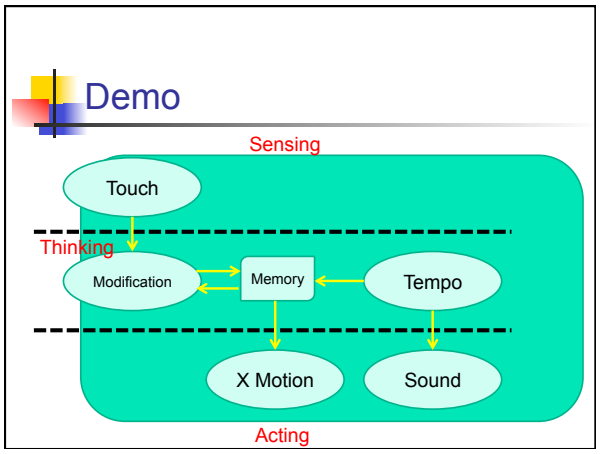
- What do I do now?
- How do I translate perceived motion to a joint?
- How fast should I move?
  - Use desired joint configuration and tempo to determine joint velocities
- Am I doing this right?

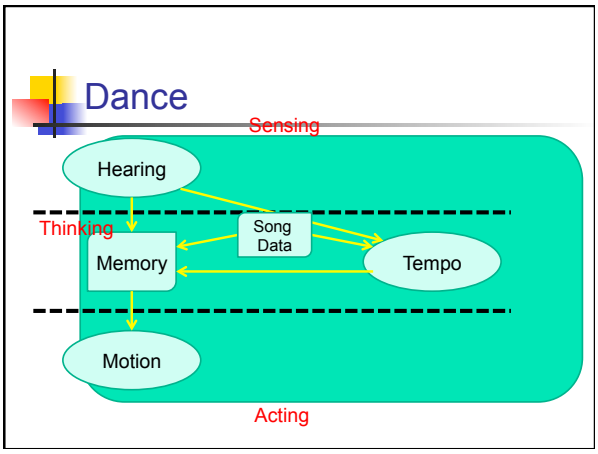
Acting

- I need to let them know what I'm thinking!
  - Use facial expression and audio cues.
- Let's get moving
  - Move joints at each beat.


State Diagram
LEARN
Tap X
Tap X
IDLE
Tap Chin
Tap Chin
DANCE
Hold X
Hold X
DEMO


Learn
Sensing
Vision
Thinking
Mapping
Memory
Tempo
Head Motion
X Motion
Sound
Acting


Demo
Sensing
Touch
Thinking
Modification
Memory
Tempo
X Motion
Sound
Acting


Dance
Sensing
Hearing
Thinking
Song Data
Memory
Tempo
Motion
Acting

DEMONSTRATION

Challenges

- Interface
  - Burn & Learn
- Battery Life
- Audio
  - Poor Resolution / Precision
- Vision
  - Poor Resolution / Tracking
- Motion
  - No Stop!
- Communication (IrDA)

Conclusion

- can **Learn** new dance moves
  - For each joint
  - By sensing visual cues
- can **Act & Correct** the learned motion
  - By demonstrating each joint
  - By sensing the tap motion (right/left arm)
- can **Interact** with human teacher
- can **Dance**!
  - By manipulating every joint's learned motions together
  - Synchronizing with beat
- Future Work?
  - Improving Camera – multiple object tracking
  - Robots teaching robots

Team Pioneer: A Towers of Hanoi Adventure!

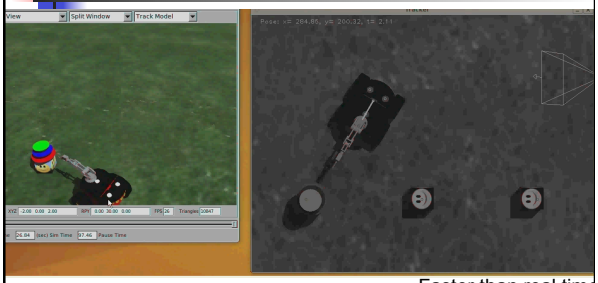
David Lenz, Sebastian Hilsenbeck ,Scott Koziol, Smriti Chopra

