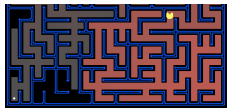
CS 188: Artificial Intelligence
Fall 2010

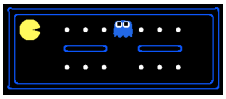
Advanced Applications:
Robotics / Vision / Language

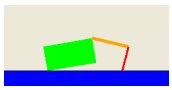
Dan Klein – UC Berkeley

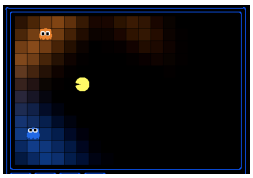
Many slides from Sebastian Thrun, Pieter Abbeel, Jitendra Malik

So Far: Foundational Methods

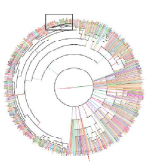


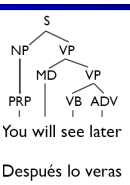






Now: Advanced Applications





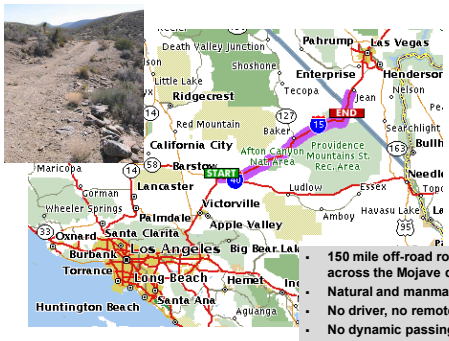
[DEMO: Race, Short]

Autonomous Vehicles

Autonomous vehicle slides adapted from Sebastian Thrun

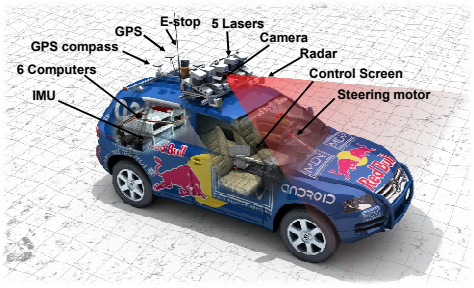
[DEMO: GC Bad, Good]

Grand Challenge: Barstow, CA, to Primm, NV

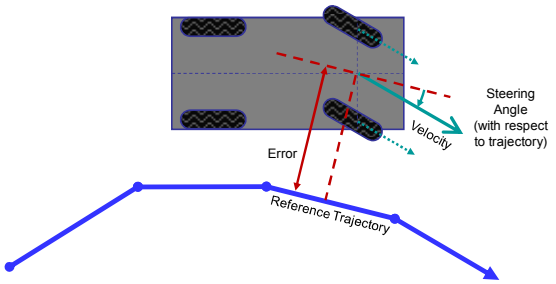


- 150 mile off-road robot race across the Mojave desert
- Natural and manmade hazards
- No driver, no remote control
- No dynamic passing

An Autonomous Car

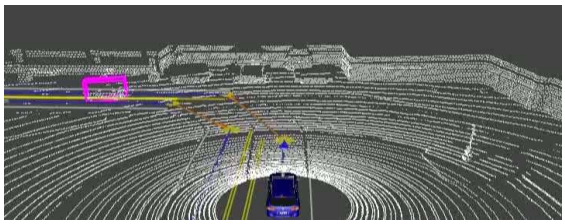


Actions: Steering Control

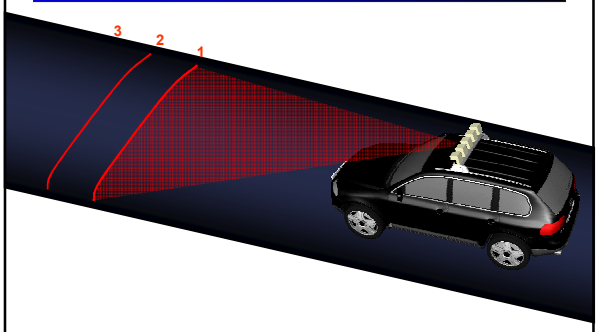


[DEMO: LIDAR]