Problem Statement

- *We are using Player and Gazebo to simulate a robot solving the classic Tower of Hanoi problem*

http://www.installerapps.com/wp-content/uploads/2007/12/snap_211335.png

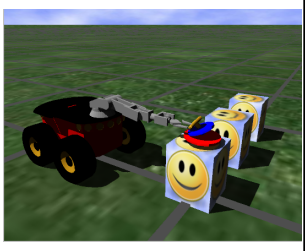
Big Picture: Successful Result



Faster than real time

Project Setup: Gazebo

- *World file includes*
  - *Pioneer robot*
  - *manipulator arm*
  - *disks to create the towers*
  - *Overhead camera*
- *Configuration file includes*
  - *interface information for controlling the arm, pioneer, and camera.*

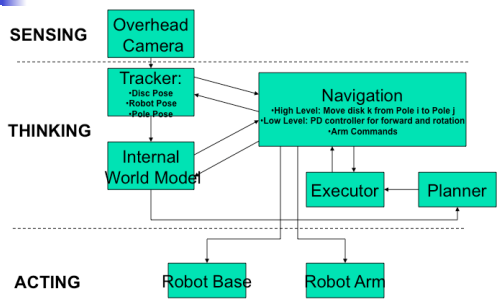


Autonomy

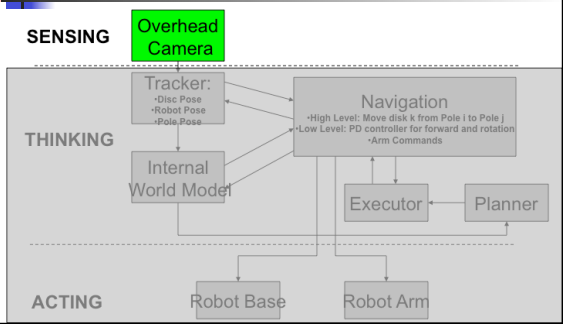
- Robot prior knowledge:
  - A list of potential disk colors
  - Initial estimate of pole position
  - Height of disks

Everything else is computed on-line

System Block Diagram

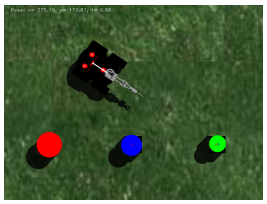


System Block Diagram: Sensing



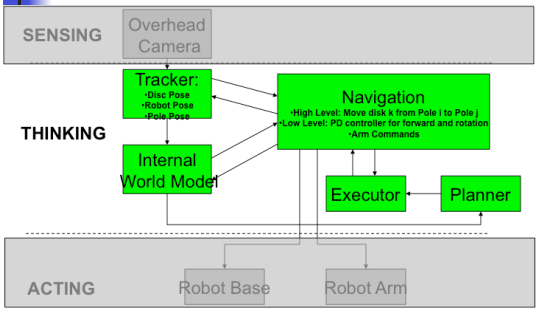
Sensing

- *Computer vision = primary sensor*
- *Assumption:*
  - *There is an overhead camera.*
- *Image processing tasks*
  - *Segment disks from background*
  - *Identify disk size and position*
  - *Identify pole positions*

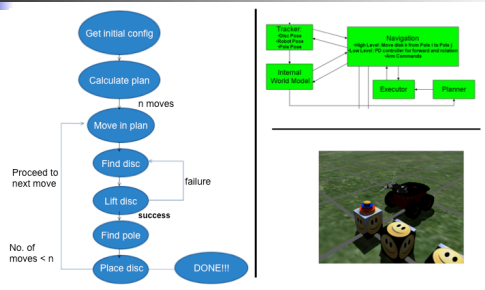


31

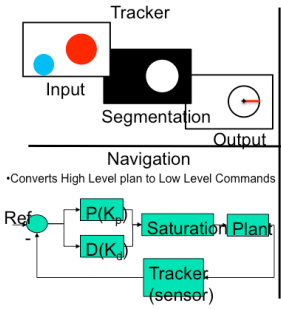
System Block Diagram: Thinking



Thinking: Algorithm Flow



Thinking: Subroutines



World Model

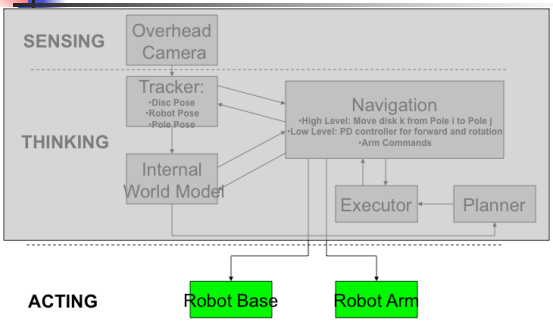
-List of disks with:
Position
Color
Radius
-List of positions of the poles
-Color list

Planner

From Pole 1 to Pole 2
From Pole 1 to Pole 3

http://www.kernelthread.com/projects/hanoi/

System Block Diagram: Acting

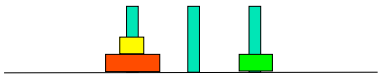


Acting

- Action takes place in the robot frame
  - Forward/reverse velocity
  - Angular velocity
- Arm Joint angles are commanded

Special features

- Potential fields method keeps robot from backing up too far
- Error checking for if the robot drops a disk
- Planner feature searches existing plan for potential solution when all disks are not on the same pole.



Georgia Institute of Technology

BOTHOVEN

"The Musical Robot"

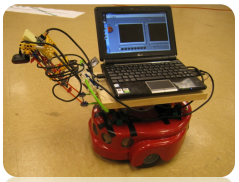
(An Autonomous Music Finder and Player)

Team Members:
Dinh Bao Nguyen
Brian Post
Rahul Ravu
William Thoreau

Georgia Institute of Technology

Outline

- Overview
- Description of methodology
  - Sensing
  - Thinking
  - Acting
- Results
- Conclusion
- Class discussion



Georgia Institute of Technology

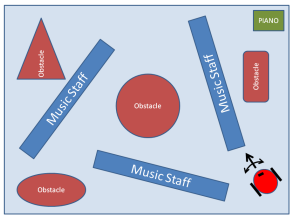
Motivation

- Hardware platform
- Behaviors
- Music

- The concept:
  - Bothoven only wants to play music. He scours his environment in search of scores to read.

Overview

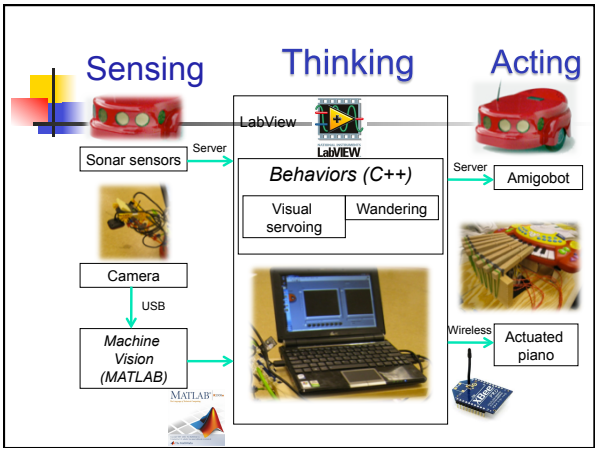
- Robot
  - 2 behaviors
    - Wandering
    - Visual Servoing
- Piano
  - Playing Data Sent by Bothoven



Challenges

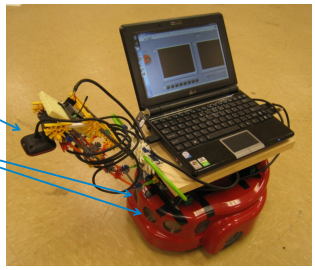
- Build an automated piano
- Communication
  - PC - Amigobot
  - PC - piano
- Find the music score
- Follow the line
- Play the right notes

- 5 different programming languages!



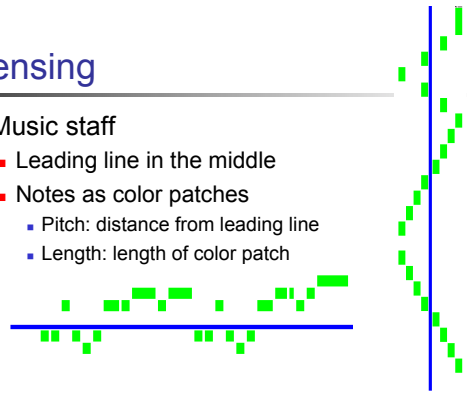
Sensing

- Overhead camera
- 8 sonar sensors
- Camera
- Sonars



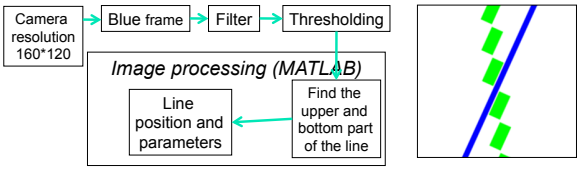
Sensing

- Music staff
  - Leading line in the middle
  - Notes as color patches
    - Pitch: distance from leading line
    - Length: length of color patch