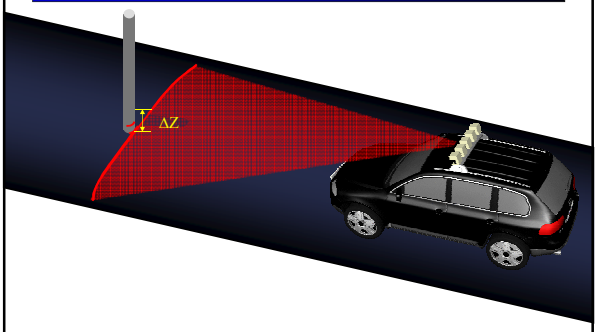
Sensors: Laser Readings



Readings: No Obstacles

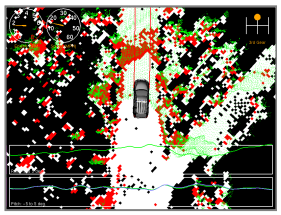


Readings: Obstacles



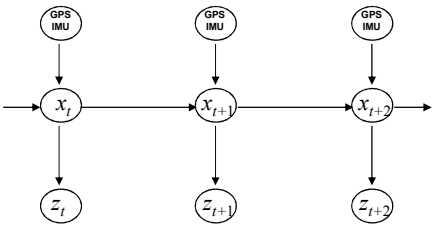
Obstacle Detection

Trigger if $|Z^i - Z^j| > 15\text{cm}$ for nearby z^i, z^j

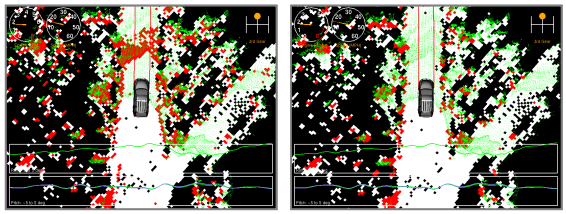


Raw Measurements: 12.6% false positives

Probabilistic Error Model



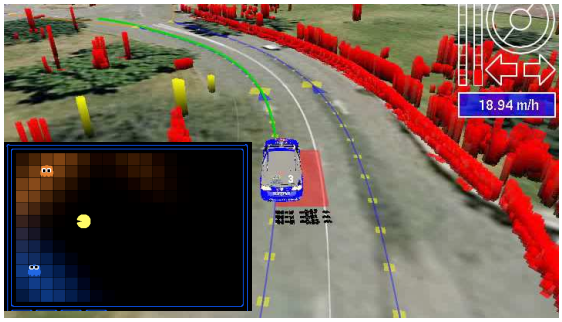
HMMs for Detection



Raw Measurements: 12.6% false positives

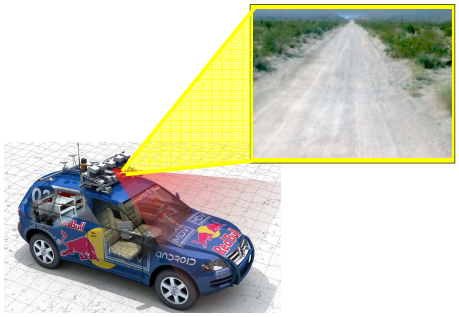
HMM Inference: 0.02% false positives

Environmental Tracking

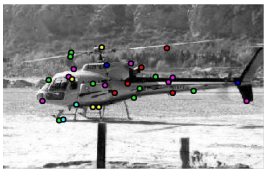


[DEMO: PEOPLE]

Sensors: Camera



Object Recognition

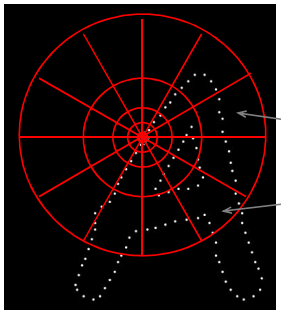


Template

Query

Vision slides adapted from Jitendra Malik

Shape Context



Count the number of points inside each bin, e.g.:

Count = 4

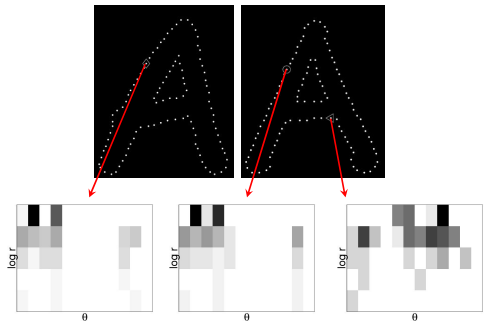
⋮

Count = 10

- Compact representation of distribution of points relative to each point

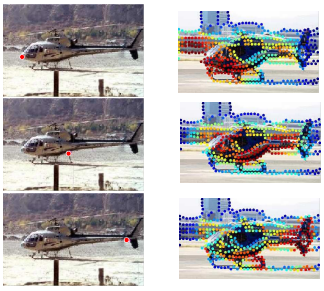
20

Shape Context



21

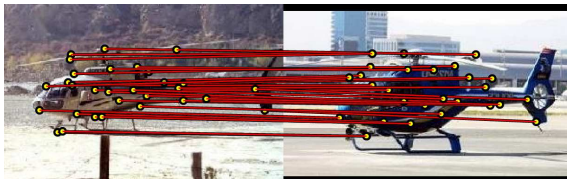
Similar Regions



Color indicates similarity using local descriptors

22

Match for Image Similarity



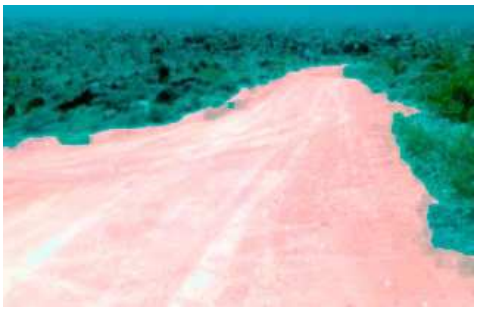
23

[DEMO: LIDAR 1]

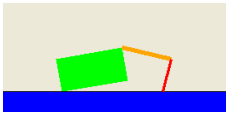
Vision for a Car

[DEMO: LIDAR 2]

Self-Supervised Vision



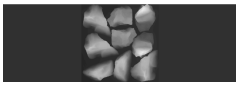
Complex Robot Control



[demo – quad initial]

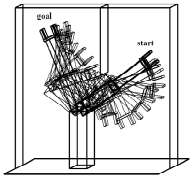
Robotic Control Tasks

- Perception / Tracking
  - Where exactly am I?
  - What's around me?
- Low-Level Control
  - How to move from position A to position B
  - Safety vs efficiency
- High-Level Control
  - What are my goals?
  - What are the optimal high-level actions?



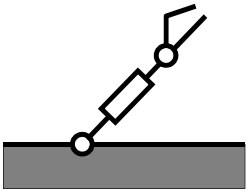
Low-Level Planning

- Low-level: move from configuration A to configuration B



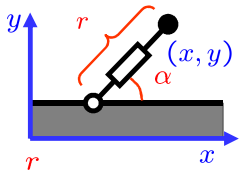
A Simple Robot Arm

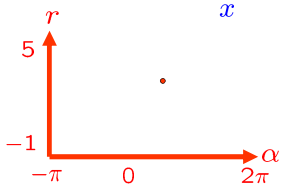
- Configuration Space
  - What are the natural coordinates for specifying the robot's configuration?
  - These are the *configuration space* coordinates
  - Can't necessarily control all degrees of freedom directly
- Work Space
  - What are the natural coordinates for specifying the effector tip's position?
  - These are the *work space* coordinates



Coordinate Systems

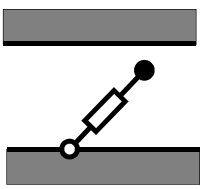
- Workspace:
  - The world's (x, y) system
  - Obstacles specified here
- Configuration space
  - The robot's state
  - Planning happens here
  - Obstacles can be projected to here

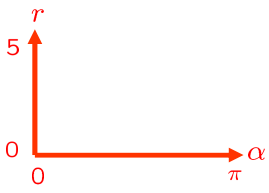




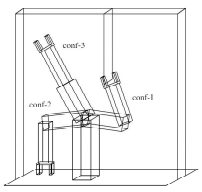
Obstacles in C-Space

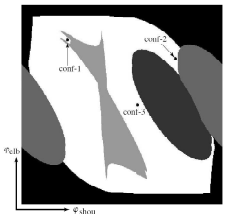
- What / where are the obstacles?
- Remaining space is *free space*





Example: A Less Simple Arm

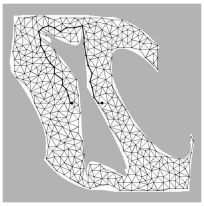




[DEMO]

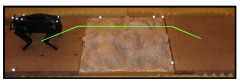
Probabilistic Roadmaps

- Idea: sample random points as nodes in a visibility graph
- This gives *probabilistic roadmaps*
  - Very successful in practice
  - Lets you add points where you need them
  - If insufficient points, incomplete or weird paths

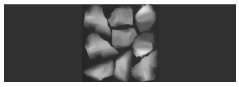


High-Level Control

- Demonstrate path across the “training terrain”

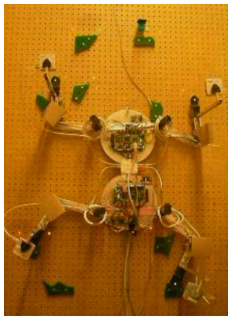


- Run apprenticeship learning to find a set of weights w
- Receive “testing terrain” (a height map)



- Find a policy for crossing the testing terrain.

High DOF Robots [DEMOS]



Videos from Pieter Abbeel, Jean-Claude Latombe