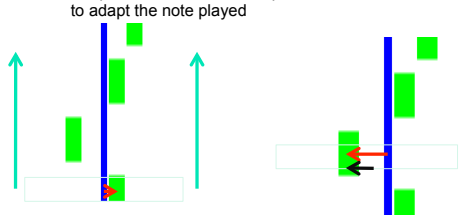
Sensing

- Line following
  - Line finding
  - Angle detection
  - Position of the line in the Image



Sensing

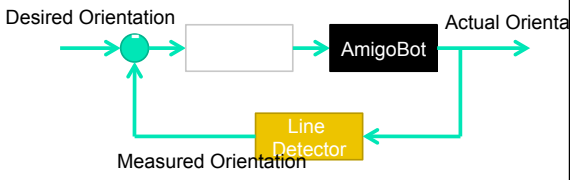
- Notes reading
  - Real time: fast algorithm (see video)
  - Adaptive: uses the relative position of the line in the image to adapt the note played



Georgia Institute of Technology

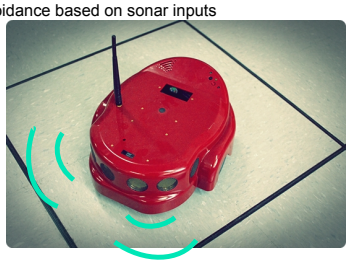
Thinking

- Behaviors: Visual servoing and wandering
- Visual servoing
  - Proportional controller for line following
    - With Gain Scheduling



Thinking

- Wandering
  - Moving in random direction
  - Obstacle avoidance based on sonar inputs



Thinking

Line detection

Line detected
Angle of the line relative to the robot orientation
Position of the line in the Image (left/right/centered in the image)

Gain scheduling

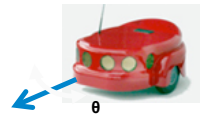
Gain scheduling:
Allows to associate different gains according to the sharpness of the turn required to catch the line.

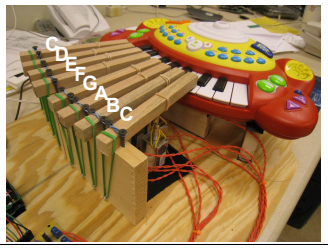
3 Gains used: - wander mode (high)
- line following with high correction needed (medium)
- line following with little correction needed (low)

Georgia Institute of Technology

Acting

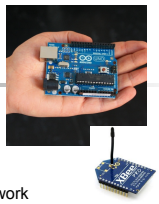
- Amigobot is moving
- Piano playing (via the actuation of the keys)

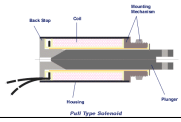




Acting

- Piano Details:
  - Control
    - Arduino Micro Processor
    - Xbee Wireless Point to Point Network
  - Actuation
    - 8 Individually Actuated 12 Volt 1/4 inch Stroke Solenoids
    - Mechanical Fingers





Results

- Obstacle Avoidance
  - With the Exception of One Server Crash all Obstacles Were Avoided
- Line Detection
  - Almost Every Line Detected With Proper Orientation
- Line Following
  - Convergence is a Bit Slow
    - Gains Limited by Update Rate
    - PID Control Would Offer Better Performance
- Note Detection
  - Largely Dependant on Note Spacing and Lighting
  - Also Dependant on Robot Orientation
- Playing Of Notes:
  - Every Note Identified Was Played Successfully
- Longest Continuous Test ~ 45 mins

Issues faced

- PC – Amigobot Communication
- MATLAB scipt embedded in LabView with Basic Functionalities
- Gain Tuning for Line Following
- Speed of the Image Processing Algorithm
- Speed of the Sonar Sensors
- Choosing the Right Colors for the Music

Interesting Notes:

- Since it Performs Music in the Key of C-Major Only, any Misread Notes Sound Like Improvisation
- Time Delay Between Image Processing and Acting Results in Oscillation When Line Following

Videos

Conclusion

- It Works!!
  - Sensing
    - Bothoven Was Able to Sense the Obstacles, Notes, and Lines
  - Thinking
    - It Used the Sensed Information to determine the Appropriate Action
  - Acting
    - It Moved And Played Notes Correctly

Ideas For Discussion

- Possible Future Expansion:
  - Multi Robot Collaboration
    - Different Instruments Improvising or Synchronously Playing a Tune
  - Interactivity
    - User Tells Bothoven What Song to Play and it Finds and Plays it